

**AJUSTES NOTIFICAR NOTAS DEVOLUTIVAS Y CTL**

Actualmente en el aplicativo SIR no existe una fase donde se puedan enviar los turnos de registro que han sido devueltos al público e inscritos parcialmente y que requieren ser notificados al ciudadano, con el fin de dar cumplimiento a lo dispuesto por el código administrativo y de procedimiento administrativo, Artículo 25 Ley 1579 de 2012, en lo concerniente a la notificación de notas devolutivas, de acuerdo con el comunicado SNR2019EE013151 del 14 de marzo enviado por el Director Técnico de Registro, se debe crear una nueva fase, con el fin de enviar a esta fase los turnos con resolución devueltos y que es necesario notificar.

El sistema tendrá un nuevo Rol llamado Notificador Notas Devolutivas a donde lleguen todos los turnos que sean devueltos al público y donde la nota devolutiva debe ser notificada al usuario, el turno debe estar en estado devuelto y con su respectiva nota devolutiva; esto desde el rol Registrador una vez aprobada y firmada la nota Devolutiva. Este nuevo Rol será creado para que se envíen los turnos que cumplan esta condición a una nueva fase llamada Devolución y dentro de esta se deben poder enviar los turnos a otras dos fases llamadas Notificar Nota Devolutiva y Nota Devolutiva Notificada. Adicionalmente, en el Rol Registrador se debe crear una tercera fase llamada Recursos Nota Devolutiva. Debe ser un rol denominado Recursos Nota Devolutiva que se pueda signar al registrador jurídico o quien haga las veces de jefe jurídico .En la Fase Nota Devolutiva Notificada, en esta fase el turno permanecerá hasta que se interponga el recurso o se presente renuncia a términos y en caso contrario, el turno avanzará a la siguiente fase (Certificados Asociados Registro).

Se realizan ajustes sobre este requerimiento :

- \* Verificación del orden de prioridad en firma registro.
- \* Solución a BUG al avanzar turnos individuales devueltos en firma registro con turnos posteriores en notificador de notas devolutivas.
- \* Cambio de mensaje de aviso de vencimiento para las Oficinas Origen en Nota Devolutiva Notificada.

Se ajusta el CTL para que se genera con los datos de las áreas: Area, Area construida, Area privada, se muestre los valores con su respectiva descripción de medida de área.

ESTADO DEL FOLIO: **ACTIVO**

COD CATASTRAL: SIN INFORMACION

**DESCRIPCIÓN: CABIDA Y LINDEROS:**

PISO12 CON AREA DE 52 COEFICIENTE DE PROPIEDAD 5% CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN DOCUMENTO 1, 2012/08/07, DPTO POLICIA - SIJIN MONTELIBANO. ARTICULO 8 PARÁGRAFO 1º. DE LA LEY 1579 DE 2012

**LINDEROS TECNICAMENTE DEFINIDOS:****AREA Y COEFICIENTE**

AREA: 10 HECTAREAS 200 METROS CUADRADOS 120 CENTIMETROS CUADRADOS

AREA PRIVADA: 150 METROS CUADRADOS 193 CENTIMETROS CUADRADOS

AREA CONSTRUIDA: 200 METROS CUADRADOS 100 CENTIMETROS CUADRADOS

COEFICIENTE: 0.987634

**COMPLEMENTACIÓN:****DIRECCIÓN DEL INMUEBLE:**

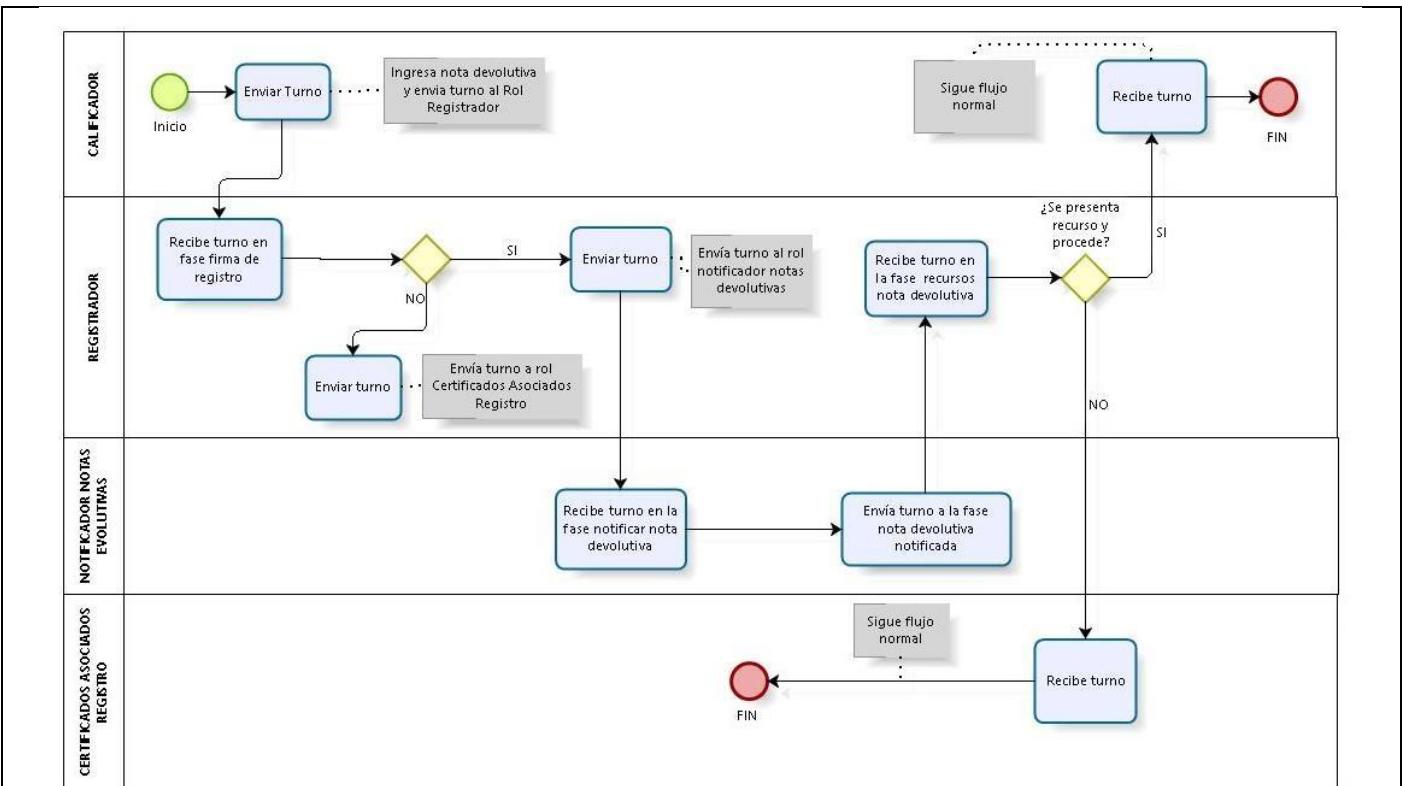
TIPO DE PREDIO: RURAL

DETERMINACION DE INMUEBLE: SIN DETERMINAR

DESTINACIÓN ECONOMICA: SIN DETERMINAR

SUPERINTENDENCIA  
DE NOTARIADO  
& REGISTRO  
*La guarda de la fe pública*

 <b>Pear</b> <b>Solutions</b>	 <b>MÉDIA</b> <i>commerce</i> <small>Conecta oportunidades</small>	<b>DESPLIEGUE 9.0</b> <b>04/08/2022</b>	<b>SNR</b> SUPERINTENDENCIA <b>DE NOTARIADO &amp; REGISTRO</b> <small>La guarda de la fe pública</small>
<b>NOTIFICAR NOTAS DEVOLUTIVAS</b>			
<p>Actualmente en el aplicativo SIR no existe una fase donde se puedan enviar los turnos de registro que han sido devueltos al público e inscritos parcialmente y que requieren ser notificados al ciudadano, con el fin de dar cumplimiento a lo dispuesto por el código administrativo y de procedimiento administrativo, Articulo 25 Ley 1579 de 2012, en lo concerniente a la notificación de notas devolutivas, de acuerdo con el comunicado SNR2019EE013151 del 14 de marzo enviado por el Director Técnico de Registro, se debe crear una nueva fase, con el fin de enviar a esta fase los turnos con resolución devueltos y que es necesario notificar.</p>			
<p>El sistema tendrá un nuevo Rol llamado Notificador Notas Devolutivas a donde lleguen todos los turnos que sean devueltos al público y donde la nota devolutiva debe ser notificada al usuario, el turno debe estar en estado devuelto y con su respectiva nota devolutiva; esto desde el rol Registrador una vez aprobada y firmada la nota Devolutiva. Este nuevo Rol será creado para que se envíen los turnos que cumplan esta condición a una nueva fase llamada Devolución y dentro de esta se deben poder enviar los turnos a otras dos fases llamadas Notificar Nota Devolutiva y Nota Devolutiva Notificada. Adicionalmente, en el Rol Registrador se debe crear una tercera fase llamada Recursos Nota Devolutiva. Debe ser un rol denominado Recursos Nota Devolutiva que se pueda signar al registrador jurídico o quien haga las veces de jefe jurídico .En la Fase Nota Devolutiva Notificada, en esta fase el turno permanecerá hasta que se interponga el recurso o se presente renuncia a términos y en caso contrario, el turno avanzará a la siguiente fase (Certificados Asociados Registro).</p>			
<b>FLUJO NOTIFICADOR NOTAS DEVOLUTIVAS</b>			



Los tiempos que se establecen para NOTIFICAR NOTA DEVOLUTIVA son: 40 días hábiles (Citación 5 días hábiles; Aviso 5 días hábiles; Publicación 5 días hábiles; Recurso 10 días hábiles) Solamente se creará el campo de los 40 días hábiles para el proceso de Notificar Nota Devolutiva, sin necesidad de crear el detalle de las subfases, ya que el manejo de los 40 días será responsabilidad de las ORIP.

Después de haber notificado la nota devolutiva, los tiempos establecidos para presentar el Recurso a la Nota Devolutiva son de 10 días hábiles y debe estar parametrizado en el sistema. Por lo tanto, si el recurso llega después de los 10 días, el sistema no debe permitir ingresarla y se debe mostrar un mensaje donde diga que ya no se puede presentar el recurso porque el tiempo está vencido. Se va a radicar el recurso en SIR y asociar al turno devuelto Nota: Los 10 días hábiles para presentar el recurso se deben contar a partir del día hábil siguiente a la notificación. Una vez presentado el recurso, el funcionario de la oficina de registro tiene 2 meses calendario para resolverlo, contados a partir de la fecha radicación del recurso. Por lo tanto, se debe mostrar una alerta informando al registrador y/o al funcionario que tenga el turno que el tiempo para resolver el recurso, está próximo a vencerse o está vencido, es decir se debe asociar la fecha y radicación del recurso con el turno devuelto. Por otro lado, para el recurso de apelación, la oficina internamente podrá tomar el tiempo necesario que considere para resolverlo ya que no hay un tiempo establecido para esto.

## PANTALLA REGISTRADOR

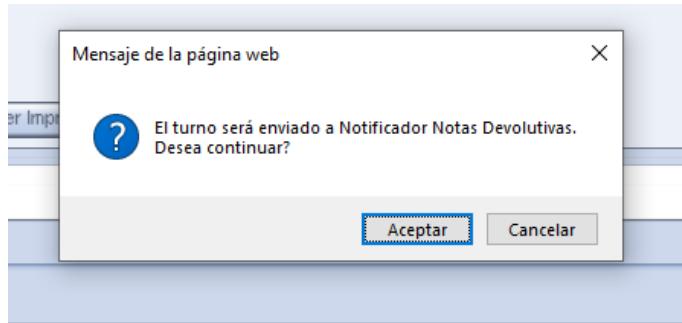
Cuando el turno llega al rol registrador y cumpla con las condiciones de turno devuelto y la nota Devolutiva es aceptada y firmada por el registrador o con solo seleccionar el botón notificador notas Devolutivas y enviar el turno el sistema hace estas acciones se debe mostrar en la pantalla la opción de enviarlo al nuevo rol; NOTIFICADOR NOTAS DEVOLUTIVAS. Ver Imagen 1. En la pantalla registrador deben continuar las demás funcionalidades entre estas devolver a calificación, tener en cuenta la funcionalidad de avanzar turnos por relación cuando son devueltos al público, el nuevo botón de Notificador Notas Devolutivas solo permitirá enviar turnos con resolución devueltos al público y el botón aceptar los turnos diferentes a devueltos se avanzaran.



Para avanzar al Rol NOTIFICADOR NOTAS DEVOLUTIVAS, se debe ingresar una nota informativa que cumpla con las condiciones ya existentes.

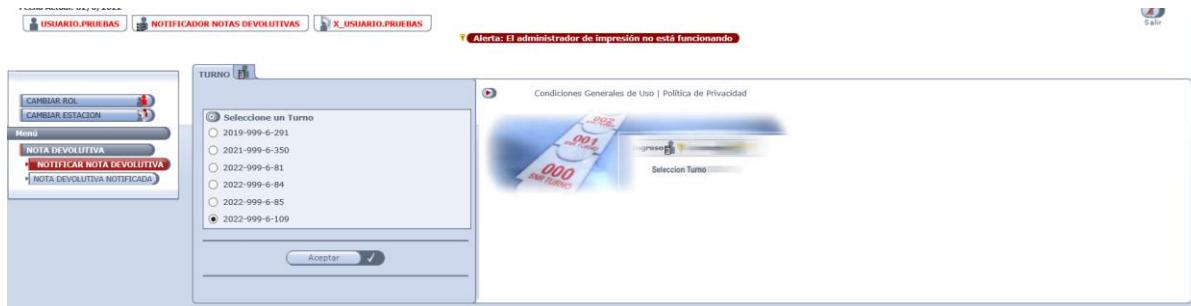
### ROL NOTIFICADOR NOTAS DEVOLUTIVAS

En el rol NOTIFICADOR NOTAS DEVOLUTIVAS llegan todos los turnos que hayan sido objeto de devolución luego de pasar por el rol REGISTRADOR, el sistema informará que el turno devuelto será enviado al rol notificador notas devolutivas, Ver imagen.



### SUB-FASE NOTIFICAR NOTA DEVOLUTIVA

Una vez se avanza el turno llega a la fase Notificador Notas Devolutivas a la sub-fase Notificar Nota Devolutiva, donde se listan los turnos recibidos del Rol Registrador, ver imagen.



**NOTA:** El sistema maneja una validación de 40 días hábiles desde la fecha en la que el turno fue avanzado a esta fase. Además, debe mostrar los días restantes para la gestión de la notificación de la nota devolutiva, Cuando el plazo de los 40 días se haya cumplido, la alerta debe notificar que se han vencido, Ver imagen.



**Alertas** !

**Se han vencido los 40 días para la gestión de notificar nota devolutiva.**

Al tomar el turno el sistema solicita los datos de notificación como fecha de notificación (Esta será la fecha del día que se toma el turno), destinatario: usuario u oficina origen, Tipo de notificación: Personal o Correo electrónico, para el caso de elegir usuario se solicitará si es apoderado, nombre, apellido, tipo de documento, numero de documento, dirección y teléfono, para oficina origen se visualizara la oficina origen que emitió el documento del turno y solicitará dirección y teléfono, ver imagen.

*Notificación a usuario.*

*Notificación a oficina origen*

**NOTA:**

Si el turno se encuentra en esta sub-fase (notificar nota devolutiva) el aplicativo permitirá re anotar el turno desde pantallas administrativas.

El sistema solicitará una nota informativa obligatoria para avanzar el turno, ver imagen. (es necesario previo a diligenciar los datos de notificación agregar la nota informativa obligatoria)

Cuando el funcionario que tenga el rol entrega correspondencia debe haber generado los oficios de envío a la oficina origen y realizar el respectivo descargue del turno, deberá informar inmediatamente al funcionario que tenga este rol (notificación notas devolutivas), para que ejecuté los datos de notificación oficina origen.

Se debe agregar la nota informativa de tipo notificar nota devolutiva.

El turno se avanza a la siguiente sub fase Nota Devolutiva Notificada, donde se listan los turnos enviados desde la sub-fase Notificar Nota devolutiva.

## SUB-FASE NOTA DEVOLUTIVA NOTIFICADA

Al momento que llegue el usuario a la ventanilla de correspondencia, con el fin de interponer un recurso ya sea personal o por correo, el funcionario tiene la obligación de comunicar inmediatamente al funcionario que tenga asignado el rol NOTIFICACION NOTAS DEVOLUTIVAS, para que pueda seleccionar que SI presenta RECURSO Y avanzar el turno en la FASE NOTA DEVOLUTIVA NOTIFICADA.

Al tomar el turno el sistema visualiza el resumen de los datos de la notificación registrada en la sub-fase anterior, y habilita la opción de selección de interponer SI o NO recurso, ver imagen.

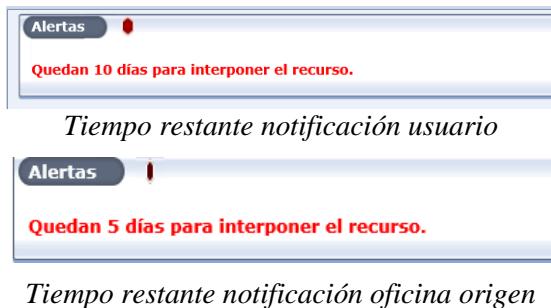
Si NO se interpone recurso, se elige la opción NO y al dar clic en botón aceptar el sistema avanzara el turno a la fase de Certificados Asociados para continuar con su flujo normal.

**NOTA:** En caso de que se renuncie a términos se debe indicar claro y expreso en la respectiva nota informativa de que se renuncia a términos Y SELECCIONAR LA OPCION NO.

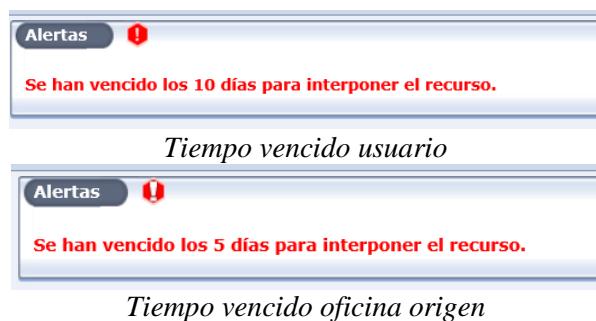
En caso de que, SI se interponga recurso y al dar clic en el botón aceptar el sistema avanzara el turno a la fase de Recursos nota devolutiva, para continuar con el proceso de interponer un recurso.

**NOTA:** Por cada notificación de nota devolutiva se establece una alerta con un plazo de 10 días hábiles, si es notificación para usuario y 5 días hábiles si es notificación para oficina origen, Por cada día se debe informar los días

restantes para el vencimiento del plazo, Ver imagen



Cuando el plazo de los 10 días para notificación de usuario y 5 días para notificación de oficina origen se haya cumplido, el sistema visualiza una alerta informando que se ha superado el tiempo establecido para interponer un recurso, Ver imagen.



El sistema solicitará una nota informativa obligatoria para avanzar el turno a la siguiente fase, ver imagen



Alerta Nota informativa.

Se debe agregar la nota informativa de tipo Nota informativa documentos usuario o Nota informativa documentos oficina de origen, de acuerdo al destinatario de la nota devolutiva; usuario u oficina origen, ver imagen.



Nota informativa

## ROL RECURSOS NOTA DEVOLUTIVA

Al ingresar a la fase recursos nota devolutiva el sistema lista los turnos enviados desde la fase notificador notas devolutivas, ver imagen.

*Recursos nota devolutiva*

Al tomar el turno el sistema visualiza el resumen de los datos de la notificación registrada en la fase de notificación nota devolutiva, y habilita la opción de ingresar los datos del recurso, ver imagen.

Se solicita los siguientes datos:

- Tipo de recurso: Recurso de reposición, Recurso de reposición de subsidio de apelación o no procede recurso.
- Usuario recurso.
- Observación recurso.
- Fecha de recurso.

Para un Recurso de reposición se ingresa el recurso correspondiente y sus datos:

*Ingreso recurso.*

Una vez se agreguen los datos del recurso, este se lista en el historial de recursos del turno, permitiendo a su vez eliminarlo o editararlo mediante los botones de editar o eliminar, (al editar el recurso se debe dar clic en el botón guardar de lo contrario el sistema alertara que tiene algún recurso en edición y no permite continuar) adicionalmente se debe diligenciar la opción:

- Procede recurso: Si o No. (una vez resuelto el recurso se selecciona esta opción.) ver imagen.

*Editar o eliminar recurso*

Al agregar el recurso y este permanezca en esta fase el sistema alertara el tiempo de 60 días (2 meses calendario) para resolver el recurso, ver imagen.

*Tiempo de recurso.*

Resuelto el recurso, el sistema solicita agregar La resolución y fecha ejecutoria, de manera obligatoria, para avanzar el turno, ver imagen.

Resoluciones				
<input type="checkbox"/>	<input type="checkbox"/>	Número	123456789	Fecha
				03/06/2022
<input type="button" value="Eliminar Seleccionadas"/>				
<input type="checkbox"/>	Número	<input type="text"/>	Fecha	<input type="text"/> 
<input type="button" value="Agregar Resolución"/>				
<input type="checkbox"/>	Fecha Ejecutoria	<input type="text"/> 		

## *Resolución.*

**NOTA:** La fecha ejecutoria debe ser mayor a la fecha de la resolución, de lo contrario el sistema mostrara una alerta, ver imagen.

**● Error en la Validación de Datos del Formulario**

**● Los Siguientes Errores han sido encontrados:**

- La fecha ejecutoria debe ser mayor a la fecha de la última resolución

[« Volver](#)

*Alerta fecha ejecutoria.*

NOTA: Para avanzar el turno el sistema solicitará una nota informativa obligatoria de tipo Recurso de Reposición, ver imagen.

**! Error en la Validación de Datos del Formulario**

**Los Siguientes Errores han sido encontrados:**

- Debe ingresar una nota informativa de tipo RECURSO DE REPOSICION

[« Volver](#)

### *Nota informativa recurso de reposición.*

Se ingresa la nota informativa correspondiente para poder avanzar el turno, ver imagen.

**Agregar Nota Informativa**

**Notas Informativas** 

**Nota** 

<input type="button" value="Escoger Tipo de Nota"/>	Z3192 - NOTA INFORMATIVA RECURSO DE REPOSICION	<input type="button" value="NOTA INFORMATIVA RECURSO DE REPOSICION"/>
<b>Descripción</b>		
PRUEBA DE NOTA INFORMATIVA DE RECURSO DE REPOSICIÓN.		









### *Ingreso nota informativa*

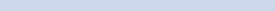
Finalmente, al ingresar los datos del recurso, la resolución, fecha ejecutoria, nota informativa y seleccionar SI procede, el turno es avanzado y regresa a la fase de Calificación para su correspondiente trámite.  
Si se selecciona NO procede, el turno avanza a certificados asociados.

Para un Recurso de reposición de subsidio de apelación se ingresa el recurso correspondiente y sus datos, ver imagen.

**Interposición de Recursos PROCESO DE DEVOLUCIONES**

Procede Recursos Devolución 

**Agregar Información Recursos**  

Tipo de Recurso	RECURSO DE REPOSICIÓN DE SUBSIDIO DE APELACIÓN 
Usuario Recurso	PRUEBAS 
Observación Recurso	PRUEBAS DE RECURSO DE REPOSICIÓN DE SUBSIDIO DE APELACIÓN  
Fecha de Recurso	29/06/2022 

**Agregar Recurso** 

## *Recurso de subsidio de apelación*

Una vez se agreguen los datos del recurso, este se lista en el historial de recursos del turno, permitiendo a su vez eliminarlo o editararlo mediante los botones de editar o eliminar, (al editar el recurso se debe dar clic en el botón guardar de lo contrario el sistema alertara que tiene algún recurso en edición y no permite continuar).

Al agregar el recurso y este permanezca en esta fase el sistema alertara el tiempo de 60 días (2 meses calendario) para resolver el recurso, ver imagen.

RECURSO DE REPOSICIÓN DE SUBSIDIO DE APELACIÓN  
29/06/2022  
PRUEBAS  
PRUEBAS DE RECURSO DE REPOSICIÓN DE SUBSIDIO DE APELACIÓN

Guarda →

*Recurso de reposición de subsidio de apelación.*

Una vez se ingresa la resolución, el sistema informa mediante un mensaje que el turno se encuentra en segunda instancia, LA OFICINA DE REGISTRO RECIBIRA LA INFORMACION QUE RESOLVIO EL RECURSO, SI es favorable o NO, se debe elegir la opción correspondiente, ver imagen.

El turno se encuentra en segunda instancia

Recurso Favorable:

Favorable      No Favorable

*Segunda instancia*

Se debe ingresar la fecha ejecutoria la cual debe ser mayor a la fecha de la resolución, de lo contrario el sistema mostrara una alerta, ver imagen.

Error en la Validación de Datos del Formulario

Los Siguientes Errores han sido encontrados:  
La fecha ejecutoria debe ser mayor a la fecha de la última resolución

Volver

*Alerta Fecha ejecutoria.*

NOTA: Para avanzar el turno el sistema solicitará una nota informativa obligatoria de tipo Recurso de reposición con subsidio de apelación, ver imagen.

Error en la Validación de Datos del Formulario

Los Siguientes Errores han sido encontrados:  
Debe ingresar una nota informativa de tipo RECURSO DE REPOSICIÓN CON SUBSIDIO DE APELACIÓN

Volver

*Nota informativa recurso de reposición con subsidio de apelación.*

Se ingresa la nota informativa correspondiente para poder avanzar el turno, ver imagen.

### *Nota informativa*

Finalmente, si el recurso SI es favorable el turno se avanzará de nuevo a la fase de calificación para continuar con su debido proceso, si el recurso NO es favorable el turno se avanza a la fase de certificados asociados.

Para un recurso de tipo No procede se ingresa el recurso correspondiente y sus datos, ver imagen.

### *Recurso no procede*

Una vez se agreguen los datos del recurso, este se lista en el historial de recursos del turno, permitiendo a su vez eliminarlo o editarlo mediante los botones de editar o eliminar, (al editar el recurso se debe dar clic en el botón guardar de lo contrario el sistema alertara que tiene algún recurso en edición y no permite continuar), adicional el sistema mostrara la opción de proceder recurso en NO.

Al agregar el recurso y este permanezca en esta fase el sistema alertara el tiempo de 60 días (2 meses calendario) para resolver el recurso, ver imagen.

### *Recurso no procede.*

El sistema solicitará agregar La resolución y fecha ejecutoria, de manera obligatoria, para avanzar el turno, ver imagen.

### *Resolución y fecha ejecutoria.*

NOTA: La fecha ejecutoria debe ser mayor a la fecha de la resolución, de lo contrario el sistema mostrara una alerta,

ver imagen.



*Alerta fecha ejecutoria*

NOTA: Para avanzar el turno el sistema solicitará una nota informativa obligatoria de tipo No procede recurso, ver imagen.



*Nota informativa recurso de reposición.*

Se ingresa la nota informativa correspondiente para poder avanzar el turno, ver imagen.

*Ingreso nota informativa*

Finalmente, al ingresar los datos del recurso, la resolución, fecha ejecutoria, nota informativa se avanza el turno a la fase de certificados asociados.

NOTA: Todas las alertas de tiempos son solo informativas, no bloquearán ni restringirán ninguna funcionalidad del sistema.

## PANTALLAS ADMINISTRATIVAS

Por la opción pantallas administrativas al momento de consultar el turno, si este tiene resolución devuelto y fueron ingresadas las notificaciones estas se listan en el campo historial notificaciones, ver imagen.

**Detalles de Turno**

Datos Basicos de Turno	
Número del Turno	2022-999-6-102
Fecha de Radicación	31/05/2022
Número de Recibo	1234595245
Valor Pagado	22,200
Valor por Pagar	0
Valor Devuelto	0
Valor por Devolver	0
Estado Actual	TURNO ENTREGADO.
Resolución	DEVUELTO
Relación Actual	2022-999-153
Fase Relación Actual	REG_ENTREGA
Origen	SIR

**Turnos de Devoluciones Asociados**

**Matrículas Asociadas**

Matrículas Agregadas Rol Liquidador	Matrículas Agregadas Rol Confrontador	Matrículas Agregadas Rol Confrontador - Correctiva
999-5031		

Cantidad de Matrículas a Agregar:	1
Matrículas Agregadas Rol Liquidador:	1
Matrículas Agregadas Rol Confrontador:	
Matrículas Agregadas Rol Confrontador Correctiva:	
Matrículas Eliminadas Rol Confrontador:	
Matrículas Eliminadas Rol Confrontador Correctiva:	

**Historial de Notificaciones**

Fecha de Notificación: 28/05/2022	Destinatario: USUARIO	Correo Electrónico: PEPITO@GMAIL.COM
Tipo de Notificación: CORREO_ELECTRONICO	Apoderado: SI	
Nombre: PEPITO	Apellidos: PEREZ PEREZ	
Tipo de Documento: CC	Dirección: DSASFCVXC	Número de Documento: 54234535
Teléfono: 564582		

Fecha de Notificación: 01/06/2022	Destinatario: USUARIO	Correo Electrónico: PEDROJUAN@GMAIL.COM
Tipo de Notificación: CORREO_ELECTRONICO	Apoderado: SI	
Nombre: PEDRO	Apellidos: PEREZ PEREZ	
Tipo de Documento: CC	Dirección: FSDASFDFASDF	Número de Documento: 142424
Teléfono: 5142524544		

*Detalles de turno*

## REANOTAR TURNO

Para el proceso de reanotar un turno, este se realiza por las pantallas administrativas/ reanotar turno, ver imagen.

**Administración**

- MANEJO DE TURNOS
  - Ver Detalles de Turnos
  - Manual Turnos Registro
  - Manual Turnos Certificados
  - Traslados de Turnos
  - Reanotar Turnos (se ha resaltado)
  - Administración de Relaciones
  - Anular Turnos
  - Resignación de Turnos
  - Habilitar Turnos

*Reanotar turno*

Se debe ingresar el turno que se desea reanotar, este solo se podrá reanotar cuando se encuentre en la fase notificar nota devolutiva, ver imagen.

**Pantalla Administrativa** Reanotar Turnos

Consulta del Turno a Reanotar

Identificador del Turno

ESTA OPCIÓN ES PARA ENMENDAR ERRORES DEL TIPO DE CALIFICACIÓN DEL DOCUMENTO (Devuelto al Público)

Número de Turno: 2022-999-6-47

Cuando Termine de Reanotar los Turnos haga click en Volver.

Iniciar Reanotación Turno

*Reanotar turno*

Se ingresa la nota informativa correspondiente y se da clic en reanotar turno, este será enviado a la fase de calificación para su correspondiente proceso, ver imagen.

*Reanotar turno*

Si el turno ya fue notificado y se encuentra en la fase de nota devolutiva notificada, Recursos nota devolutiva o certificados asociados, el sistema no permitirá reanotar el turno y mostrará la alerta correspondiente, ver imagen.

*Alerta reanotar*

## TURNOS REL

Para los turnos de origen REL con notificación DEVUELTO, se debe tener en cuenta los siguientes aspectos:

-Al avanzar el turno de la fase de FIRMA REGISTRO, este llegará a la fase notificar nota devolutiva, pero al ser de origen REL, NO se visualizará en la bandeja del usuario SIR, ver imagen.

*Notificar turno REL*

REL será el encargado de cargar la notificación y posteriormente enviar el turno a la fase de NOTA DEVOLUTIVA NOTIFICADA. Al llegar el turno a la fase de NOTA DEVOLUTIVA NOTIFICADA, este se visualizará en SIR, ver imagen.

NOTIFICADO TURNO REL

Una vez se seleccione el turno de origen REL, y no se interponga recurso el turno será avanzado a la fase de certificados asociados, donde por ser origen REL no se visualizará en la bandeja del usuario de SIR, ver imagen.

No interpone recurso

NOTA: REL será el encargado de finalizar el turno cuando se encuentre en la fase de certificados asociados.

Si el turno SI interpone recurso este será avanzado a la fase de RECURSOS NOTA DEVOLUTIVA, y seguirá siendo visible por SIR, donde se debe agregar el recurso correspondiente, ver imagen.

Recurso nota devolutiva REL

Al agregar el recurso si se determina que no procede o no es favorable, esto de acuerdo al tipo de recurso agregado, el turno será avanzado a la fase de certificados asociados, ver imagen.

Recurso de reposición NO procede

Recurso de reposición de subsidio de apelación NO favorable

*Recurso NO procede*

Al avanzar a certificados asociados y por ser un turno REL, este ya no se visualizará en SIR, y REL será el encargado de avanzar y finalizar el turno.



**DESPLIEGUE 8.9**

**04/03/2022**



**SUPERINTENDENCIA  
DE NOTARIADO  
& REGISTRO**  
La guarda de la fe pública

**AJUSTE TARIFAS REGISTRALES 2022 - UVT**

El Sistema de Información Registral SIR debe ajustarse teniendo en cuenta las nuevas tarifas registrales estipuladas en la resolución No.02170 del 28 de Febrero de 2022 para todos los procesos que impliquen un cobro en el sistema.

Actualmente el Sistema de Información Registral (SIR), dentro del proceso de su proceso de radicación utiliza como base el salario mínimo por medio de la tabla SIR\_OP\_TARIFA, validando la cuantía que sea ingresada dentro de un rango de valores utilizado en la tabla SIR\_OP\_TARIFA\_REGISTRAL, de esta manera se genera el valor de los derechos registrales.

Se realiza un ajuste en el cálculo del valor de derechos registrales basados en UVT de acuerdo con la siguiente tabla;

<b>RANGOS UVT *</b>	<b>INICIO</b>	<b>FINAL</b>	<b>TARIFA 2022</b>
< = 250,23 UVT	-	< = \$ 9.509.741	\$39.000
> 250,23 UVT; < = 3.753,41 UVT	> \$9.509.741	< = \$142.644.594	6,68 X MIL
> 3.753,41 UVT; < = 6.505,91 UVT	> \$142.644.594	< = \$247.250.604	8,29 X MIL
> 6.505,91 UVT; < = 9.633,76 UVT	> \$247.250.604	< = \$366.121.415	9,23 X MIL
>9.633,76 UVT	> \$366.121.415		9,76 X MIL

\*El valor de la UVT para el año 2022 es de \$ 38.004, el cual se mantendrá vigente hasta tanto se expida la resolución que actualice las tarifas para la prestación del servicio público registral establecidas.

Se modifica el proceso de radicación en su generación del valor de derechos registrales, por medio de una nueva tabla de valores UVT, se crea una nueva tabla en la base de datos con la información de los rangos en la aplicación de la tarifa de acuerdo con el valor de UVT generado en el cálculo de transformación de pesos a UVT y así determinar el valor a cobrar por derechos de registro.

Se crea una tabla nueva llamada SIR\_OP\_TARIFA\_REGISTRAL\_UVT, donde se almacenan los rangos y tarifa a aplicar de acuerdo con los valores calculados para los derechos registrales, adicional se crea un registro en la tabla SIR\_OP\_TARIFA, donde se almacenará el valor del UVT actual: 38.004, el cual se podrá versionar anualmente, de acuerdo con el valor establecido.

**SIR\_OP\_TARIFA**

ID_TIPO	ID_VERSION	ID_CODIGO	TRFS_VALOR	TRFS_FECHA_INICIO	TRFS_FECHA_CREACION
1	CONFIGURACION	1	VALOR_UVT	38004	01/01/22
2	CONFIGURACION	17	SALARIO_MINIMO	908526	01/01/21
3	CONFIGURACION	16	SALARIO_MINIMO	877803	01/01/20
4	CONFIGURACION	15	SALARIO_MINIMO	820115	01/01/19

## SIP\_OP\_TARIFA\_REGISTRAL\_UVT

ID_TARIFA	INICIO CUANTIA	INICIO_INCLUSIVE	FINAL CUANTIA	FINAL_INCLUSIVE	TIPO_APPLICACION_TARIFA	VALOR_TARIFA
1	1	0	250,23	1 a		36900
2	2	250,23	0	3753,41	1 p	6,32
3	3	3753,41	0	6505,91	1 p	7,85
4	4	6505,91	0	9633,76	1 p	8,74
5	5	9633,76	0	(null)	1 p	9,24

El sistema realiza los cálculos necesarios para la generación del valor de derechos registrales de la siguiente forma:  
Se ingresa un valor de 700.000.000 como cuantía, el sistema inicialmente valida en que rango de valores UVT se encuentra realizando el siguiente proceso:

Valor cuantía / Valor UVT

$$700.000.000 / 38.004$$

$$= 18,419.11$$

Con el valor resultante se valida el rango en que se encuentra el valor 18,419.11, de acuerdo a la tabla.

RANGOS UVT *	INICIO	FINAL	TARIFA 2022
< = 250,23 UVT	-	< = \$ 9.509.741	\$39.000
> 250,23 UVT; < = 3.753,41 UVT	> \$9.509.741	< = \$142.644.594	6,68 X MIL
> 3.753,41 UVT; < = 6.505,91 UVT	> \$142.644.594	< = \$247.250.604	8,29 X MIL
> 6.505,91 UVT; < = 9.633,76 UVT	> \$247.250.604	< = \$366.121.415	9,23 X MIL
>9.633,76 UVT	> \$366.121.415		9,76 X MIL

El sistema debe calcular el valor realizando la aplicación del rango correspondiente de acuerdo con la tabla:

$$(\text{Valor cuantía} / 1000) * \text{Tarifa UVT} \quad \text{o} \quad (\text{Valor cuantía} * \text{Tarifa UVT}) / 1000$$

$$(700.000.000 / 1000) * 9.76 \quad (700000000 * 9.76) / 1000$$

$$700.000 * 9.76 \quad 6.832.000.000 / 1000$$

$$6.832.000 \quad 6.832.000$$

El sistema genera el valor de derechos registrales luego de hacer los cálculos necesarios, de acuerdo con el ejemplo el valor de derechos registrales es 6.832.000 pesos.

**Nota:** El sistema mantendrá los cálculos de redondeo para el valor de los derechos registrales , se redondeará a la centena más cercana.

Es importante tener en cuenta que los valores de la conservación documental no se alteran con la aplicación de los cobros de derechos con UVT.



## DESPLIEGUE 8.8

20/01/2022



### CONSULTA NACIONAL

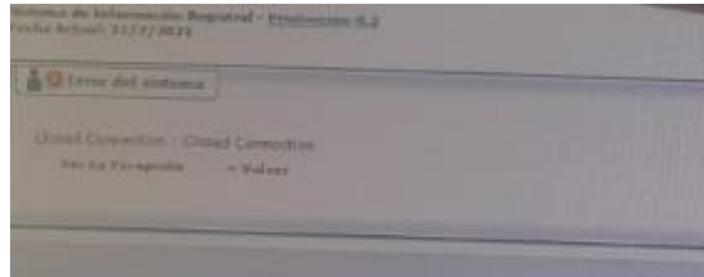
Actualmente en el Sistema SIR por pantallas administrativas/ administración de perfiles, permite agregar el Rol Consulta nacional desde el usuario REGISTRADOR de todas las ORIP, se requiere limitar esta asignación del rol CONSULTA NACIONAL, solo al usuario Administrador Nacional, quien será el único que posea privilegios para asignar el rol Consulta Nacional a algún usuario, además de ser requerida esta asignación debe realizarse con previa autorización de la Dirección Técnica de Registro.

1. Es necesario que SIR cuente con una validación al momento de realizar la asignación de roles, por medio de la pantalla administrativa/ “Administración de perfiles”, donde solo se le permita realizar la asignación del Rol CONSULTA NACIONAL, al usuario Administrador Nacional, el cual podrá realizar esta asignación con previa autorización de la Dirección Técnica de Registro. Se debe modificar la siguiente pantalla y validar por código para que se visualice únicamente la opción de agregar “Rol Consulta Nacional” cuando se acceda desde el Administrador Nacional, ver imagen 1.

2. Es necesario que SIR permita crear un usuario sin necesidad de agregarle Roles, los cuales pueden ser agregados posteriormente por el Registrador, para esto se debe deshabilitar en la pantalla Crear Usuario y validar por código la vista; “roles para el usuario”, ver imagen 2



3. Evaluar y corregir el error que ocurre al momento de agregar o quitar roles el sistema, ya que presenta el mensaje: "error: Closed Connection", ver imagen 3.



4. Evaluar casos donde se inactivan roles y los turnos siguen pasando a estos roles aun cuando se encuentran inactivos.
5. Permitir eliminar roles de un usuario sin que se requiera que otro usuario de la Orip a la que pertenece tenga asignados estos roles.
6. Se debe verificar que el sistema no permite eliminar roles cuando tiene turnos asignados o estaciones cuando están asignadas a otro usuario, ya que se han presentado casos en que no se cumple esta validación.
7. Se requiere en la pantalla administrativa, administrador de perfiles, en el historial de novedades, adicionar el campo de rol novedad, de manera que se identifique el rol que fue creado, modificado o eliminado. Así mismo, revisar el funcionamiento del filtro existente de rango de fecha y el paginado de la parte inferior para reducir la presentación de los registros de dicho historial. Ver imagen 4.

Resultados de la Consulta					
Listado					
Rango Fecha, desde:	Hasta:		Buscar		
<a href="#">Descargar En Excel</a>					
Nombre de Novedad	Rol Novedad	Fecha	Usuario	Tipo Nota	Descripción
ACTIVAR/INACTIVAR ROL	CAJIFICADOR	23/12/2020	USUARIO.PRUEBAS	Vacaciones	SE INACTIVA ROL POR ENCONTRARSE EN VACACIONES
ADICIONAR ROL	CAJERO REGISTRO	17/11/2020	USUARIO.PRUEBAS	Otros	BUSQUEDA DE TURNO 465
ELIMINAR ROL	DECISOR	15/11/2020	USUARIO.PRUEBAS	Otros	
ELIMINAR ROL	DECISOR	15/11/2020	USUARIO.PRUEBAS	Otros	HK.HV.GCHMG.GH.DHZEDHBEDHDMHFGJMNFGMFMGFG
ADICIONAR ROL	DIGITADOR	11/11/2020	USUARIO.PRUEBAS	Otros	VBVB.VLNUFCOHNL.LNLG_NCLV.HNZSFSGSXDRDHSBDGB85XDGFHxD
ADICIONAR ROL	DIGITADOR	11/11/2020	USUARIO.PRUEBAS	Otros	FGCHMCGMHGCH_ÑÑCHB_-GIMGHMGMGHMGHG_HRFXMFHMFM
ELIMINAR ROL	DIGITADOR	25/11/2019	USUARIO.PRUEBAS	Vacaciones	SE ELIMINA POR PRUEBAS PRUEBAS PRUEBAS
ADICIONAR ROL	DIGITADOR	25/11/2019	USUARIO.PRUEBAS	Vacaciones	SE CREA CASO POR PRUEBAS DE CAPACITACION
ELIMINAR ROL	ENTREGA	22/11/2019	USUARIO.PRUEBAS	Vacaciones	TERMINA REEMPLAZO DE VACACIONES ...
ADICIONAR ROL	CAJIFICADOR	22/11/2019	USUARIO.PRUEBAS	Vacaciones	REEMPLAZO DE VACACIONES (...)
ADICIONAR ROL	CAJERO REGISTRO	25/09/2019	USUARIO.PRUEBAS4	Otros	SE NOMBRÁ COMO USUARIO DE PRUEBAS PARA CREAR DOCUMENTO DE ROLES VS PANTALLAS

8. Adicionalmente se debe modificar la pantalla administrativa, crear usuarios, de forma que se incluyan los campos: tipo de identificación y número, y que sean obligatorios para la creación de usuarios. ver imagen 5

9. Se requiere inactivación del botón de Crear ubicado actualmente en la pantalla administrativa, Administración de usuarios, rol Administrador regional y dejarlo habilitado únicamente al administrador nacional, quien es el encargado de creación de usuario y vinculación a la orip. ver imagen 6.



**DESPLIEGUE 8.7**  
**27/11/2021**



### DE VOLUCION DE DINERO

Actualmente el Sistema de Información Registral (SIR), dentro del proceso de devoluciones de dinero, suma el valor de la conservación documental al total del valor que se debe devolver al usuario por el no registro del documento.

Es necesario modificar las pantallas del Rol usuario administrativo en la fase de Devolución, sub-fases Análisis, Recursos Devolución y Resolución:

1. En el Rol Usuario Administrativo, en la fase “Devolución”, Sub-fase “Análisis”, al ingresar un turno para realizar una devolución de dinero y se carguen los valores totales se debe modificar la pantalla del menú “valores totales”, de manera, que no sea visualizado, ni incluido en el valor total a devolver, el valor que corresponda a conservación documental. Ver imagen 1.

Imagen 1

Valor	Detalles
\$ 2,250,000.00	Valor a devolver por derechos
\$ 0.00	Valor a devolver por derechos Mayor Valor
\$ 0.00	Saldo a Favor
\$ 32,600.00	Valor a devolver por certificados asociados
\$ 0.00	Valor a devolver por certificados Mayor Valor
\$ 0.00	Valor a devolver por impuestos
\$ 0.00	Valor a devolver por impuestos Mayor Valor

2. En el Rol Usuario Administrativo, en la fase “Devolución”, Sub-fase “Resolución”, al avanzar un turno desde la sub-fase Análisis y se carguen los valores a devolver se debe modificar la pantalla del menú “Valores a Devolver”, de manera, que no sea visualizado, ni incluido en el valor total a devolver, el valor que corresponda a conservación documental. Ver imagen 2

Imagen 2

Valores a devolver	
Valor a devolver por derechos	\$ 20,300.00
Valor a devolver por derechos Mayor Valor	\$ 00
Saldo a Favor	\$ 00
Valor a devolver por certificados asociados	\$ 00
Valor a devolver por certificados asociados Mayor Valor	\$ 00
Valor a devolver por Impuestos	\$ 00
Valor a devolver por Impuestos Mayor Valor	\$ 00

Aceptar Devolver a Análisis Cerrar

3. En el Rol Usuario Administrativo, en la fase “Devolución”, Sub-fase “Recursos Devolución.”, al avanzar un turno desde la sub-fase “Resolución” y se carguen los valores a devolver se debe modificar la pantalla del menú “Valores a Devolver”, de manera, que no sea visualizado, ni incluido en el valor total a devolver, el valor que corresponda a conservación documental. Ver imagen 3

Imagen 3

Valores a devolver	
Valor a devolver por derechos	\$ 30,400.00
Valor a devolver por derechos Mayor Valor	\$ 00
Saldo a Favor	\$ 00
Valor a devolver por certificados asociados	\$ 16,800.00
Valor a devolver por certificados asociados Mayor Valor	\$ 00
Valor a devolver por Impuestos	\$ 00
Valor a devolver por Impuestos Mayor Valor	\$ 00

Aceptar Devolver a Resolución Cerrar

Se debe ajustar la pantalla del rol Pago devolución, cuando se carguen los valores a devolver se debe modificar la pantalla del menú “Valores a Devolver”, de manera, que no sea visualizado, ni incluido en el valor total a devolver, el valor que corresponda a conservación documental. Ver imagen 4-

Imagen 4

Valores a devolver	
Valor a devolver por derechos	\$ 36,900.00
Valor a devolver por derechos Mayor Valor	\$ 00
Valor a devolver por certificados asociados	\$ 00
Valor a devolver por certificados asociados Mayor Valor	\$ 00
Valor a devolver por impuestos	\$ 00

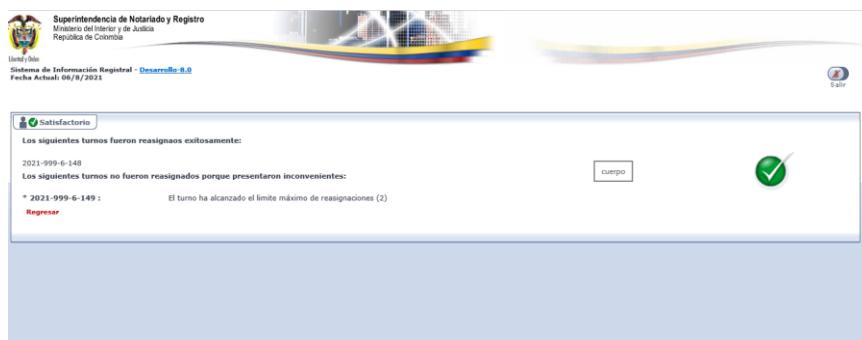
Aceptar Devolver a Resolución Cerrar

### REASIGNACION TURNOS POR FASES

Actualmente en el Sistema al Realizar la reasignación de un turno, limita esta a solo dos (2) reasignaciones por turno, sin importar la fase donde se encuentre adicionándole una nota informativa obligatoria para realizar la reasignación.

Es necesario que SIR permita la reasignación de turnos por **fases** con una nota informativa al momento de realizar reasignación turnos, donde se pueda estipular el motivo de la reasignación, además se debe mantener el número máximo de reasignaciones (2), y los roles en la oficina que pueden ejecutar esta acción (Administrador Regional, Registrador y Coordinador jurídico).

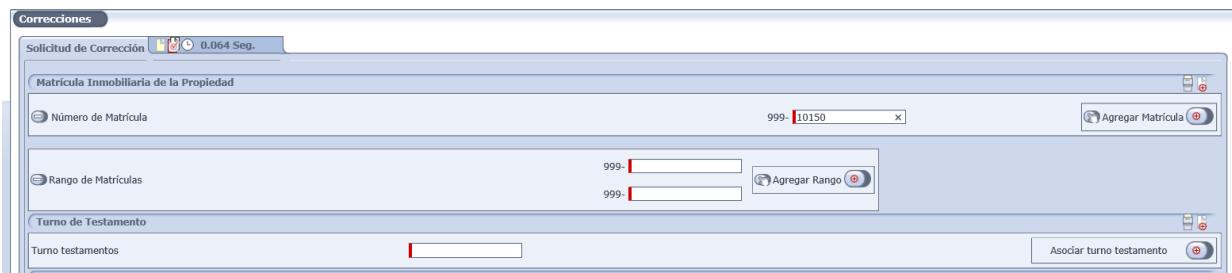
**“El turno ha alcanzado el límite máximo de reasignaciones (2)”**



### CORRECCION BUGS

Actualmente en el aplicativo SIR se permite ingresar folios de círculos diferentes en el rol Cajero correcciones y Responsable correcciones, por lo tanto se ajustan las pantallas para estos dos roles limitando el ingreso de las matrículas únicamente al círculo en el cual se encuentre logueado, ver imagen 1

Imagen 1



The screenshot shows a "Correcciones" (Corrections) screen in the SIR application. It displays a "Solicitud de Corrección" (Correction Request) with a timestamp of "0.064 Seg.". Below this, there are three input fields: "Número de Matrícula" (Matrícula number) containing "999-10150", "Rango de Matrículas" (Matrícula range) with two entries "999-", and "Turno de Testamento" (Testament appointment) with an empty input field. To the right of these fields are buttons for "Agregar Matrícula" (Add Matrícula), "Agregar Rango" (Add Range), and "Asociar turno testamento" (Associate testament appointment). The background shows other parts of the application interface.

Actualmente el aplicativo SIR permite visualizar turnos avanzados por relación de REL, se ajustan las validaciones para impedir que por SIR se visualicen los turnos de REL avanzados por relación, ver imagen 2.

Imagen 2



**DESPLIEGUE 8.4**  
**11/08/2021**



#### CONTROL MATRICULAS CONFRONTADOR

Actualmente en el aplicativo SIR no cuenta con la validación de la cantidad de matrículas agregar desde la fase de liquidador, ni tampoco controla el número de matrículas que puede adicionar y/o eliminar en el rol confrontador en sus dos fases “confrontación y confrontación correcctiva”, lo cual no permite determinar si lo realizado desde el rol liquidador esta correcto y mucho menos si lo adicionado y/o eliminado del turno en el rol confrontador es válido.

SIR cuenta con nuevos campos que permiten capturar el número de matrículas a agregar, este valor es almacenado en base de datos para llevar un control sobre el número de matrículas asociadas en un turno, tanto en la fase de liquidador como la fase de confrontador. Además, el sistema alerta y restrinje al funcionario cuando este no digite dicho campo. Para los casos que se selecciona tipo de solicitud antiguo sistema, testamento no es obligatorio agregar cantidad de matrícula. De otra parte cuando se selecciona tipo normal pero no se agrega matricula como en el caso de los documentos con Actos de Baldío y Pertenencia que pasan a calificación para apertura de matrícula y carecen de tradición, no se obliga agregar cantidad de matrículas.

El sistema alerta al liquidador si se requiere agregar más matriculas en la fase confrontador, debe agregar nota informativa, esto ayuda advertir cuando se trata de un documento con gran cantidad de matrículas y no es posible para el liquidador agregarlas.

En el rol confrontador el sistema debe solicitar al funcionario la selección de una nota informativa donde estipule el motivo de adicionar o eliminar alguna matricula.

**Nota** de no agregar y/o eliminar alguna matricula el sistema realizar el avance normalmente como lo hace hasta el momento.

Se modifica la pantalla de **liquidador** adicionando un campo de captura nombrado “**Cantidad de Matrículas Agregar**”, Se visualiza en imagen 1

## Imagen 1

Nuevas entradas

Número del Turno Anterior

Número de Liquidación L-2021-999-6-

Asociar una Matrícula

Cantidad de Matrículas Agregar  Número de Matrícula 999- Agregar Matrícula

Datos Opcionales para Antiguo Sistema

Haga click para modificar una liquidación existente

Si en el caso que en el campo de captura no se digita ningún valor el sistema debe alertar y restringir que agregue una matrícula. Se visualiza en imagen 2

## Imagen 2

Superintendencia de Notariado y Registro  
Ministerio del Interior y de Justicia  
República de Colombia

Sistema de Información  
Registral - Desarrollo-8.0  
Fecha Actual: 14/7/2021

Alerta: El administrador de impresión no está funcionando

Error en la Validación de Datos del Formulario

Los Siguientes Errores han sido encontrados en el Formulario:  
Debe diligenciar el campo - Cantidad de Matrículas Agregar

Solicitud de Registro 0.109 Seg.

Haga click para modificar una liquidación existente

Haga click para agregar certificados asociados a un turno ya existente

## Rol Confrontador

Se desarrolla y/o modifica en la pantalla del rol confrontador lo siguiente:

- Si el funcionario confrontador elimina una o varias matrículas asociadas en el turno por el rol liquidador, el sistema deberá contar la cantidad de matrículas eliminadas y almacenar tanto el conteo como las matrículas desasociadas en la base de datos. Ejemplo el numero 2 si se eliminó 2 matrículas y guardar el número de matrícula 999-001 y 999-020, para posteriormente mostrarlas en ver detalles de turno.

Vista a modificar con las validaciones anteriormente mencionada. Se visualiza en imagen 3

## Imagen 3

- Es necesario que el sistema solicite al funcionario agregar una nota informativa de carácter obligatorio cuando se intente agregar una o varias matriculas “**Nota Informativa agregar matrícula**”. Se visualiza en imagen 4

Imagen 4

## Rol Calificador

El calificador con el botón Desasociar selecciona las matriculas que necesita desasociar del turno, en esta acción el sistema debe obligar a incluir nota informativa de la razón de eliminar las matriculas, el sistema debe guardar la información de las matriculas eliminadas y reflejarse en la consulta de detalles del turno en el historial de matriculas .

## Rol confrontador – Confrontación correctiva

Es necesario desarrollar y/o modificar en la pantalla de la fase confrontación-correctiva lo siguiente:

- Si el funcionario confrontador elimina una o varias matriculas asociadas en el turno por el rol liquidador, el sistema deberá contar la cantidad de matrículas eliminadas y almacenar tanto el conteo como las matriculas desasociadas en la base de datos Ejemplo el numero 2 si se eliminó 2 matrículas

y guardar el número de matrícula 999-001 y 999-020, para posteriormente mostrarlas en ver detalles de turno.

Vista a modificar con las validaciones anteriormente mencionada. Se visualiza en imagen 5

The screenshot shows a web-based application for document registration. At the top, there's a header with the SNR logo and some administrative information. Below it, the main form is titled 'Registro de Documentos'. It contains several sections: 'Confrontación' (with a dropdown menu), 'Encabezado del Documento' (with fields for 'Tipo' set to 'CERTIFICADO', 'Número' set to '2344534', and 'Fecha' set to '15/01/2021'), 'Oficina de Procedencia' (with 'Departamento' set to 'CAQUETA' and 'Municipio' set to 'FLORENCIA'), and 'Asociar una Matrícula' (with a dropdown menu showing '999-9875'). There are also sections for 'Número de Matrícula' and 'Rango de Matrículas' (each with three input fields). A large red box highlights the 'Número' field in the 'Asociar una Matrícula' section. At the bottom, there are buttons for 'Agregar Matrícula' and 'Agregar Rango'.

- En el caso que el funcionario intente agregar una o varias matriculas el sistema debe solicitar al funcionario agregar una nota informativa de carácter obligatorio, “Nota Informativa agregar matricula”. Se visualiza en imagen 5.

Imagen 5

This screenshot shows the 'Agregar Nota Informativa' (Add Informational Note) screen. It has a title bar 'Agregar Nota Informativa' and a 'Notas Informativas' tab. Below that, there's a 'Nota' section with a 'Escoger Tipo de Nota' dropdown containing 'XXXX - NOTA INFORMATIVA - AGREGAR MATRICULA'. To the right, there's a scrollable area with the heading 'NOTA INFORMATIVA AGREGAR MATRICULA'. At the bottom, there are buttons for 'Agregar', 'Regresar', 'Cancelar', and a close button.

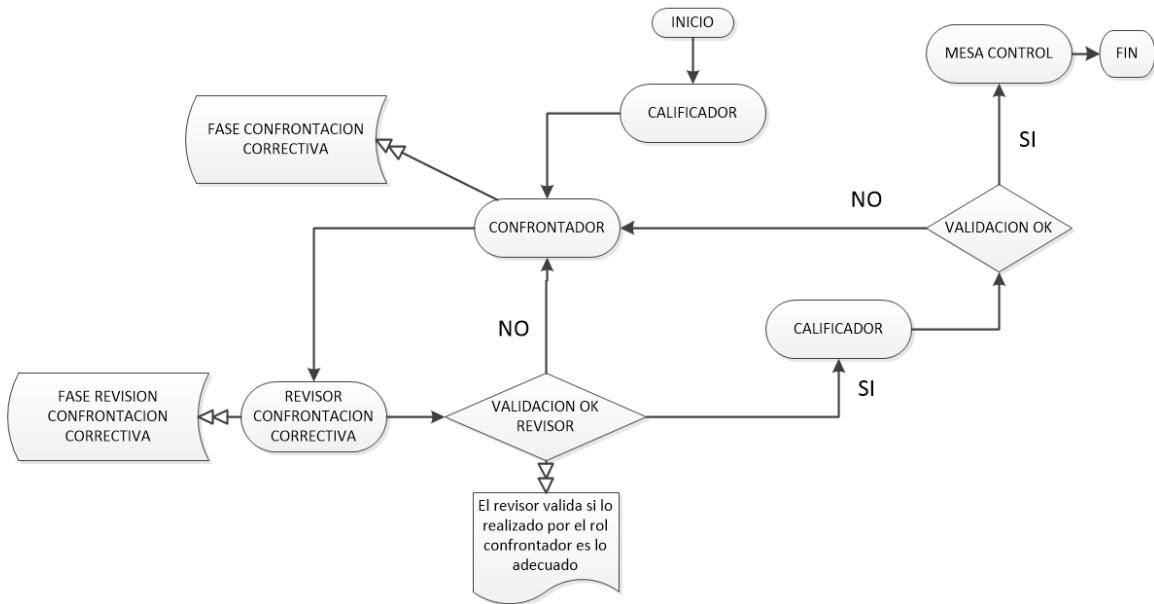
- En el caso que el funcionario intente eliminar una o varias matriculas el sistema debe solicitar al funcionario agregar una nota informativa de carácter obligatorio, “Nota Informativa Eliminar matricula”. Se visualiza en imagen 6.

Imagen 6

This screenshot shows the same 'Agregar Nota Informativa' screen as the previous one, but with a different selection in the 'Escoger Tipo de Nota' dropdown. It now displays '13051 - NOTA INFORMATIVA ELIMINAR MATRICULA'. The rest of the interface is identical to Image 5, including the scrollable area on the right and the bottom navigation buttons.

- Es necesario que una vez el rol confrontador en la fase confrontación correctiva realicé la acción de eliminar y/o agregar alguna matrícula este envíe el turno al rol revisor en la fase “REVISIÓN CONFRONTACIÓN CORRECTIVA”.

### Esquema



## Rol Revisor Confrontación correctiva

Es necesario adicionar en el menú un nuevo rol llamado “Revisor Confrontación Correctiva” que al dar clic abra una nueva pantalla con una opción o fase nombrada “Revisión Confrontación Correctiva”, en esta opción se debe listar los turnos que se han enviado a este desde el rol confrontador fase confrontación correctiva. Se visualiza en imagen 7, 8 y 9.

Imagen 7

Superintendencia de Notariado y Registro  
Ministerio del Interior y de Justicia  
República de Colombia

Liberad y Orden

Sistema de Información Registral - Producción-8.2  
Fecha Actual: 19/7/2021

**ROLES**

<input checked="" type="radio"/> Seleccione un Rol
<input type="radio"/> ABOGADO DIRECCION REGISTRO
<input type="radio"/> ACTUACIONES ADMINISTRATIVAS
<input type="radio"/> AMPLIACION TRADICION
<input type="radio"/> ANALISTA REPARTO NOTARIAL
<input type="radio"/> AUXILIAR ACTUACIONES ADMINISTRATIVAS
<input type="radio"/> AUXILIAR DE CORRECCIONES
<input type="radio"/> BUSCADOR ANTIGUO SISTEMA
<input type="radio"/> CAJERO
<input type="radio"/> CAJERO CERTIFICADOS
<input type="radio"/> CAJERO CERTIFICADOS MASIVOS
<input type="radio"/> CAJERO CORRECCIONES
<input type="radio"/> RECEPTOR DEVOLUCIONES
<input type="radio"/> RECEPTOR FOTOCOPIAS
<input checked="" type="radio"/> REVISOR CONFRONTACION CORRECTIVA
<input type="radio"/> CAJERO REGISTRO
<input type="radio"/> RECEPTOR REPARTO NOTARIAL
<input type="radio"/> CALIFICADOR

Condiciones Generales de Uso | Política de Privacidad

... Nota ...  
Espere 5 segundos a que se muestre el diálogo de adv

Iniciar Aplicativo de Impresión

Imagen 8



Imagen 9



*Nota: este rol solo se puede asignar en la oficina al funcionario registrador o coordinador jurídico, de lo contrario el sistema debe alertar que no se puede realizar la asignación.*

- **Nueva vista Rol Revisor Confrontación Correctiva**

Es necesario realizar una nueva pantalla donde el funcionario valide lo enviado por el rol confrontador en la fase revisan confrontación correctiva, la nueva pantalla debe tener lo siguiente.

- Historial de matrículas = visualiza las matriculas agregadas y/o eliminadas en alguna de las fases del turno.
- Botón Aceptar = Envía al mismo rol calificador que asigno la acción correctiva
- Botón enviar confrontación = envía turno al mismo confrontador en la fase confrontación correctiva.
- Listar notas informativas = lista las notas informativas del turno
- Botón Notas informativas = Activa la pantalla de agregar notas informativas.

Se visualiza en imagen 10

Imagen 10

### Nota informativa rol revisor

Se debe incluir en el sistema dos notas informativas en la nueva fase Rol una llamada “xxx- Nota informativa envío Confrontación correctiva” y la otra “xxx- Nota informativa envío Calificación”. En los dos casos el sistema debe validar que la nota ingresada corresponde al rol donde se va enviar el turno y es de manera obligatoria para realizar el avance.

La nota informativa de revisor a calificación no debe ser obligatoria.

La nota informativa aplica obligatorio de Revisor a Confrontación correctiva,

Se visualiza en imagen 11.

Imagen 11

### Ver detalles de turno

Es necesario que le sistema en la pantalla ver detalles de tueno visualice la cantidad de matrículas agregadas en cada uno de los roles y fases. Se visualiza en imagen 13 y 14.

Imagen 13

Imagen 14

**Matrículas Asociadas**

Número de matrículas Agregar Rol Liquidador	3
Matrículas Agregadas Rol Liquidador	2
Matrículas Agregadas Rol Confrontador	3
Matrículas Eliminadas Rol Confrontador	1
Matrículas Agregadas Rol Confrontador - Correctivo	0
Matrículas Eliminadas Rol Confrontador - Correctivo	0

**INFOTIC**  
SOLUCIONES INTELIGENTES

**DESPLIEGUE 8.3**  
**06/08/2021**

**SNR SUPERINTENDENCIA DE NOTARIADO & REGISTRO**  
La guarda de la fe pública **60 Años**

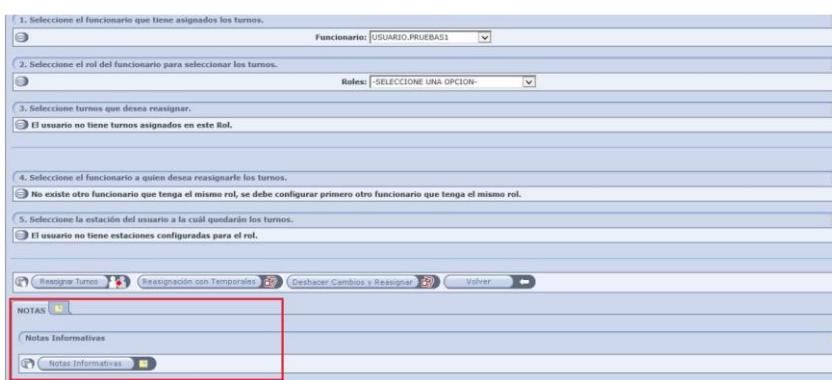
**REASIGNACION DE TURNOS**

Actualmente en el Sistema al Realizar la reasignación de un turno, actualmente el sistema no tiene opción para incluir nota informativa que justifique este hecho, por lo que se hace necesario la inclusión de la misma. Es necesario que SIR cuente con una nota informativa al momento de realizar reasignación turnos, donde se pueda estipular el motivo de la reasignación, además se debe controlar el número máximo reasignaciones (2), y los roles en la oficina que pueden ejecutar esta acción (Administrador Regional, Registrador y Coordinador jurídico).

**Nota informativa**

Se debe incluir en el sistema una nota informativa en la vista reasignacionTurnos.do, esta nota debe ser obligatoria al momento de reasignar el turno, de no seleccionar esta opción el sistema debe alertar al funcionario con un el siguiente mensaje ***No es posible reasignar el turno. Se requiere una nota informativa***.

**“El turno ha alcanzado el límite máximo de reasignaciones (2)”**



### ANULACION DE TURNOS-CERTIFICADOS ESPECIALES

Actualmente en el aplicativo SIR, permite anular turnos de radicación de documentos antes de ser calificados o en la misma fase de calificación, hoy en día las oficinas de registro al realizar la operación de anulación de turnos, el sistema le permite anular el turno de registro, sin dejar una restricción de tiempo para la anulación.

SIR tiene la funcionalidad de anular el turno durante los dos (2) días hábiles siguientes a la radicación del mismo, para lo cual se incluye la validación en la pantalla anulación de turno. consultando la fecha de radicación en las tablas del sistema y en caso que el turno no cumpla con la validación arroja un mensaje informativo indicando que: “El turno supera la fecha límite de dos (2) días hábiles para ser anulado”, y no permita realizar dicha acción

Este desarrollo da cumplimiento a la Instrucción Administrativa 20 de 2020, de acuerdo con la solicitud realizada por el doctor Mauricio Rivera – Director Técnico de Registro del 08 de marzo de 2021.

De otra parte, para dar cumplimiento al numeral 7 de la instrucción administrativa el usuario que tenga el rol registrador no tendrá límite de tiempo de anulación; Ademas, el sistema permite incluir la justificación que supere los 100 caracteres.

Para el reporte 156; Diario anulados del día, se modificó de la siguiente manera;

-Cambio el nombre del reporte en la lista de, Diario anulados del día a **Reporte turnos Anulados por rango de fecha**.

-Cambio el rango de fechas para realizar la consulta de una fecha de inicio a una fecha de fin, no superior a un rango de 3 meses, igualmente el reporte permite generar el reporte por un día.



 Superintendencia de Notariado y Registro  
 Ministerio del Interior y de la Justicia  
 República de Colombia  
[www.snnrnotariado.gov.co](http://www.snnrnotariado.gov.co)

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
 REPORTE TURNOS ANULADOS  
 REALIZADO POR: USUARIO PRUEBAS  
 Impreso el 02 de Junio de 2021 a las 13:30:17 PM  
 Página 1

Proceso	CERTIFICADOS							
Radicación	Fecha Rad. Recibo	Canal de Recaudo	Forma de Pago	Valor	Conservación Documental	Hora	Motivo	Usuario
2021-204-1-42	02/06/2021	20000133 CONSIGNACION_C_PRODUCTO	CUENTA PRODUCTO	16.800	0	1:04 PM	PRUEBA DE ANULACIÓN DE TURNO.	54533

Proceso	REGISTRO							
Radicación	Fecha Rad. Recibo	Canal de Recaudo	Forma de Pago	Valor	Conservación Documental	Hora	Motivo	Usuario
2021-204-6-9	02/06/2021	8 20000133 CONSIGNACION_C_PRODUCTO	CUENTA PRODUCTO	11.458.700	224.700	1:04 PM	PRUEBA DE ANULACIÓN DE TURNO.	54533

Total Anulado: **11.475.500**  
 Total Conservación Documental : **\$ 224.700**

### CERTIFICADOS ESPECIALES

Actualmente en el aplicativo SIR bloquea la generación de certificados cuando estos están en trámite y solo deja generar cuando pasa de la fase desanotado.

SIR genera el certificado inmediato, ya que tiene todos los datos de los folios de matrícula inmobiliaria del país y las firmas digitalizadas de todos los registradores, para que 24/7 expidan los certificados inmediatos que la ciudadanía requiera desde cualquier parte del mundo, sin que los registradores hagan un estudio de este. SIR tiene una opción que permite generar certificados especiales con una nota que identifique el tipo del certificado. Se incorpora en el sistema las validaciones para identificar que el folio esta en las fases estipuladas de bloqueo y que en el imprimible refleje una nota aclarando que dicho folio se encuentra en trámite.

Se agrega en el rol cajero certificado la opción que permita generar certificados especiales, por medio de un botón llamado “Certificados Especiales”.

The screenshot shows a software interface for generating certificates. At the top, there are fields for 'Oficina Origen' (Local or National), 'Tipo de Certificado' (INMEDIATO selected), 'Certificados Especiales' (button highlighted with a red box), 'Tipo de Tarifa' (NORMAL), and 'Cantidad' (1). Below these are fields for 'Asociación Turno Anterior' and 'Número del Turno Anterior'. At the bottom right is a 'Agregar Turno Anterior' button.

Se genera el imprimible del certificado el cual una vez se haya realizado la validación de si la matrícula ingresada esta en actuación administrativa. Para lo anterior se agrega la siguiente nota: **“ESTE CERTIFICADO NO REFLEJA LA SITUACION JURIDICA DEL INMUEBLE TENIENDO EN CUENTA QUE SE ENCUENTRA CON UNA ACTUACION ADMINISTRATIVA Y SU ESTADO JURIDICO PUEDE CAMBIAR”**.



Si la matrícula ingresada se encuentra en trámite en un turno que no permita su expedición normal y que esto implique generarla de manera especial, pero que a demás este sea diferente de una actuación administrativa el imprimible tendrá la siguiente nota: **“ESTE CERTIFICADO NO REFLEJA LA SITUACION JURIDICA DEL INMUEBLE TENIENDO EN CUENTA QUE SE ENCUENTRA EN TRAMITE CON EL TUNO 2021-999-6-1, Y SU ESTADO JURIDICO PUEDE CAMBIAR”**.



## INCLUSIÓN NUEVOS CAMPOS REQUERIMIENTO CATASTRO MULTIPROPÓSITO

Actualmente le Sistema de Información Registral (SIR), no cuenta con campos que permitan incluir Linderos actualizados, área de construcción, área privada, Propiedad Horizontal, Coeficiente de copropiedad, Avalúo, fecha avalúo, modalidad, (VIS, VIPA, VIP, NO VIS), destinación económica, determinación del inmueble (garaje, local, casa, lote, entre otros). Lo anterior, implica realizar desarrollos y modificaciones en todas las pantallas donde se captura la información para crear folios, por el rol calificador y por pantallas administrativas, creación directa de folios y grabación directa de folios.

Se incluye en el sistema los siguientes ajustes:

En el rol CALIFICADOR por medio de la opción de crear folio y por pantallas administrativas en la opción Creacion difica de folio, se agregan los siguientes campos de captura al momento de realizar la creación del folio.

- **Determinación del inmueble:** En este campo se presentará una lista desplegable con las diferentes determinaciones de inmuebles.
- **Modalidad:** tipo de modalidad del inmueble, lista desplegable con los siguientes valores; VIS, VIP, VIPA Y NO VIS, para los códigos 0125 y 0126
- **Área Privada:** Área privada del inmueble en metros y centímetros.
- **Área construida:** Área del inmueble construida en metros y centímetros.
- **Coeficiente:** Porcentaje o coeficiente del inmueble.
- **Área:** Área de terreno total, en hectáreas, metros y centímetros.
- **Linderos:** descripción de los linderos históricos.
- **Linderos Técnicamente definidos:** Descripción de los linderos definidos.
- **Avalúo** – este campo no lo debe diligenciar la ORIP, se diligenciará con información proveniente de los gestores catastrales.
- **Fecha avalúo** se diligenciará con información proveniente de los gestores catastrales
- **Destinación económica:** Esta información se alimentará con información expuesta por el IGAC.

Se visualiza en la imagen 1

Al agregar una anotación en la creación del folio, si su naturaleza jurídica corresponde al código 0125 o 0126, el sistema habilita el campo de Modalidad, mediante la lista desplegable (VIS, VIP, VIPA, NOVIS), se visualiza en la imagen 2

The screenshot shows a software interface for creating a folio. At the top, it says 'CALIFICACIÓN - CREACIÓN FOLIO - INGRESO ANOTACIONES'. Below that, under 'ANOTACIONES', there are several sections: 'Datos Básicos' (Basic Data) with fields for 'Número anotación' (1), 'Tipo' (DOCUMENTO), 'Número' (9), and 'Fecha' (12/11/2020); 'Oficina de Procedencia' (Office of Origin) with 'Código' (3827), 'Nombre' (OFICINA REGISTRO OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA), 'Departamento' (Caldas), and 'Municipio' (LA DORADA); and 'Especificación' (Specification) with a 'Fecha de Radicación' (12/11/2020). The 'Modalidad' dropdown is highlighted with a red box and contains the options: VIS, VIP, VIPA, and NOVIS. A note below the dropdown says 'SELECCIONE UNA OPCIÓN'.

NOTA: al agregar el código de naturaleza jurídica 0901, en una anotación el sistema debe permitir realizar la actualización de los campos de dirección.

Para el campo Determinación del inmueble el sistema debe agregar una lista desplegable donde se visualice las diferentes opciones, como se visualiza en el cuadro 1.

Apartamento	Cementerio	Iglesia
Garaje	Centro comercial	Hacienda
Hotel	Museo	Puesto de salud/ hospital
Penthouse	Semisótano	Bodega
Circunvalar	Almacén	Solar
Planta	Sótano	Colegio
Altillo	Local	Suite
Oficina	Predio	Consultorio
Lote	Parcela	Casa Lote
Zona franca	Carretera	Parque
Terminal	Casa	Finca
Mezzanine	Parqueadero	Salón
Terraza	Deposito	Garaje
Unidad deportiva (canchas)	Teatro	Centro de animales (pesebrera, establo, etc.)
Mausoleo	Bóveda	Zona de Cesión
Cabaña		

Para el campo Modalidad el sistema debe agregar una lista desplegable donde se visualice las diferentes opciones, como se visualiza en la imagen 3.

NOTA: Esta opción se habilita para los códigos 0125 y 0126 de naturaleza jurídica.

**Modalidad** -SELECCIONE UNA OPCION- ▾

VIS  
VIP  
VIPA  
NOVIS

Para los siguientes roles el sistema debe permitir visualizar los campos agregados en la opción de creación de folios:

Rol Actuaciones Administrativas; sección consultar folio.

Rol Ampliación Tradición; sección consultar folio.

Rol Auxiliar Actuaciones Administrativas; sección consultar folio, sección editar folio

Rol Auxiliar de Correcciones; sección consultar folio, sección editar folio

Rol calificador; sección consultar folio, crear folio

Rol Coordinador Jurídico; sección consultar folio.

Rol Digitador; sección visualizar folio.

Rol Mesa Control Correcciones; sección consultar folio.

Rol Oficio de Pertenencia; sección folio.

Rol Registrador; sección folio, consultar folio.

Rol Responsable Correcciones/Revisión y análisis; sección consultar folio.

Rol Responsable Correcciones/Revisar aprobar; sección consultar folio.

Se visualiza en la imagen 4

Anotación: N° 1 del Folio #999-10448

Radicación	2020-999-6-322	Del	12/11/2020
Doc	DOCUMENTO 9	Del	12/11/2020
Oficina de Origen	OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA	De	LA DORADA
Valor		Estado	VALIDA
Especificación	COPRAVENTA PARCIAL	Naturaleza Jurídica	0126
Modalidad	VIS	Anotación Temporal	
Comentario			
Personas que intervienen en el acto (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)			
A TORRES VARGAS CARLOS - CC 987654			
Paninodor			

## ROL ACTUACIONES ADMINISTRATIVAS

Actuación Administrativa

ESTUDIO Y ANÁLISIS

Datos de la solicitud

999-10448

Consultar Seleccionada

Consultar seleccionada

Información Apertura de Folio

Círculo Registral:	CIRCULO_PRUEBAS	Dept:	BOYACA	Municipio:	BOYACA	Vereda:	BOYACA
Fecha Apertura:	21/12/2007	Radicación:	2007-999-6-1594	Con:	ESCRITURA	No. Documento:	10
Cód. Catastral:	171	Cód. Catastral Ant:		Fecha Documento:	20/12/2000		
NUPRE	987654	Avaluó	1.500.000,00	Fecha de Avaluó	17/11/2020		

ESTADO DEL FOLIO

ACTIVO

Comentario del estado:

Datos del Documento

Tipo: ESCRITURA Número: 10 Fecha: 20/12/2000

Dirección del inmueble

Tipo del Predio: RURAL

NO TIENE DIRECCIONES REGISTRADAS PARA ESTE FOLIO

Determinación del inmueble: Museo

Destinación Económica: Por determinar

Descripción: Cabida y Linderos

Underos: LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTÁ D.C.. ARTICULO 11 DECRETO 1711 DE 1984

Linderos técnicamente definidos: LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTÁ D.C.. ARTICULO 11 DECRETO 1711 DE 1984

Area privada	Metros: 50	Centímetros:	Coeficiente: 10
Area construida	Metros: 48	Centímetros:	
Area de terreno	Hectáreas: 00	Metros: 90	Centímetros: 80

Anotación: N° 1 del Folio #999-10448

Radicación	2020-999-6-322	Del	12/11/2020
Doc	DOCUMENTO 9	Del	12/11/2020
Oficina de Origen	OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA	De	LA DORADA
Valor		Estado	VALIDA
Especificación	COPRAVENTA PARCIAL	Naturaleza Jurídica	0126
Modalidad	VIS	Anotación Temporal	
Comentario			
Personas que intervienen en el acto (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)			
A TORRES VARGAS CARLOS - CC 987654			
Paninodor			

## ROL AMPLIACION TRADICION

**Certificados**

Término	INP	2020-099-1-484	Haga click para maximizar
---------	-----	----------------	---------------------------

**Folio**

Número de Matrícula	999-9873	Ver Detalles Folio
---------------------	----------	--------------------

**Cancelar**

Ver detalles de folio

**Información Apertura de Folio**

Círculo Registral:	CIRCULO_PRUEBAS	Dpto:	BOYACA	Municipio:	BOYACA	Vereda:	BOYACA
Fecha Apertura:	21/12/2007	Redicisión:	2007-999-6-1594	Con:	ESCRITURA		
Cód. Catastral:	12	Cód. Catastral Ant:		No. Documento:	10	Fecha Documento:	20/12/2000
NÚMRE	987654	Avaluó	1.500.000,000	Fecha de Avaluó	17/11/2020		

**ESTADO DEL FOLIO** ACTIVO

Comentario del estado:

**Datos del Documento**

Tipo:	ESCRITURA	Número:	10	Fecha:	20/12/2000
-------	-----------	---------	----	--------	------------

**Dirección del inmueble**

Tipo del Predio	RURAL
NO TIENE DIRECCIONES REGISTRADAS PARA ESTE FOLIO	
Determinación del inmueble	Museo
Destinación Económica	Por determinar

**Descripción: Cabida y Linderos**

Linderos	LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C.. ARTICULO 11 DECRETO 1711 DE 1984	Area privada	Metros 50 Centímetros	Coefficiente 10
Linderos	LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C.. ARTICULO 11 DECRETO 1711 DE 1984	Area construida	Metros 48 Centímetros	
Linderos técnicamente definidos		Area de terreno	Hectáreas 0 Metros 90 Centímetros 80	

**Anotación: N° 1 del Folio #999-10448**

Radicación	2020-999-6-322	Del	12/11/2020
Doc	DOCUMENTO 9	Del	12/11/2020
Oficina de Origen	OFICINA DE REGISTRO DE INSTRUMENTOS PUBLICOS DE LA DORADA	De	LA DORADA
Valor		Estado	VALIDA
Especificación	COMPRAVENTA PARCIAL	Naturaleza Jurídica	0126
Modalidad	VIS		
Comentario		Anotación Temporal	<input type="button" value=""/>

**Personas que intervienen en el acto** (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)

A TORRES VARGAS CARLOS - CC 987654

**Participación**

**Paninador**

## ROL AUXILIAR ACTUACIONES ADMINISTRATIVAS

**Correcciones**

**Realizar Corrección**

**Matrícula Inmobiliaria de la Propiedad**

Matrícula	999-546	<input type="button" value=""/>
	999-10452 - PISO 10	<input type="button" value=""/>

Ver detalles de folio

**Información Apertura de Folio**

Círculo Regional:	CIRCULO_PRUEBAS	Depto:	BOYACA	Municipio:	BOYACA	Vereda:	BOYACA
Fecha Apertura:	21/12/2007	Radicación:	2007-999-6-1594	Con:	ESCRITURA	Fecha Documento:	20/12/2000
Cód. Catastral:	123	Cód. Catastral Ant:		No. Documento:	10		
NUPRE:	987654	Avaluó:	1.500.000,00	Fecha de Avaluó:	17/11/2020		

**ESTADO DEL FOLIO** ACTIVO

Comentario del estado:

**Datos del Documento**

Tipo: ESCRITURA Número: 10 Fecha: 20/12/2000

**Dirección del inmueble**

Tipo del Predio: RURAL

NO TIENE DIRECCIONES REGISTRADAS PARA ESTE FOLIO

Determinación del inmueble: Museo

Destinación Económica: Por determinar

Descripción: Cabida y Linderos

Linderos	LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C., ARTICULO 11 DECRETO 1711 DE 1984	Area privada Metros 50 Centímetros 00	Coeficiente 10
Linderos	LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C., ARTICULO 11 DECRETO 1711 DE 1984	Area construida Metros 48 Centímetros 00	
Linderos técnicamente definidos		Area de terreno Hectáreas 00 Metros 90 Centímetros 80	

**Anotación: Nº 1 del Folio #999-10446**

Radicación	2020-999-6-322	Del	12/11/2020
Doc	DOCUMENTO 9	Del	12/11/2020
Oficina de Origen	OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA	De	LA DORADA
Valor		Estado	VALIDA
Especificación	COMPRAVENTA PARCIAL	Naturaleza Jurídica	0126
Modalidad	VIS	Anotación Temporal	
Comentario			

Personas que intervienen en el acto (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)

A TORRES VARGAS CARLOS - CC 987654

## Editar folio

**Correcciones**

Realizar Corrección

**Matrícula Inmobiliaria de la Propiedad**

Matrícula	999-546	Editar
	999-10452 - PISO 10	Editar

**Documento**

Datos Básicos

Tipo: 11 ESCRITURA Número: Fecha: 14/3/1990

Oficina de Procedencia

Departamento: Municipio: Número:

Radicación y Fecha

Num Radicación: 1701 Fecha Apertura: 21/09/1990

**Datos Folio**

Datos Folio

Tipo de Predio: RURAL Estado folio: ACTIVO

Código Central: Código Catastral:

Código Asesor:

Nupre:

Coeficiente:

Determinación del inmueble:

Modalidad:

Avaluó: Fecha Avaluó:

Destinación Económica:

Linderos: VERLOS ESCRITURA 230, 30-07-90 NOTARIA DE TURMEQUE, LOTE:

Linderos Técnicamente definidos: VERLOS ESCRITURA 230, 30-07-90 NOTARIA DE TURMEQUE, LOTE:

Area privada: Metros: Centímetros:

Area construida: Metros: Centímetros:

Area de terreno: Hectáreas: Metros: Centímetros:

Complementación: 011- PINZON ALBINO JOSE, ADQUISICION EN MAYOR EXTENSION POR COMPA

Comentario: SE REABRE

## ROL AUXILIAR DE CORRECCIONES

**Correcciones**

Realizar Corrección

Matrícula Transmisible de la Propiedad

Matrículas	Editar
999-896	
999-997	
999-10312	
999-10324	
999-10409 - LOTE 1	
999-10410 - LOTE 2	
999-10411 - LOTE 3	
999-10412 - LOTE 4	
999-10413 - LOTE 5	
999-10414	

Ver detalle de folio

**Información Apertura de Folio**

Círculo Registral: CIRCULO_PRUEBAS	Dept: BOYACA	Municipio: BOYACA	Vereda: BOYACA
Fecha Apertura: 21/12/2007	Radicación: 2007-999-6-1594	Con: ESCRITURA	Fecha Documento: 20/12/2000
Cód. Catastral: 121	Cód. Catastral Ant:	No. Documento: 10	
NUPRE: 987654	Avaluó: 1.500.000,00	Fecha de Avaluó: 17/11/2020	

**ESTADO DEL FOLIO** ACTIVO

Comentario del estado:

**Datos del Documento**

Tipo: ESCRITURA Número: 10 Fecha: 20/12/2000

**Dirección del inmueble**

Tipo del Predio: RURAL

NO TIENE DIRECCIONES REGISTRADAS PARA ESTE FOLIO

Determinación del inmueble: Museo

Destinación Económica: Por determinar

**Descripción: Cabida y Linderos**

LOTE 15 CON EXTENSION DE 40 MTRS2, CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C.. ARTICULO 11 DECRETO 1711 DE 1984

Linderos	Area privada: Metros 50 Centímetros <input type="text"/> Coeficiente: 10
	Area construida: Metros 48 Centímetros <input type="text"/>
	Area de terreno: Hectáreas <input type="text"/> Metros 90 Centímetros 80 <input type="text"/>

LOTE 15 CON EXTENSION DE 45 MTRS2, CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C.. ARTICULO 11 DECRETO 1711 DE 1984

Linderos técnicamente definidos

**Anotación: Nº 1 del Folio #999-10448**

Radicación: 2020-999-6-322	Del: 12/11/2020
Doc: DOCUMENTO 9	Del: 12/11/2020
Oficina de Origen: OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA	De: LA DORADA
Valor:	Estado: VALIDA
Especificación: COMPRAVENTA PARCIAL	Naturaleza Jurídica: U125
Modalidad: VIS	
Comentario:	Anotación Temporal
Personas que intervienen en el acto (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)	
A TORRES VARGAS CARLOS - CC 987654	Participación <input type="checkbox"/>
Parenado:	

## Editar folio

## ROL CALIFICADOR

## Consultar folio

Anotación: N° 1 del Folio #999-10448

Radicación	2020-999-6-322	Del	12/11/2020
Doc	DOCUMENTO 9	Del	12/11/2020
Oficina de Origen	OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA	De	LA DORADA
Valor		Estado	VALIDA
Especificación	COMPRAVENTA PARCIAL	Naturaleza Jurídica	0126
Modalidad	VIS	Anotación Temporal	
Comentario			
Personas que intervienen en el acto (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)			
A TORRES VARGAS CARLOS - CC 987654		Participación	
Patinador			

## Crear folio

Menú

- Revisión y Análisis
- Inscripción y Registro
- Inscripción
- Inscripción Parcial
- Devolución

Inscripción de folios

Matrículas

999-6464

**Crear Folio**

Cerrar Folios

Básicos

Tipo de Predio	-SELECCIONE UNA OPCIÓN-		
Código Catastral			
NUPRE			
Determinación del inmueble	-SELECCIONE UNA OPCIÓN-		
Avaluó			
Destinación económica			
Área privada	Metros	Centímetros	
Área construida	Metros	Centímetros	
Coeficiente			
Área	Hectáreas	Metros	Centímetros
Linderos			
Linderos técnicamente definidos			
Complementación	NUEVA	Cargar Complementación	

CALIFICACIÓN - CREACIÓN FOLIO - INGRESO ANOTACIONES

ANOTACIONES

Datos Básicos	Número anotación	1				
Tipo	DOCUMENTO	Número	9			
Oficina de Procedencia	Código	3827	Nombre	OFICINA REGISTRO OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA		
	Departamento	CALDAS	Municipio	LA DORADA		
Especificación	Fecha de Radicación	12/11/2020				
Naturaleza Jurídica	Código	0125	Descripción	COMPRAVENTA	Modalidad	-SELECCIONE UNA OPCIÓN-
Comentario	VIS	VIP	VIPA	NOVIS		

## Segregar folio

**ASISTENTE SEGREGACIÓN - PASO 1 - CREACIÓN ANOTACIÓN**

Datos Básicos  
Número anotación: 999-562 | Folio:

Datos del documento  
Tipo: JUICIO PERTENENCIA | Número: 2020-896  
Fecha: 11/9/2020  
Oficina procedencia: 313 | Nombre: JUZGADO JUZGADO SETENTA Y DOS CIVIL MUNICIPAL  
Departamento: CUNDINAMARCA | Municipio: AGUA DE DIOS

Especificación  
Fecha de Radicación: 14/9/2020  
Naturalidad Jurídica: ID 0126 | Descripción: COMPRAVENTA PARCIAL | Modalidad: SELECCIONE UNA OPCIÓN  
Comentario:  
Copiar comentario en los folios segregados? Sí | No

**ASISTENTE SEGREGACIÓN - PASO 3 - CREAR FOLIOS DERIVADOS**

Folios Derivados  
Agregar Datos  
Inmueble: Área (Seleccione las unidades de medida del predio)  
Hectáreas: [ ] | Metros: [ ] | Centímetros: [ ] | Coeficiente: [ ]  
Agregar Segregación

Segregar | Guardar | Volver | Cancelar | Eliminar Segregación

## ROL COORDINADOR JURIDICO

**Aprobar Ajuste Calificación**

Datos Básicos  
Tipo: CERTIFICADO | Número: 1234567 | Fecha: 27/8/2020

Datos Oficina de procedencia  
Codigo: 5178 | Departamento: BOGOTA D.C. | Nombre: JUZGADO JUZGADO PRIMERO CIVIL DEL CIRCUITO | Municipio: SOACHA

Turno Anterior  
Número del Turno Anterior:

Matrículas Asociadas al Turno  
Matrículas: 999-10390, 999-10391, 999-10392 | Revisar Folio | Revisar Folio | Revisar Folio

## Revisar folio

**Información Apertura de Folio**

Círculo Registral: CÍRCULO\_PRUEBAS | Depto: BOYACA | Municipio: BOYACA | Vereda: BOYACA  
Fecha Apertura: 21/12/2007 | Radicación: 2007-999-6-1594 | Con: ESCRITURA | Nro. Documento: 10 | Fecha Documento: 20/12/2000  
Cód. Catastral: 123 | Cód. Catastral Ant: 123  
NUPRE: 987654  
Avaluó: 1,500,000.000 | Fecha de Avaluó: 17/11/2020

**ESTADO DEL FOLIO**  
ACTIVO  
Comentario del estado:

**Datos del Documento**  
Tipo: ESCRITURA | Número: 10 | Fecha: 20/12/2000

**Dirección del inmueble**  
Tipo del Predio: RURAL  
NO TIENE DIRECCIONES REGISTRADAS PARA ESTE FOLIO  
Determinación del inmueble: Museo  
Destinación Económica: Por determinar

**Descripción: Cabida y Linderos**  
LOTE 15 CON EXTENSION DE 45 MTRS2. CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C.. ARTICULO 11 DECRETO 1711 DE 1984  
Underos  
Linderos técnicamente definidos  
Area privada: Metros: 50 | Centímetros: [ ] | Coeficiente: 10  
Area construida: Metros: 48 | Centímetros: [ ]  
Area de terreno: Hectáreas: [ ] | Metros: 90 | Centímetros: 80

Anotación: N° 1 del Folio #999-10448

Radicación	2020-999-6-322	Del	12/11/2020
Doc	DOCUMENTO 9	Del	12/11/2020
Oficina de Origen	OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA	De	LA DORADA
Valor		Estado	VALIDA
Especificación	COMPRAVENTA PARCIAL	Naturaleza Jurídica	0126
Modalidad	VIS		
Comentario		Anotación Temporal	
Personas que intervienen en el acto (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)			
A TORRES VARGAS CARLOS - CC 987654			
<input type="checkbox"/> Paninador			

## ROL DIGITADOR

folios

Matrículas			
999-456			
999-9629 - LOTE 1			
999-9630 - LOTE 2			
999-9631 - LOTE 3			
999-9632 - LOTE 4			
999-9633 - LOTE 5			
999-9634 - LOTE 6			
999-9635 - LOTE 7			
999-9636 - LOTE 8			
999-9637 - LOTE 9			
999-9638 - LOTE 10			

Ver folio

Información Apertura de Folio

Círculo Registral:	CIRCULO_PRUEBAS	Depo:	BOYACA	Municipio:	BOYACA	Vereda:	BOYACA			
Fecha Apertura:	21/12/2007	Radicación:	2007-999-6-1594	Con:	ESCRITURA	No. Documento:	10	Fecha Documento:	20/12/2000	
NUPRE	987654	Cód. Catastral Apt:	123							
Avaluó	1,500,000,000			Fecha de Avaluó	17/11/2020					
ESTADO DEL FOLIO <b>ACTIVO</b>										
Comentario del estado:										
Datos del Documento										
Tipo:	ESCRITURA	Número:	10	Fecha:	20/12/2000					
Dirección del inmueble										
Tipo del Predio	RURAL									
NO TIENE DIRECCIONES REGISTRADAS PARA ESTE FOLIO										
Determinación del inmueble	Museo									
Destinación Económica	Por determinar									
Descripción: Cabida y Linderos										
Underos	LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTÁ D.C.. ARTICULO 11 DECRETO 1711 DE 1984									
	Area privada	Metros	50	Centímetros					Coefficiente	10
	Area construida	Metros	48	Centímetros						
	Area de terreno	Hectáreas		Metros	90	Centímetros	80			
Linderos técnicamente definidos	LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTÁ D.C.. ARTICULO 11 DECRETO 1711 DE 1984									

Anotación: N° 1 del Folio #999-10448

Radicación	2020-999-6-322	Del	12/11/2020
Doc	DOCUMENTO 9	Del	12/11/2020
Oficina de Origen	OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA	De	LA DORADA
Valor		Estado	VALIDA
Especificación	COMPRAVENTA PARCIAL	Naturaleza Jurídica	0126
Modalidad	VIS		
Comentario		Anotación Temporal	
Personas que intervienen en el acto (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)			
A TORRES VARGAS CARLOS - CC 987654			
<input type="checkbox"/> Paninador			

## ROL MESA CONTROL CORRECCIONES

**Correcciones**

Mesa Control

Matrícula Inmobiliaria de la Propiedad

○ 999-10190

Consultar Seleccionada

Consultar Folios válidos para imprimir Consultar

Turbo Asociado

**Consultar seleccionada**

**Información Apertura de Folio**

Círculo Regional: CÍRCULO\_PRUEBAS Depto: BOYACÁ Municipio: BOYACÁ Vereda: BOYACÁ  
 Fecha Apertura: 21/12/2007 Radicación: 2007-999-6-1594 Con: ESCRITURA No. Documento: 10 Fecha Documento: 20/12/2000  
 Cód. Catastral: 123 Cód. Catastral Ant: 123

NUIPSE: 987654  
 Avalúo: 1,500,000,000 Fecha de Avalúo: 17/11/2020

ESTADO DEL FOLIO: ACTIVO  
 Comentario del estado:

Datos del Documento  
 Tipo: ESCRITURA Número: 10 Fecha: 20/12/2000

Dirección del inmueble  
 Tipo del Predio: RURAL  
 IN TIENE DIRECCIONES REGISTRADAS PARA ESTE FOLIO  
 Determinación del inmueble: Museo  
 Destinación Económica: Por determinar

Descripción: Cabida y Linderos

Lote 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTÁ D.C.. ARTICULO 11 DECRETO 1711 DE 1984

Underos: Área privada Metros: 50 Centímetros: 0 Coeficiente: 10  
 Área construida Metros: 48 Centímetros: 0  
 Área de terreno Hectáreas: 0 Metros: 90 Centímetros: 80

Linderos técnicamente definidos: LOTE 15 CON EXTENSION DE 45 MTRS2, CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTÁ D.C.. ARTICULO 11 DECRETO 1711 DE 1984

**Anotación: N° 1 del Folio #999-10448**

Radicación: 2020-999-6-322 Del: 12/11/2020  
 Doc: DOCUMENTO 9 Del: 12/11/2020  
 Oficina de Origen: OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA De: LA DORADA  
 Valor: Estado: VALIDA  
 Especificación: COMPROVANTIA PARCIAL Naturaleza Jurídica: 0126  
 Modalidad: VIS Anotación Temporal:

Personas que intervienen en el acto (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)  
 A TORRES VARGAS CARLOS - CC 987654 Participación

Página: 1 de 1

**ROL REGISTRADOR**

Turbo

Nº 2020-999-6-240 Haga click para maximizar

Folios

Listado de Folios del Turno

○ Número de Matrícula: 999-5682  
 ○ Número de Matrícula: 999-7885  
 ○ Número de Matrícula: 999-9532

**Información Apertura de Folio**

Círculo Registrado:	CÍRCULO PRUEBAS	Dpto:	BOYACA	Municipio:	BOYACA	Vereda:	BOYACA
Fecha Apertura:	21/12/2007	Realización:	2007-999-6-1594	Con:	ESCRITURA		
Cód. Catastral:	152	Cód. Catastral Ant:		Nro. Documento:	10	Fecha Documento:	20/12/2000
NUPRE:	987654	Avaluó:	1.500.000,00	Fecha de Avaluó:	17/11/2020		

**ESTADO DEL FOLIO** ACTIVO

Comentario del estado:

**Datos del Documento**

Tipo: ESCRITURA Número: 10 Fecha: 20/12/2000

**Dirección del inmueble**

Tipos del Predio: RURAL

NO TIENE DIRECCIONES REGISTRADAS PARA ESTE FOLIO

Determinación del inmueble: Museo

Destinación Económica: Por determinar

**Descripción: Cabida y Linderos**

LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C.. ARTICULO 11 DECRETO 1711 DE 1984

Underos: Coeficiente: 10

Area privada: Metros 50 Centímetros

Area construida: Metros 48 Centímetros

Area de terreno: Hectáreas 00 Metros 90 Centímetros 80

LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C.. ARTICULO 11 DECRETO 1711 DE 1984

Linderos técnicamente definidos:

**Anotación: Nº 1 del Folio #999-10448**

Radicación:	2020-999-6-322	Del:	12/11/2020
Doc:	DOCUMENTO 9	Del:	12/11/2020
Oficina de Origen:	OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA	De:	LA DORADA
Valor:		Estado:	VALIDA
Especificación:	COMPRAVENTA PARCIAL	Naturaleza Jurídica:	0126
Modalidad:	VIS		
Comentario:		Anotación Temporal:	(Iconos)
Personas que intervienen en el acto (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)			
A TORRES VARGAS CARLOS - CC 987654		Participación:	
Patinador:			

### ROL RESPONSABLE CORRECCIONES / REVISAR APROBAR

**Correcciones**

Aprobar Corrección

**Matrícula Inmobiliaria de la Propiedad**

999-2589

Consultar Seleccionada

Consulta Seleccionada

**Información Apertura de Folio**

Círculo Regional:	CIRCULO_PRUEBAS	Deptó:	BOYACA	Municipio:	BOYACA	Vereda:	BOYACA
Fecha Apertura:	21/12/2007	Radicación:	2007-999-6-1594	Con:	ESCRITURA	Fecha Documento:	20/12/2000
Cód. Catastral:	123	Cód. Catastral Ant:		No. Documento:	10		
NUPRE	987654	Avaluó	1.500.000,00	Fecha de Avaluó	17/11/2020		

**ESTADO DEL FOLIO** ACTIVO

Comentario del estado:

**Datos del Documento**

Tipo: ESCRITURA Número: 10 Fecha: 20/12/2000

**Dirección del inmueble**

Tipo del Predio: RURAL

NO TIENE DIRECCIONES REGISTRADAS PARA ESTE FOLIO

Determinación del inmueble: Museo

Destinación Económica: Por determinar

**Descripción: Cabida y Linderos**

Linderos	LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C., ARTICULO 11 DECRETO 1711 DE 1984	Area privada Metros 50 Centímetros 0	Coeficiente 10
Linderos	LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C., ARTICULO 11 DECRETO 1711 DE 1984	Area construida Metros 48 Centímetros 0	
Linderos técnicamente definidos		Area de terreno Hectáreas 0 Metros 90 Centímetros 80	

**Anotación: Nº 1 del Folio #999-10448**

Radicación	2020-999-6-322	Del	12/11/2020
Doc	DOCUMENTO 9	Del	12/11/2020
Oficina de Origen	OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA	De	LA DORADA
Valor		Estado	VALIDA
Especificación	COMPRAVENTA PARCIAL	Naturaleza Jurídica	0126
Modalidad	VIS		
Comentario		Anotación Temporal	
Personas que intervienen en el acto (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)		Participación	
A TORRES VARGAS CARLOS - CC 987654			

## ROL RESPONSABLE CORRECCIONES / REVISION Y ANALISIS

**Correcciones**

Revisión y Análisis

**Matrícula Inmobiliaria de la Propiedad**

<input checked="" type="checkbox"/> 999-9876	<input type="button" value="Borrar"/>
--	---------------------------------------

Ver folio

**Información Apertura de Folio**

Círculo Regional:	CIRCULO_PRUEBAS	Depto:	BOYACA	Municipio:	BOYACA	Vereda:	BOYACA
Fecha Apertura:	21/12/2007	Radicación:	2007-999-6-1594	Con:	ESCRITURA		
Cód. Catastral:	123	Cód. Catastral Ant:		No. Documento:	10	Fecha Documento:	20/12/2000
NUPRE	987654	Avaluó	1.500.000,00	Fecha de Avaluó	17/11/2020		

**ESTADO DEL FOLIO** ACTIVO

Comentario del estado:

**Datos del Documento**

Tipo: ESCRITURA Número: 10 Fecha: 20/12/2000

**Dirección del inmueble**

Tipo del Predio: RURAL

NO TIENE DIRECCIONES REGISTRADAS PARA ESTE FOLIO

Determinación del inmueble: Museo

Destinación Económica: Por determinar

**Descripción: Cabida y Linderos**

Linderos	LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C., ARTICULO 11 DECRETO 1711 DE 1984	Area privada Metros: 50 Centímetros: 0	Coeficiente: 10
Linderos técnicamente definidos	LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C., ARTICULO 11 DECRETO 1711 DE 1984	Area construida Metros: 48 Centímetros: 0	
		Area de terreno Hectáreas: 0 Metros: 90 Centímetros: 80	

**Anotación: Nº 1 del Folio #999-10448**

Radicación	2020-999-6-322	Del	12/11/2020
Doc	DOCUMENTO 9	Del	12/11/2020
Oficina de Origen	OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA	De	LA DORADA
Valor		Estado	VALIDA
Especificación	COMPRAVENTA PARCIAL	Naturaleza Jurídica	0126
Modalidad	VIS		
Comentario		Anotación Temporal	
Personas que intervienen en el acto (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)		Participación	
A TORRES VARGAS CARLOS - CC 987654			

## PANTALLAS ADMINISTRATIVAS

Ver detalles de folio

**Pantalla Administrativa Consulta de Detalles de Folios**

**Consulta de Detalles de Folios**

Datos del Folio

Número del folio: 999-10294

Buscar Volver

**Información Apertura de Folio**

Círculo Regional:	CIRCULO_PRUEBAS	Depto:	BOYACA	Municipio:	BOYACA	Vereda:	BOYACA
Fecha Apertura:	21/12/2007	Radicación:	2007-999-6-1594	Con:	ESCRITURA		
Cód. Catastral:	123	Cód. Catastral Ant:		No. Documento:	10	Fecha Documento:	20/12/2000
NUPRE	987654	Avaluó	1.500.000,00	Fecha de Avaluó	17/11/2020		

**ESTADO DEL FOLIO** ACTIVO

Comentario del estado:

**Datos del Documento**

Tipo: ESCRITURA Número: 10 Fecha: 20/12/2000

**Dirección del inmueble**

Tipo del Predio: RURAL

NO TIENE DIRECCIONES REGISTRADAS PARA ESTE FOLIO

Determinación del inmueble: Museo

Destinación Económica: Por determinar

**Descripción: Cabida y Linderos**

Linderos	LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C., ARTICULO 11 DECRETO 1711 DE 1984	Area privada Metros 50 Centímetros 0	Coeficiente 10
Linderos técnicamente definidos	LOTE 15 CON EXTENSION DE 45 MTRS2 CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 10, 2000/12/20, CAJA DE RETIRO DE LAS FUERZAS MILITARES BOGOTA D.C., ARTICULO 11 DECRETO 1711 DE 1984	Area construida Metros 48 Centímetros 0	
		Area de terreno Hectáreas 0 Metros 90 Centímetros 80	

**Anotación: Nº 1 del Folio #999-10448**

Radicación	2020-999-6-322	Del	12/11/2020
Doc	DOCUMENTO 9	Del	12/11/2020
Oficina de Origen	OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE LA DORADA	De	LA DORADA
Valor		Estado	VALIDA
Especificación	COMPRAVENTA PARCIAL	Naturaleza Jurídica	0126
Modalidad	VIS		
Comentario		Anotación Temporal	
Personas que intervienen en el acto (La X indica a la persona que figura como titular de derechos reales de dominio, I-Titular de dominio incompleto)		Participación	
A TORRES VARGAS CARLOS - CC 987654			

## Grabación directa de folios

**FOLIOS ANOTACIONES Y ACTOS**

- Reabrir Folios
- Ver detalles de Folio
- Administración de Traslados de Folios
- Grabación Directa de Folios
- Desbloqueo Folio Restitución
- Creación Directa de Folios
- Administración de Traslados de Folios Masivo
- Historial archivos trasladados masivo

**Básicos**

Tipo de Predio: -SELECCIONE UNA OPCION- ▾  
Código Catastral:  
NUPRE:  
Determinación del inmueble: -SELECCIONE UNA OPCION- ▾  
Avaluó:  
Destinación económica:  
Area privada: Metros: [input field] Centímetros: [input field]  
Area construida: Metros: [input field] Centímetros: [input field]  
Coeficiente:  
Area: Hectáreas: [input field] Metros: [input field] Centímetros: [input field]  
Linderos:  
Linderos técnicamente definidos:  
Complementación: NUEVA ▾ [input field] Cargar Complementación [button]

Al agregar una anotación en la creación del folio, si su naturaleza jurídica corresponde al código 0125 o 0126, el sistema habilita el campo de Modalidad, mediante la lista desplegable (VIS, VIP, VIPA, NOVIS), se visualiza en la imagen

Especificación  
Fecha de Radicación: 24/11/2020  
Naturalidad Jurídica: Código 0126 Descripción: COMPRAVENTA PARCIAL  
Modalidad: -SELECCIONE UNA OPCION- ▾  
VIS  
VIP  
VIPA  
NOVIS  
Comentario:

Creación directa de folios

FOLIOS ANOTACIONES Y ACTOS  
Reabrir Folios  
Ver detalles de Folio  
Administración de Traslados de Folios  
Grabación Directa de Folios  
Desbloqueo Folio Restitución  
Creación Directa de Folios  
Administración de Traslados de Folios Masivo  
Historial archivos trasladados masivo

**Básicos**

Tipo de Predio	-SELECCIONE UNA OPCION-		
Código Catastral			
NUPRE			
Determinación del inmueble	-SELECCIONE UNA OPCION-		
Avaluó			
Destinación económica			
Área privada	Metros	Centímetros	
Área construida	Metros	Centímetros	
Coeficiente			
Área	Hectáreas	Metros	Centímetros
Linderos			
Linderos técnicamente definidos			
Complementación	NUEVA		Cargar Complementación

Al agregar una anotación en la creación del folio, si su naturaleza jurídica corresponde al código 0125 o 0126, el sistema habilita el campo de Modalidad, mediante la lista desplegable (VIS, VIP, VIPA, NOVIS), se visualiza en la imagen

**Especificación**

Fecha de Radicación	24/11/2020				
Naturaleza Jurídica	Código 0126	Descripción	COMPRAVENTA PARCIAL	Modalidad	-SELECCIONE UNA OPCION-
Comentario	<input type="checkbox"/> VIS <input type="checkbox"/> VIP <input type="checkbox"/> VIPA <input type="checkbox"/> NOVIS				



## DESPLIEGUE 7.4.9

27/11/2020



### 448204- IMPLEMENTACIÓN NUEVAS NOTAS DEVOLUTIVAS

Actualmente en el aplicativo SIR, se implementa las causales de no inscripción de documentos de registro por bloques temáticos, por lo que se realiza una parametrización en el aplicativo SIR de estas causales, de acuerdo con el comunicado SNR2020IE015448 del 06 de noviembre de 2020, firmado por el doctor Marco Antonio Zuñiga Arrieta – Director Técnico de Registro.

Se configura por base de datos las diferentes causales de no inscripción de documentos (devolución) con su correspondiente bloque temático y código correspondiente, se visualiza en la imagen 1 y 2, esto se visualiza por el rol Calificador en su fase de devolución.

Imagen 1

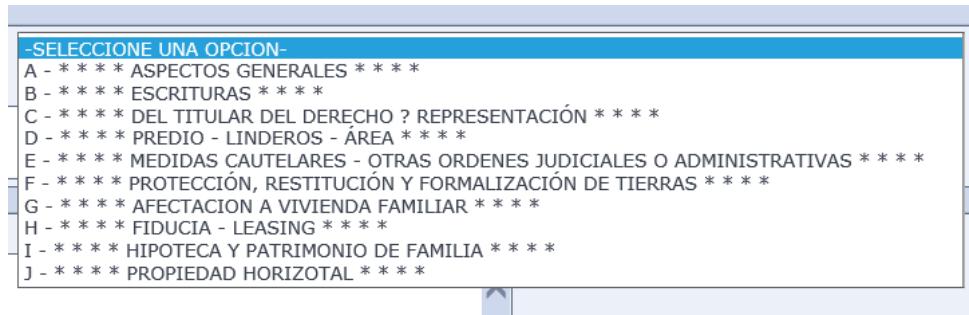
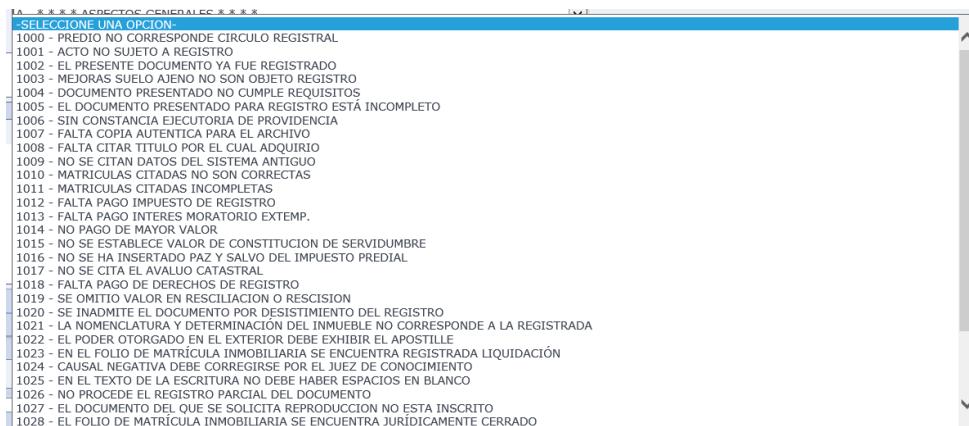


Imagen 2



**OC 2418-CORRECCIONES CANALES DE RECAUDO Y FORMAS DE PAGO**

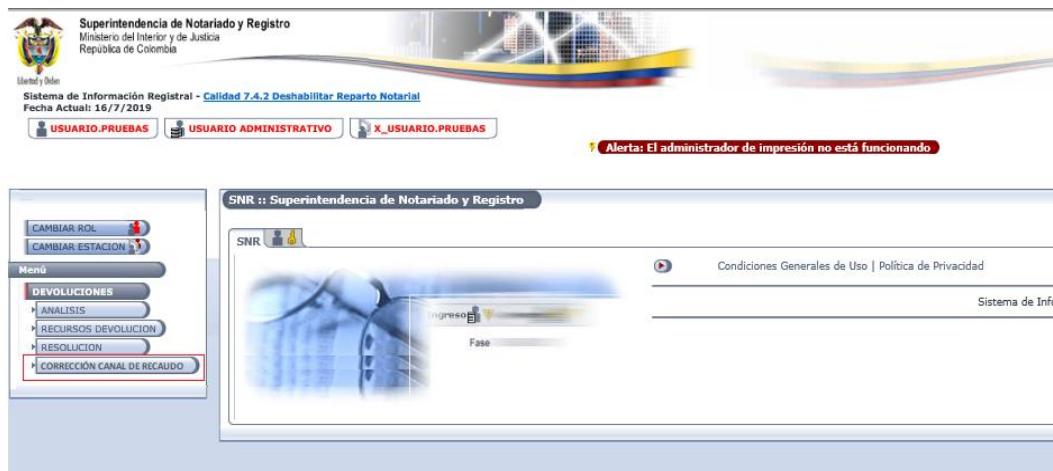
Actualmente en el aplicativo SIR, se encuentra implementado el desarrollo que permite el ingreso de canales de recaudo y forma de pago; sin embargo no existe una opción por donde se puedan corregir estos datos en caso de cometer errores en el momento de la radicación del documento, de acuerdo con el comunicado SNR2019IE01403 del 02 de mayo enviado por el Director Técnico de Registro, solicita se permita realizar la corrección de los datos dentro de los tres días hábiles siguientes a la radicación del documento.

Es necesario crear una nueva pantalla en el rol usuario administrativo en SIR, con el fin que por intermedio de esta se permita realizar la corrección de los datos de canales de recaudo y formas de pago que por error sean almacenados al momento de registrar los turnos en el sistema misional.

**❖ PANTALLA USUARIO ADMINISTRATIVO**

Se crea en la vista de las fases del usuario administrativo una nueva fase nombrada corrección canal de recaudo, la cual al dar clic activa la pantalla “Corrección canal de recaudo”. Se visualiza en imagen 1.

Imagen 1.

**❖ PANTALLA CORRECCIÓN CANAL DE RECAUDO**

Se crea una pantalla con las siguientes especificaciones:

- Nombre corrección canal de recaudo
- Subtítulo “Turno asociado”
- Caja de texto: en este campo se digitará el turno sobre el cual se realizará la corrección del canal y la forma de pago.

- Botón regresar, este debe volver al menú de roles del sistema.
- Botón “Aregar Turno”, al dar clic sobre dicho botón el sistema debe consultar todas las formas de pago del turno, Nota tener en cuenta que un turno puede tener formas de pago sobre un mayor valor. El modelo de la pantalla se visualiza en la imagen 2

Imagen 2

- Función consultar turno

Se desarrolla en el sistema la función que permite consultar un turno y mostrar las formas de pago asociadas a este.

Al ingresar el turno el sistema visualiza las formas de pago seguido de un botón nombrado “corregir” o de un campo de texto informativo indicando el siguiente estado “vencido para corrección”, adicionalmente se implementa un botón de nota informativa. Se visualiza en imagen 3

Imagen 3

- El estado “vencido para corrección” se visualiza cuando hayan pasado 3 días hábiles a partir de la creación de la forma de pago en el sistema.
- El botón “corregir” se visualiza si la forma de pago esta habilitada para su corrección, al dar clic sobre este se activa la pantalla de corrección liquidador desde donde se visualiza los campos de canales de recaudo parametrizados para el circulo registral.
- Nota informativa visualiza la opción de ingresar la descripción del motivo de corrección, es obligatoria y los campos de forma de pago no podrán ser corregidos si previamente no se a ingresado la nota informativa.

### ❖ NOTA INFORMATIVA

- Se crea una nota informativa en el sistema que permita llevar la traza de la corrección de forma de pago
- La nota es obligatoria para la corrección de la forma de pago.
- Estructura de la nota: “XXX- NOTA CORRECCIÓN FORMA DE PAGO”  
Se visualiza en imagen 4

Imagen 4

NOTA INFORMATIVA CORRECCIÓN FORMA DE PAGO

USUARIO.PRUEBAS

FGHGFJHGJFGJJ

### ❖ PANTALLA CORRECCIÓN LIQUIDADOR

Se desarrolla una pantalla la cual se activa al dar clic en el botón “Corregir”, desde esta pantalla se puede modificar la forma de pago seleccionada con las siguientes características:

- Título Corrección liquidador.
- Grilla donde se visualiza los datos de la forma de pago seleccionada esto solo es informativo.
- Grilla donde se selecciona el canal de recaudo, forma de pago y los datos de captura de dicha forma de pago. Nota este no muestra los campos de valores.
- Botón regresar, este regresa a la pantalla corrección canal de recaudo.
- Botón aceptar activa la alerta si está seguro de guardar los cambios.

Se visualiza en imagen 5

Imagen 5

Turno asociado

Turno sobre el cual se realizará la corrección 2019-999-6-1073

Razón del pago: MAYOR VALOR

Forma de Pago: CONSIGNACION\_LDR

Banco: BANCAMIA

Número Consignación: GOGOFG11

Saldo a favor: 0.00

Canal de Recaudo: SELECCIONE UNA OPCION-

Forma de Pago: SELECCIONE UNA OPCION-

BANCO/FRANQUICIA: SELECCIONE UNA OPCION-

NO APROBACION

FECHA DOCUMENTO

CUENTA DESTINO: SELECCIONE UNA OPCION-

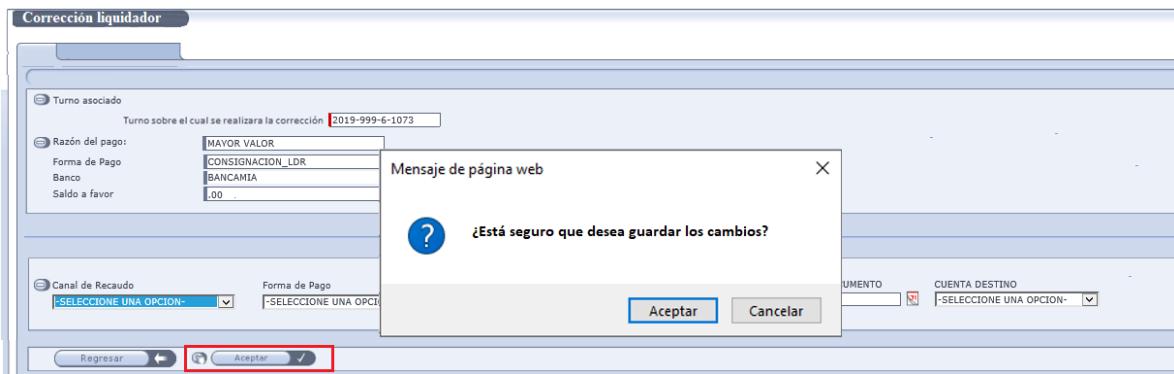
Regresar Aceptar

Nota: los datos del turno asociados son solo de consulta, y los datos de la forma de pago tienen el mismo comportamiento de la pantalla de liquidador (Canal de recaudo – Forma de pago – campos de captura).

## ❖ ALERTAS

Una vez se haya realizado la corrección de la forma de pago y dado clic en el botón aceptar el sistema alerta al funcionario de si esta seguro de realizar la modificación. Se visualiza en imagen 6

Imagen 6



Al dar clic en aceptar el sistema valida que se haya agregado una nota informativa de lo contrario se notifica mediante un mensaje en la parte superior de la pantalla. Se visualiza en imagen 7

Imagen 7

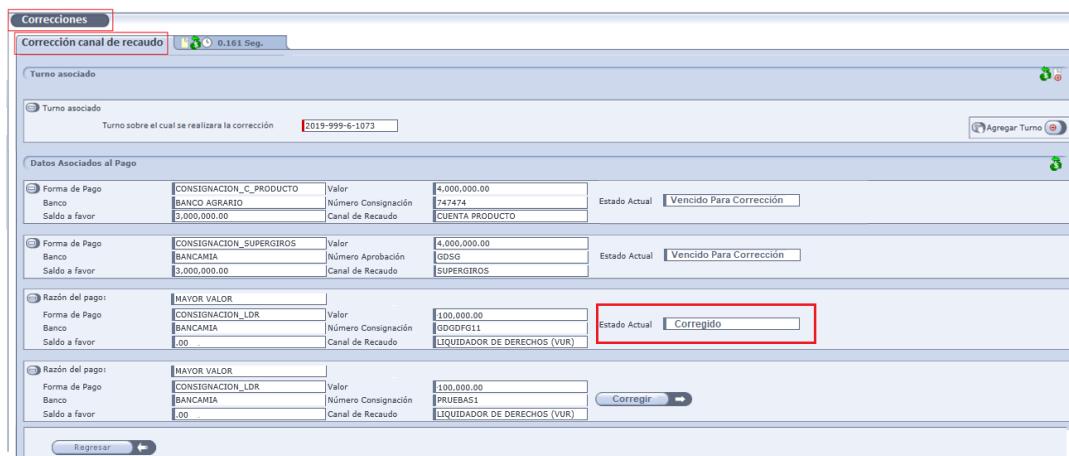


Si la nota informativa fue agregada el sistema debe realizar las siguientes acciones:

- Almacenar los datos modificados.
- volver a la pantalla anterior “Corrección canal de recaudo”.
- Cambiar el estado a “Corregido”.
- Desactivar y dejar de mostrar el botón corregir para dicha forma de pago.

Se visualiza en imagen 8

Imagen 8



Nota el sistema sigue con la lógica de guardar en turno historia las fases por donde a pasado el turno, cual se visualiza desde la pantalla ver detalles de turno.

#### ❖ PANTALLA VER DETALLES DE TURNO

Se adiciona en la pantalla ver detalles de turno un botón de visualización y una etiqueta “Forma de pago corregida” que identifica las formas de pago que hayan sido sujetas de alguna modificación, el botón de visualización permite activar una pantalla emergente con los siguientes datos:

- Datos anteriores a la modificación “Canal de recaudo, forma de pago, campos captura”
- Usuario y fecha de la creación del pago.
- Usuario y fecha de la modificación.

Se visualiza en imágenes 9 y 10

Imagen 9

Pago Mayor Valor

Razón del pago:	MAYOR VALOR
Valor derechos de registro:	355.000
Valor de impuestos:	0
Valor Conservación Documental	7.100

Canal de Recaudo:

Forma de Pago:	SUPERGIROS
Banco/Franquicia:	CONSIGNACION_SUPERGIROS
CONSIGNACION_SUPERGIROS	BANCO AGRARIO - CREDENCIAL

Canal de Recaudo:

Forma de Pago:	CUENTA PRODUCTO
Banco/Franquicia:	CONSIGNACION_C_PRODUCTO
BANCAMIA - MASTER CARD	BANCAMIA

Forma de pago corregida:

Cuenta Destino:	BANCOLOMBIA-963852
No. Aprobación:	PRUEBAS1
Fecha Documento:	29/07/2019
Valor Liquidado:	100.000
Banco:	BANCO AGRARIO
Valor Documento Pago:	100.000

Cuenta Destino:	BANCOLOMBIA-123456789000
Fecha Documento:	29/07/2019
Valor Liquidado:	252.100
Banco:	BANCAMIA
No. Consignación:	PRUEBAS002
Valor Documento Pago:	200.000

Detalles Antiguo Sistema

Libro	Número-Letra
Tipo	Año
Página	
Tomo	Página
Número	Año
Municipio	

Notas informativas del turno

MAYOR VALOR (TIEMPO LIMITE PAGO)	USUARIO.PRUEBAS	PRUEBAS
NOTA INFORMATIVA CORRECCION FORMAS DE PAGO	USUARIO.PRUEBAS	VBCVBCVBCV

Imagen 10

DATOS ANTERIORES DE LA FORMA DE PAGO

Forma de Pago

Razón del pago:	MAYOR VALOR
Forma de Pago:	CONSIGNACION_LDR
Banco:	BANCAMIA
Saldo a favor:	1.00
Usuario Creación:	USUARIO.PRUEBAS
Usuario que modifica:	USUARIO.PRUEBAS1

Alerta: El administrador de impresión no está funcionando

Sistema de Información Registral - Calidad 7.4.3  
Fecha Actual: 26/8/2019

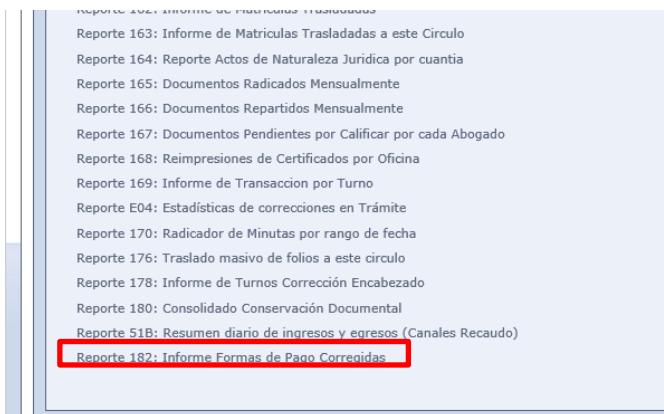
Cerrar Ventana

#### ❖ CREACIÓN DE REPORTE PANTALLA ADMINISTRATIVA

Se crea una opción en la pantalla administrativa/ lista de reportes que activa una nueva pantalla la cual consulta un informe en pdf “Informe formas de pago corregidas”; “Reporte 182: Informe formas de pago corregidas”.

Se Visualiza en imagen 11

Imagen 11



#### ❖ CREACIÓN DE REPORTE PANTALLA ADMINISTRATIVA

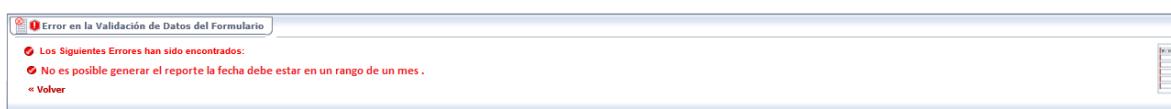
Se crea una pantalla administrativa con los siguientes criterios de búsqueda: Fecha inicial, fecha final y Circulo registral, la pantalla se activa una vez se da clic en la opción “Reporte 182: Informe formas de pago corregidas” Se visualiza en imagen 12

Imagen 12



Nota se genera una alerta la cual indica que la fecha debe estar en un rango no mayor a un mes. Se visualiza en imagen 13

Imagen 13

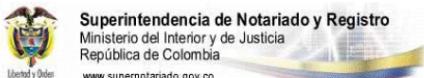


#### ❖ CREACIÓN DE REPORTE PDF

Se crea un reporte donde se visualiza las formas de pago modificadas, dicho reporte contiene los siguientes datos:

- Turno, Canal de recaudo, Forma de pago, Numero de documento, Banco Franquicia, Cuenta destino, Fecha creación/Modificación, usuario creación/Modificación, registro “Original, Corregido” Se visualiza en imagen 14.

Imagen 14



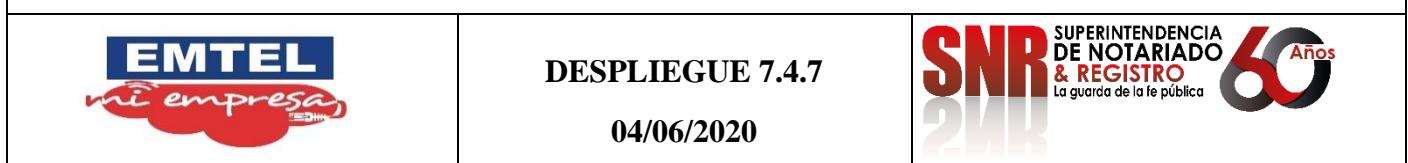
Superintendencia de Notariado y Registro  
Ministerio del Interior y de Justicia  
República de Colombia  
www.supernotariado.gov.co

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999  
INFORME FORMAS DE PAGO CORREGIDAS 10-07-2019  
REALIZADO POR: USUARIO.PRUEBAS

Impreso el 26 de Agosto de 2019 a las 12:24:41 PM  
Página 1  
USUARIO: USUARIO.PRUEBAS

Turno	Canal de Recaudo	Forma de Pago	No Doc	Banco/ Franquicia	Cuenta Destino	Fecha Creación/ Modificación	Úsuario Creación/Modificación	Registro
2019-999-8-1030	CUENTA PRODUCTO	CONSIGNACION_C_PRODUCTO	80850192ARBM0	Bancolombia/Visa	Bancolombia- 0001	23/06/2019 10:00 am	USUARIO.PRUEBAS	Original
2019-999-8-1030	CUENTA PRODUCTO	CONSIGNACION_C_PRODUCTO	80850192ARB	Bancolombia/lisa	Bancolombia- 0001	24/06/2019 03:00 pm	USUARIO.PRUEBAS 1	Corregido

Nota el registro original es la forma de pago a la cual se le realizó la corrección, el corregido son los nuevos datos.



Actualmente en el aplicativo SIR, los imprimibles de los formularios de calificación, constancia de inscripción (reproducción de sellos) y notas devolutivas se generan cuando el turno llega al rol Calificador, para luego ser aprobados y firmados por el Registrador.

Por lo anterior, se incluye la opción de imprimir o no los documentos desde la fase de calificador, cuando se realiza la inscripción, inscripción parcial.

También se incluye la visualización de los diferentes documentos por la fase Mesa de control Registro y su opción de crear relaciones.

En la fase de registrador se incluye tener la opción de elegir la impresora a la cual se enviarán los documentos previamente revisados por el registrador, para poder aceptarlos e imprimirlas y posteriormente ser entregados al ciudadano, además si el turno es de reproducción de sellos el sistema permite visualizar los certificados correspondientes.

Finalmente se incluye que los documentos de oficio de pertenecía salgan con la firma correspondiente del registrador al realizar el proceso de certificados de pertenencia.

1. En el Rol Calificador, en la fase “Inscripción”, el sistema permite elegir mediante un checkbox llamado “Imprimir certificado de calificación” imprimir o no el formulario de calificación. (Ver imagen 1)

Imagen 1

The screenshot shows a software interface titled 'Terminar calificación'. It includes fields for 'Número de copias a imprimir' (Number of copies to print), 'Seleccionar impresora' (Select printer), and a checkbox labeled 'Imprimir certificado de calificación' (Print grade certificate). A red box highlights the checkbox. At the bottom are buttons for 'Terminar' (Finish) and 'Ver Imprimible' (View printable).

2. Si el turno es de un proceso de inscripción parcial el sistema permite elegir mediante un checkbox llamado “Imprimir certificado de calificación” imprimir o no el formulario de calificación y nota devolutiva. (Ver imagen 2)

Imagen 2

The screenshot shows a software interface titled 'Inscripción parcial de folios'. It includes sections for 'Nota Devolutiva' (Devolutive Note) with dropdown menus for 'Elegir Tipo Nota Devolutiva' (Choose Devolutive Note Type) and 'Elegir Subtipo Nota Devolutiva' (Choose Devolutive Note Subtype). Below is a note about 'NO PROcede LA AFECTACIÓN A VIVIENDA FAMILIAR POR CUANTO NO SE OSTENTA LA CALIDAD DE PLENO PROPIETARIO (LEY 258 DE 1996)'. The 'Descripción' (Description) section contains placeholder text 'SDSSSDSDSDFSDFFD'. Under 'Lista de matrículas asociadas al turno' (List of registrations associated with the turn), there are two entries: '999-1983' and '999-2839'. The 'Causales de Devolución' (Reasons for Devolution) section has a note about 'NO PROcede LA AFECTACIÓN A VIVIENDA FAMILIAR POR CUANTO NO SE OSTENTA LA CALIDAD DE PLENO PROPIETARIO (LEY 258 DE 1996)'. At the bottom, there are buttons for 'Terminar inscripción parcial' (Finish partial enrollment), 'Imprimir inscripción parcial' (Print partial enrollment), and other options like 'Número de copias a imprimir' (Number of copies to print), 'Consultar' (Search), 'Terminar' (Finish), 'Salir' (Exit), 'Ver Formulario' (View Form), and 'Ver Ruta' (View Route). A red box highlights the 'Imprimir inscripción parcial' button.

3. En el rol Mesa control registro el sistema permite visualizar el documento mediante un botón “ver imprimible”, (ver imagen 3) si el turno es de inscripción parcial el sistema debe permitir visualizar mediante dos botones, “ver formulario” y “ver nota”, (ver imagen 4), Adicionalmente, al ingresar por la opción crear relaciones el sistema permite visualizar los documentos antes de realizar la relación, (ver imagen 5), igualmente para inscripción parcial, (ver imagen 6)

Imagen 3

The screenshot shows a software interface titled 'Mesa de control'. It includes a search section for 'Listar turnos por clasificación' (List turns by classification) with fields for 'Turno desde' (Turn from) and 'Turno hasta' (Turn until). Below is a 'Resultado de la búsqueda' (Search result) section with a 'Número del turno' (Turn number) field set to '2020-999-6-' and a 'Resultado' (Result) button. There are buttons for 'Aceptar' (Accept), 'Calificación' (Grade), 'Crear Relaciones' (Create Relations), and a red box highlighting 'Ver Imprimible' (View printable). The 'NOTAS' (NOTES) section contains a 'Notas Informativas' (Informative Notes) field with a 'Notas Informativas' (Informative Notes) button. Other sections include 'Busqueda de turnos por rangos' (Search turns by range) and 'NOTAS' (NOTES).

Imagen 4

Mesa de control

Listar turnos por clasificación

Busqueda de turnos por rangos  
Turno desde: 2020-999-6  
Turno hasta: 2020-999-6

Consultar

Resultado de la búsqueda  
Número del turno Resultado

Respuesta: INSCRIPCION\_PARCIAL Aceptar Calificación Crear Relaciones Ver Formulario Ver Nota

NOTAS

Notas Informativas

Notas Informativas

Imagen 5

Crear una Relación

Datos Básicos

Proceso: REGISTRO  
Fase: MESA CONTROL  
Relación: RELACION DOCUMENTOS INSCRITOS PARA FIRMA

Turnos Disponibles

	Número de Turno	Fecha de Radicación	Número de Documento	Visualizar
<input type="checkbox"/>	2016-999-6-58	31/03/2016	02114	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>
<input type="checkbox"/>	2016-999-6-274	09/11/2016	03333	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>
<input type="checkbox"/>	2016-999-6-16	06/02/2019	12221	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>
<input type="checkbox"/>	2020-999-6-107	12/05/2020	14141	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>
<input type="checkbox"/>	2020-999-6-187	16/05/2020	234567	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>
<input type="checkbox"/>	2020-999-6-234	22/05/2020	840414	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>
<input type="checkbox"/>	2020-999-6-235	22/05/2020	234567	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>

Total de Turnos: 7

Aceptar Volver

Imagen 6

Crear una Relación

Datos Básicos

Proceso: REGISTRO  
Fase: MESA CONTROL  
Relación: RELACION DOCUMENTOS CON INSCRIPCION PARCIAL PARA FIRMA

Turnos Disponibles

	Número de Turno	Fecha de Radicación	Número de Documento	Visualizar
<input type="checkbox"/>	2016-999-6-63	01/04/2016	01	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>
<input type="checkbox"/>	2016-999-6-248	08/11/2016	765445	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>
<input type="checkbox"/>	2016-999-6-281	09/11/2016	6767656	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>
<input type="checkbox"/>	2018-999-6-8	22/02/2018	3	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>
<input type="checkbox"/>	2018-999-6-89	21/06/2018	23	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>
<input type="checkbox"/>	2020-999-6-240	27/05/2020	2345	<input type="button" value="Ver Formulario"/> <input type="button" value="Ver Nota"/>

Total de Turnos: 6

Aceptar Volver

4. En el rol Registrador el sistema permite elegir la impresora, (ver imagen 7) y generar una alerta si no es seleccionada ninguna impresora, (ver imagen 8).

Imagen 7

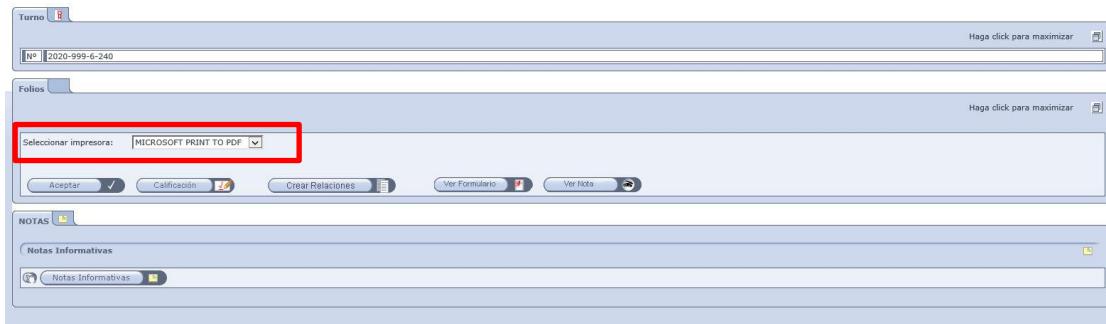
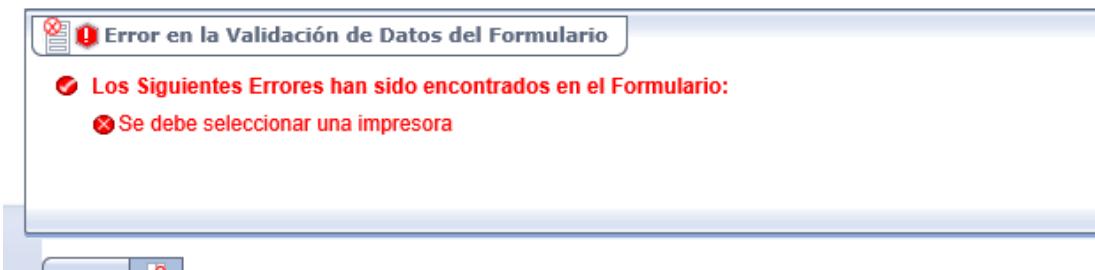


Imagen 8



5. En el rol registrador si el turno es de un proceso de constancia de inscripción (reproducción de sellos), al hacer clic en el botón “Vista previa formulario”, el sistema permite visualizar el formulario de constancia de inscripción correspondiente, (imagen 9).

Imagen 9

**REPRODUCCION DE CONSTANCIA DE INSCRIPCIÓN**

Página: 1 Impreso el 1 de Junio de 2020 a las 01:19:59 pm  
Con el turno 2020-999-6-248

Nro Matricula: 999-5671

CIRCULO DE REGISTRO: 999 CIRCULO\_PRUEBAS No. Catastro:  
MUNICIPIO: PRUEBAS DEPARTAMENTO: PRUEBAS VEREDA: PRUEBAS TIPO PREDIO: RURAL

**DIRECCION DEL INMUEBLE**

999-11

ANOTACIÓN: Nro: 1 Fecha 29/3/2007 Radicación 2007-999-6-349  
DOC: ESCRITURA 10 DEL 16/3/2007 NOTARIA SEGUNDA DE TUNJA VALOR ACTO: \$ 0  
ESPECIFICACION: MODO DE ADQUISICIÓN : 0172 TRANSFERENCIA DE DOMINIO DE BIENES FISCALES - POR  
LEGALIDAD - OBLIGACION ESPECIAL DE ENAJENAR EL INMUEBLE DESPUES DE TRANSCURRIDOS CINCO AÑOS  
PERSONAS QUE INTERVIENEN EN EL ACTO (X-Titular de derecho real del dominio, I-Titular de dominio incompleto)  
DE: ABBEY ROAD RECORDS NIT# 4123456  
A: STAR PR RINGO CC# 4123456 X

ANOTACIÓN: Nro: 2 Fecha 29/3/2007 Radicación 2007-999-6-349  
DOC: ESCRITURA 10 DEL 16/3/2007 NOTARIA SEGUNDA DE TUNJA VALOR ACTO: \$ 0  
ESPECIFICACION: LIMITACION AL DOMINIO : 0311 CONDICION RESOLUTORIA EXPRESA  
PERSONAS QUE INTERVIENEN EN EL ACTO (X-Titular de derecho real del dominio, I-Titular de dominio incompleto)  
DE: ABBEY ROAD RECORDS NIT# 4123456  
A: STAR PR RINGO CC# 4123456

**SUPERINTENDENCIA DE NOTARIADO Y REGISTRO**  
*la guarda de la fe pública*

6. En el rol Oficio de pertenencia el sistema genera el oficio de pertenencia con la respectiva firma del registrador, (ver imagen 9)

Imagen 9

 <p><b>CERTIFICACION PARA PROCESO DE PERTENENCIA</b></p> <p>FÁCILES Y EFICIENTES PARA QUE LOS PROGRAMADORES LOS USEN, PORQUE SON MÁS CERCANOS AL LENGUAJE NATURAL RESPECTO DEL LENGUAJE DE MÁQUINA 3&amp;#420;. LOS LENGUAJES DE ALTO NIVEL SE TRADUCEN A LENGUAJE DE MÁQUINA UTILIZANDO UN COMPILADOR O UN INTÉPRETE, O BIEN UNA COMBINACIÓN DE AMBOS. EL SOFTWARE TAMBIÉN PUEDE ESTAR ESCRITO EN LENGUAJE ENSAMBLADOR, QUE ES DE BAJO NIVEL Y TIENE UNA ALTA CORRESPONDENCIA CON LAS INSTRUCCIONES DE LENGUAJE MÁQUINA. SE TRADUCE AL LENGUAJE DE LA MÁQUINA UTILIZANDO UN ENSAMBLADOR.</p> <p>EL ANGLOISMO SOFTWARE ES EL MÁS AMPLIAMENTE DIFUNDIDO AL REFERIRSE A ESTE CONCEPTO, ESPECIALMENTE EN LA JERGÁ TÉCNICA, EN TANTO QUE EL TÉRMINO SINÓNIMO «LOGÍCAL», DERIVADO DEL TÉRMINO FRANCÉS LOGICIEL, ES UTILIZADO MAYORITARIAMENTE EN PAÍSES Y ZONAS DE INFLUENCIA FRANCESA SE CONOCE COMO SOFTWARE 1&amp;#6420;. O LOGICAL AL SOPORTE LÓGICO DE UN SISTEMA INFORMATICO, QUE COMPRENDE EL CONJUNTO DE LOS COMPONENTES LÓGICOS NECESARIOS QUE HACEN POSIBLE LA REALIZACIÓN DE TAREAS ESPECÍFICAS, EN CONTRAPOSICIÓN A LOS COMPONENTES FÍSICOS QUE SON LLAMADOS HARDWARE. LA INTERACCIÓN ENTRE EL SOFTWARE Y EL HARDWARE HACE OPERATIVO UN ORDENADOR (U OTRO DISPOSITIVO), ES DECIR, EL SOFTWARE ENVÍA INSTRUCCIONES QUE EL HARDWARE EJECUTA, HACIENDO POSIBLE SU FUNCIONAMIENTO.</p> <p>LOS COMPONENTES LÓGICOS INCLUYEN, ENTRE MUCHOS OTROS, LAS APLICACIONES INFORMÁTICAS, TALES COMO EL PROCESADOR DE TEXTO, QUE PERMITE AL USUARIO REALIZAR TODAS LAS TAREAS CONCERNIENTES A LA EDICIÓN DE TEXTOS; EL LLAMADO SOFTWARE DE SISTEMA, TAL COMO EL SISTEMA OPERATIVO, QUE BÁSICAMENTE PERMITE AL RESTO DE LOS PROGRAMAS FUNCIONAR ADECUADAMENTE, FACILITANDO TAMBÉN LA INTERACCIÓN ENTRE LOS COMPONENTES FÍSICOS Y EL RESTO DE LAS APLICACIONES, Y PROPORCIONANDO UNA INTERFAZ CON EL USUARIO 2&amp;#6420;.</p> <p>EL SOFTWARE EN SU GRAN MAYORÍA, ESTÁ ESCRITO EN LENGUAJES DE PROGRAMACIÓN DE ALTO NIVEL, YA QUE SON MÁS FÁCILES Y EFICIENTES PARA QUE LOS PROGRAMADORES LOS USEN, PORQUE SON MÁS CERCANOS AL LENGUAJE NATURAL RESPECTO DEL LENGUAJE DE MÁQUINA 3&amp;#420;. LOS LENGUAJES DE ALTO NIVEL SE TRADUCEN A LENGUAJE DE MÁQUINA UTILIZANDO UN COMPILADOR O UN INTÉPRETE, O BIEN UNA COMBINACIÓN DE AMBOS. EL SOFTWARE TAMBIÉN PUEDE ESTAR ESCRITO EN LENGUAJE ENSAMBLADOR, QUE ES DE BAJO NIVEL Y TIENE UNA ALTA CORRESPONDENCIA CON LAS INSTRUCCIONES DE LENGUAJE MÁQUINA. SE TRADUCE AL LENGUAJE DE LA MÁQUINA UTILIZANDO UN ENSAMBLADOR.</p> <p>EL ANGLOISMO SOFTWARE ES EL MÁS AMPLIAMENTE DIFUNDIDO AL REFERIRSE A ESTE CONCEPTO, ESPECIALMENTE EN LA JERGÁ TÉCNICA, EN TANTO QUE EL TÉRMINO SINÓNIMO «LOGÍCAL», DERIVADO DEL TÉRMINO FRANCÉS LOGICIEL, ES UTILIZADO MAYORITARIAMENTE EN PAÍSES Y ZONAS DE INFLUENCIA FRANCESA.</p> <hr/> <p><i>Bill Clinton</i></p> <p>REGISTRADOR DE INSTRUMENTOS PÚBLICOS</p>
--

	<b>DESPLIEGUE 7.4.6</b> <b>14/05/2020</b>	
<b>OC 2400 FIRMA FORMULARIOS REGISTRADOR</b>		

Actualmente en el aplicativo SIR, los imprimibles de los formularios de calificación, corrección, hoja de testamento, constancia de inscripción (reproducción de sellos) y notas devolutivas se generan cuando el turno llega al rol Calificador y al rol Responsable de correcciones respectivamente, para luego ser aprobados y firmados por el Registrador.

Por lo anterior, se incluye una vista previa del formulario con la firma digital del Registrador cuando el turno de los procesos de calificación, hoja de testamento, constancia de inscripción y notas devolutivas lleguen a la fase de registro en el rol Registrador y para el proceso de corrección cuando el turno llegue a la fase de revisar y aprobar en el rol Responsable de Correcciones.

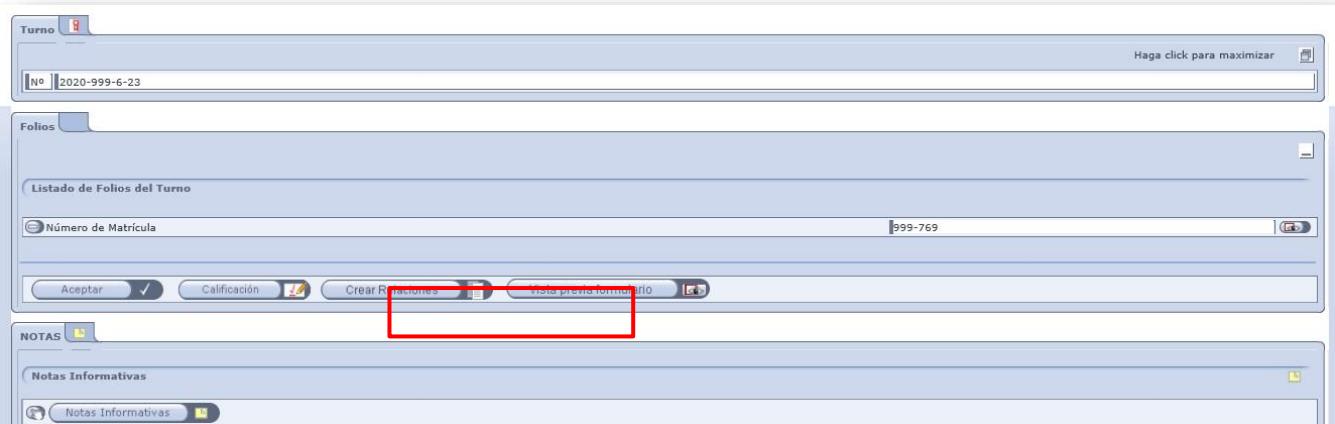
De esta manera el rol Registrador y Responsable de Correcciones pueden observar cada uno de los documentos para poder aceptarlos e imprimirlas y posteriormente ser entregados al ciudadano.

4. En el Rol Registrador, en la fase “Registro” (Ver imagen 1), el sistema permite visualizar un botón llamado “Vista previa formulario” para mostrar el imprimible del formulario. Se visualiza en imagen 2.

Imagen 1



Imagen 2



5. Si el turno es de un proceso de calificación, al hacer clic en el botón “Vista previa formulario”, el sistema permite visualizar el formulario de calificación correspondiente. Se visualiza en imagen 3.

Imagen 3

**SNR** SUPERINTENDENCIA DE NOTARIADO Y REGISTRO DE LA PROVINCIA DE SANTA CRUZ

**FORMULARIO DE CALIFICACION  
CONSTANCIA DE INSCRIPCION**

Página: 1  
Impreso el 23 de Abril de 2020 a las 12:48:34 pm

Con el turno 2020-999-6-87 se calificaron las siguientes matriculas:  
999-767

Nro Matricula: **999-767**  
CIRCULO DE REGISTRO: 999 CIRCULO\_PRUEBAS No. Catastro: 001-001-147  
MUNICIPIO: PRUEBAS DEPARTAMENTO: PRUEBAS VEREDA: PRUEBAS TIPO PREDIO: RURAL

=====

**DIRECCION DEL INMUEBLE**  
1) SIN DIRECCION .

ANOTACION: Nro: 7 Fecha 23/4/2020 Radicación 2020-999-6-87  
DOC. ESCRITURA 76 DEL: 14/4/2020 PERSONERIA MUNICIPAL DE LA MERCED VALOR ACTO: \$ 0  
ESPECIFICACION: MODO DE ADQUISICION : 0126 COMPRAVENTA PARCIAL  
PERSONAS QUE INTERVIENEN EN EL ACTO (X-titular de derecho real del dominio, I-titular de dominio incompleto)  
A: MORENO YAMILÉ CC# 12345

**SUPERINTENDENCIA DE NOTARIADO Y REGISTRO**  
FIN DE ESTE DOCUMENTO  
El Interesado debe comunicar cualquier falta o error en el registro de los documentos  
La guarda de la fe pública

[Redacted area]

Adicionalmente, después de revisar y estar de acuerdo con lo escrito en el formulario de calificación, al hacer clic en el botón “Aceptar”, se imprime el formulario con la firma correspondiente (ver imagen 4).

Imagen 4

Turno [Redacted]

Haga click para maximizar [Maximize]

2020-999-6-23

Folios

Listado de Folios del Turno

Número de Matrícula: 999-769

Aceptar [Redacted] Calificación [Redacted] Crear Relaciones [Redacted] Vista previa formulario [Redacted]

NOTAS

Notas Informativas

Notas Informativas

Imagen 5

**SNR** SUPERINTENDENCIA DE NOTARIADO Y REGISTRO  
En el servicio de la justicia

**FORMULARIO DE CALIFICACION  
CONSTANCIA DE INSCRIPCION**

Página: 1  
Impreso el 23 de Abril de 2020 a las 12:46:34 pm

Con el turno 2020-999-6-87 se calificaron las siguientes matriculas:  
999-767

Nro Matricula: **999-767**  
CIRCULO DE REGISTRO: 999 CIRCULO\_PRUEBAS No. Catastro: 001-001-147  
MUNICIPIO: PRUEBAS DEPARTAMENTO: PRUEBAS VEREDA: PRUEBAS TIPO PREDIO: RURAL

---

**DIRECCION DEL INMUEBLE**  
1) SIN DIRECCION.

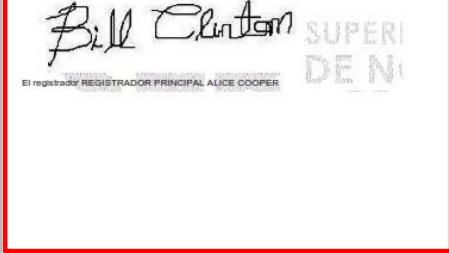
ANOTACION: Nro: 7 Fecha 23/4/2020 Radicación 2020-999-6-87  
DOC: ESCRITURA 76 DEL: 14/4/2020 PERSONERIA MUNICIPAL DE LA MERCED VALOR ACTO: \$ 0  
ESPECIFICACION: MODO DE ADQUISICION : 0126 COMPRAVENTA PARCIAL  
PERSONAS QUE INTERVIENEN EN EL ACTO (X-Titular de derecho real del dominio, I-Titular de dominio incompleto)  
A: MORENO YAMILE CCI# 12345

**SUPERINTENDENCIA DE NOTARIADO Y REGISTRO**  
La guarda de la fe pública

**FIN DE ESTE DOCUMENTO**  
El interesado debe comunicar cualquier falta o error en el registro de los documentos

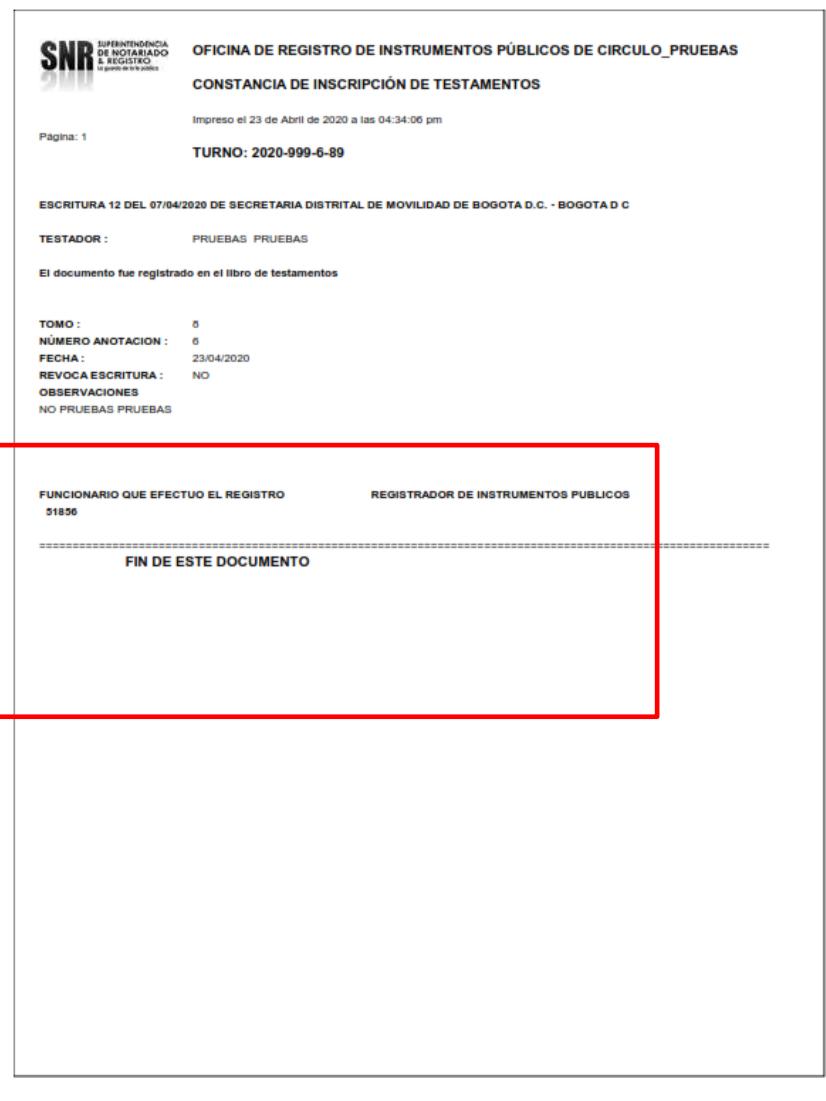
USUARIO: STSSS Impreso por: STSSS  
TURNO: 2020-999-1-95 FECHA: 22/4/2020  
NIS: UKew3g4b1Zm8kII09kBNOTLbjm1+qXbz8pYoaAUB7/s=  
Verificar en: <http://192.168.76.54:8190/WS-SIRClient/>  
EXPEDIDO EN: CIRCULO\_PRUEBAS

*Bill Clinton* SUPERINTENDENCIA DE NOTARIADO Y REGISTRO  
El registrador REGISTRADOR PRINCIPAL ALICE COOPER



- Si el turno es de un proceso de hoja testamento, al hacer clic en el botón “Vista previa formulario”, el sistema permite visualizar la hoja de testamento correspondiente. Se visualiza en imagen 6.

Imagen 6



Adicionalmente, después de revisar y estar de acuerdo con lo escrito en la hoja de testamento, al hacer clic en el botón “Aceptar” (ver imagen 4), el sistema permite imprimir el formulario con la firma correspondiente (ver imagen 7).

## Imagen 7

**SNR** DIRECCION DE NOTARIADO  
OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE CIRCULO\_PRUEBAS  
CONSTANCIA DE INSCRIPCIÓN DE TESTAMENTOS

Impreso el 23 de Abril de 2020 a las 04:34:06 pm  
Pagina: 1 TURNO: 2020-999-6-89

ESCRITURA 12 DEL 07/04/2020 DE SECRETARIA DISTRITAL DE MOVILIDAD DE BOGOTA D.C. - BOGOTA D.C.

TESTADOR : PRUEBAS PRUEBAS  
El documento fue registrado en el libro de testamentos

TOMO : 5  
NUMERO ANOTACION : 6  
FECHA : 23/04/2020  
REVOCAR ESCRITURA : NO  
OBSERVACIONES :  
NO PRUEBAS PRUEBAS

FUNCIONARIO QUE EFECTUO EL REGISTRO REGISTRADOR DE INSTRUMENTOS PUBLICOS  
51856

FIN DE ESTE DOCUMENTO

USUARIO ID: 51856 Impreso por: 51856  
USUARIO: 2020-999-6-89 FECHA: 22/4/2020  
URL: http://192.168.76.54:8190/WS-SIRClient/  
Verificar en: http://192.168.76.54:8190/WS-SIRClient/  
EXPEDIDO EN: CIRCULO\_PRUEBAS

*Bill Clinton* SUPERINTENDENTE DE NOTARIADO  
El registrador REGISTRADOR PRINCIPAL ALICE COOPER

7. Si el turno es de un proceso de constancia de inscripción (reproducción de sellos), al hacer clic en el botón “Vista previa formulario”, el sistema debe permitir visualizar el formulario de constancia de inscripción correspondiente. Se visualiza en imagen 8 y 9.

Imagen 8

**REPRODUCCION DE CONSTANCIA DE INSCRIPCION**

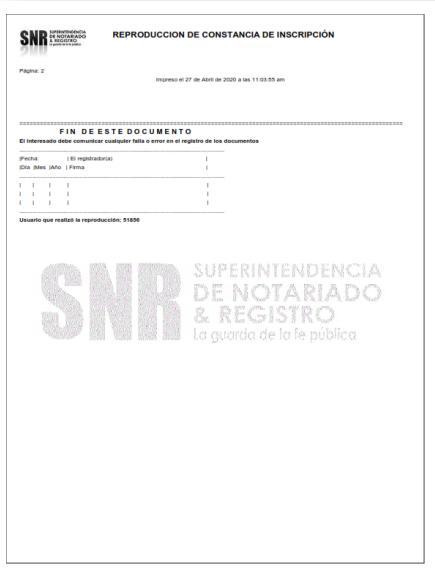
Página: 1 Impreso el 27 de Abril de 2020 a las 11:03:55 am  
Con el turno 2020-999-6-91

Nro Matricula: 999-545  
CIRCULO DE REGISTRO: 999 CIRCULO\_PRUEBAS No. Catastro: 00-000-000-111  
MUNICIPIO: PRUEBAS DEPARTAMENTO: PRUEBAS VEREDA: PRUEBAS TIPO PREDIO: RURAL

DIRECCION DEL INMUEBLE  
1) SIN DIRECCION : EL PINO

ANOTACION Nro: 1 Fecha: 07/04/2020 Radicacion: 0  
DOC. ESCRITURA: 175 CED: 8951501 NOTARIA DE LEIVA VALOR ACTO: \$ 600  
ESPECIFICACION: FALSA TRADICION: 010 COMPRA-VENTA DERECHOS Y ACCIONES DE SUTRAGO ROBLES CANDIDA  
PERSONAS QUE INTERVIENEN EN EL ACTO (X)Titular de derecho real del dominio, (T)titular de dominio incompleto  
DE APELLIDO CIUDADANO PRUEBAS - 5956130 APellido Ciudadano PRUEBAS - 5956130 NOMBRE CIUDADANO PRUEBAS - 5956130  
A: APELLIDO CIUDADANO PRUEBAS - 5956130 APellido Ciudadano PRUEBAS - 5956130 NOMBRE CIUDADANO PRUEBAS - 5956130  
5956130 X  
ANOTACION Nro: 2 Fecha: 07/04/2020 Radicacion: 0  
DOC. ESCRITURA: 175 CED: 8951501 NOTARIA DE LEIVA VALOR ACTO: \$ 600  
ESPECIFICACION: FALSA TRADICION: 010 COMPRA-VENTA DERECHOS Y ACCIONES DE SUTRAGO ROBLES CANDIDA  
PERSONAS QUE INTERVIENEN EN EL ACTO (X)Titular de derecho real del dominio, (T)titular de dominio incompleto  
DE APELLIDO CIUDADANO PRUEBAS - 5956685 APellido Ciudadano PRUEBAS - 5956685 NOMBRE CIUDADANO PRUEBAS - 5956685  
A: APELLIDO CIUDADANO PRUEBAS - 5956685 APellido Ciudadano PRUEBAS - 5956685 NOMBRE CIUDADANO PRUEBAS - 5956685  
5956685 X  
A: APELLIDO CIUDADANO PRUEBAS - 5956688 APellido Ciudadano PRUEBAS - 5956688 NOMBRE CIUDADANO PRUEBAS - 5956688

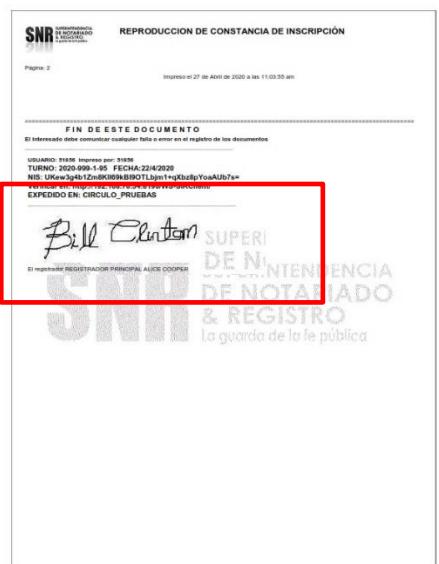
Imagen 9



Adicionalmente, después de revisar y estar de acuerdo con lo escrito en la constancia de inscripción, al hacer clic en el botón “Aceptar” (ver imagen 4), el sistema permite generar la impresión del formulario con la firma correspondiente (ver imagen 11).

## Imagen 11

REPRODUCCION DE CONSTANCIA DE INSCRIPCION			
Página: 1	Impreso el 27 de Abril de 2020 a las 11:03:55 am		
Con el turno 2020-999-6-91			
<b>Nro Matricula: 999-545</b>			
CIRCULO DE REGISTRO: 999 CIRCULO_PRUEBAS		No. Catalogo: 00-0-003-111	
MUNICIPIO: PRUEBAS DEPARTAMENTO: PRUEBAS		VEREDA: PRUEBAS TIPO PREDIO: RURAL	
<hr/> <b>DIRECCION DEL INMUEBLE</b> <hr/>			
11 SIN DIRECCION: EL PINO			
<hr/> <b>ANOTACION: Nro: 1 Fecha: 18/11/1952 Radicacion: 8</b>			
DOC.ESCRITURA: 270 DEL: 19/12/1951 NOTARIA DE LEIVA VALOR ACTO: \$ 600			
ESPECIFICACION: FALSA TRADICION : 610 COMPRA-VENTA DERECHOS Y ACCIONES DE BUTRAGO ROBLES CANIDA			
PERSONAS QUE INTERVIENEN EN EL ACTO (X)titular de derecho real del dominio, I-titular de dominio incompleto)			
DE: APELLIDO CIUDADANO PRUEBAS - 59561336 APPELLIDOZ CIUDADANO PRUEBAS - 59561336 NOMBRE CIUDADANO PRUEBAS - 59561336			
A: APELLIDO CIUDADANO PRUEBAS - 59561337 APPELLIDOZ CIUDADANO PRUEBAS - 59561337 NOMBRE CIUDADANO PRUEBAS - 59561337			
<hr/> <b>ANOTACION: Nro: 2 Fecha: 5/9/1955 Radicacion: 6</b>			
DOC.ESCRITURA: 175 DEL: 10/09/1955 NOTARIA DE LEIVA VALOR ACTO: \$ 600			
ESPECIFICACION: FALSA TRADICION : 610 COMPRA-VENTA DERECHOS Y ACCIONES ESCRITURA: 17/05			
PERSONAS QUE INTERVIENEN EN EL ACTO (X)titular de derecho real del dominio, I-titular de dominio incompleto)			
DE: APELLIDO CIUDADANO PRUEBAS - 59564434 APPELLIDOZ CIUDADANO PRUEBAS - 59564434 NOMBRE CIUDADANO PRUEBAS - 59564434			
A: APELLIDO CIUDADANO PRUEBAS - 59564435 APPELLIDOZ CIUDADANO PRUEBAS - 59564435 NOMBRE CIUDADANO PRUEBAS - 59564435			
<hr/> <b>ANOTACION: Nro: 3 Fecha: 18/4/1990 Radicacion: 8699</b>			
DOC.ESCRITURA: 340 DEL: 7/9/1990 NOTARIA DE LEIVA VALOR ACTO: \$ 500.000			
ESPECIFICACION: FALSA TRADICION : 610 COMPRA-VENTA DERECHOS Y ACCIONES ESCRITURA: 17/05			
PERSONAS QUE INTERVIENEN EN EL ACTO (X)titular de derecho real del dominio, I-titular de dominio incompleto)			
DE: APELLIDO CIUDADANO PRUEBAS - 59566854 APPELLIDOZ CIUDADANO PRUEBAS - 59566854 NOMBRE CIUDADANO PRUEBAS - 59566854			
A: APELLIDO CIUDADANO PRUEBAS - 59566855 APPELLIDOZ CIUDADANO PRUEBAS - 59566855 NOMBRE CIUDADANO PRUEBAS - 59566855			
X			



8. Si el turno es de un proceso de notas devolutivas, al hacer clic en el botón “Vista previa formulario”, el sistema debe permitir visualizar el formulario de nota devolutiva correspondiente. Se visualiza en imagen 12 y 13.

Imagen 12

**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE CÍRCULO\_PRUEBAS**

**NOTA DEVOLUTIVA**

Página: 1  
Impreso el 23 de Abril de 2020 a las 12:28:32 pm

El documento ESCRITURA Nro 76 del 14-04-2020 de PERSONERIA MUNICIPAL DE LA MERCEDE - CALDAS fue presentado para su inscripción como solicitud de registro de documentos con Radicación:2020-999-6-84 vinculado a la Matrícula Inmobiliaria: 999-767

**Y CERTIFICADOS ASOCIADOS: 2020-999-1-17**

Conforme con el principio de Legalidad previsto en el literal d) del artículo 3 y en el artículo 23 de la Ley 1579 de 2013 (Estatuto de Registro de Instrumentos Públicos) se admite y por lo tanto se devuelve sin registrar el presente documento por las siguientes razones y fundamentos de derecho:

**1. OTROS**

UNA VEZ SUBSANADA(S) LA(S) CAUSA(ES) QUE MOTIVÓ LA NEGATIVA DE INSCRIPCIÓN, FAVOR RADICAR NUEVAMENTE EN ESTA OFICINA EL DOCUMENTO PARA SU CORRESPONDIENTE TRAMITE, ADJUNTANDO LA PRESENTE NOTA DEVOLUTIVA.

CUANDO LA CAUSAL O LAS CAUSALES QUE RECHAZA(AN) LA INSCRIPCIÓN DEL DOCUMENTO NO SEA(S) SUSANABLE(S), SE CONFIERE EL PAGO DE LO QUE SE INDIQUE EN EL ACTO ADMINISTRATIVO DE RECHAZO Y SE DEBERÁ SOLICITAR LA DEVOLUCIÓN DEL DOCUMENTO PARA LOS DIFECHOS DE REGISTRO ES Dicho(S) MES(es) DE CALENDARIO, CONTADOS A PARTIR DE LA FECHA DE RECIBIDA(MA) DEL ACTO ADMINISTRATIVO QUE NEGA EL REGISTRO O SE DESANIE EL DOCUMENTO INSERITO.

PARA SOLICITAR LA DEVOLUCIÓN DE DINERO POR CONCEPTO DE IMPUESTO DE REGISTRO, DEBE DIRIGIRSE A LA GERENCIACIÓN DE DEPARTAMENTO DE I.G.C.D. EN EL CASO DE QUE EL PAGO SE HAGA POR CONCEPTO DE IMPUESTO DE REGISTRO, SE DEBERÁ PAGARLO AL DECRETO 140 DE 1998.

LOS ACTOS DE NEGOCIOS JURIDICOS A QUÉ SE REFIERE EL ARTICULO 17 DE LA LEY 1579 DE 2013, DEBERÁN PRESENTARSE PARA SU INSCRIPCION, DENTRO DE LOS DOS (2) MESES CALENDARIOS SIGUIENTES A LA FECHA DE SU OTORGAMIENTO PARA ACTOS NOTARIALES O LA FECHA DE EJECUTORIA PARA ACTOS DE AUTORIZACIONES, EXCEPTO LOS CASOS RELACIONADOS CON EL ACTO DE CONSTITUCIÓN DE PATRIMONIO DE LOS DIFECHOS DE REGISTRO, EN EL CASO DE QUE SE PRESENTEN DICHOS ACTOS DE AUTORIZACIONES, DEBERÁN PRESENTARSE DENTRO DE LOS DIFECHOS DE REGISTRO, DENTRO DE LOS DIFECHOS SIGUIENTES A SU NOTIFICACIÓN, UNA RETIRO DE LO PREVETO POR EL NUMERAL DOS (2) DEL ARTICULO 21 DEL DECRETO 140 DE 1998, EN EL CASO DE QUE SE PRESENTEN DIFERENTES TIPOS DE ACTOS DE AUTORIZACIONES, SE DEBERÁN PRESENTARLOS DENTRO DE LOS DIFECHOS DE REGISTRO, DENTRO DE LOS DIFECHOS SIGUIENTES A SU NOTIFICACIÓN, UNA RETIRO DE LO PREVETO POR EL NUMERAL DOS (2) DEL ARTICULO 21 DEL DECRETO 140 DE 1998.

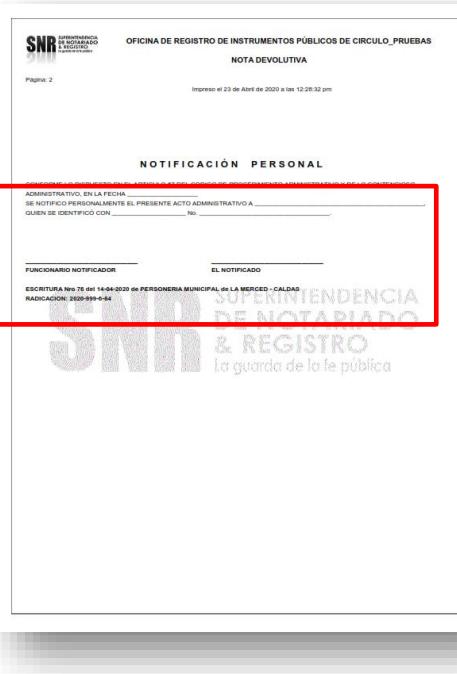
EXCEPTUENSE DE LO ANTERIOR, LOS CASOS RELACIONADOS CON EL NEGOCIO JURIDICO DE HIPOTECA Y EL ACTO DE CONSTITUCIÓN DE PATRIMONIO DE LOS DIFECHOS DE REGISTRO, EN EL CASO DE QUE SE PRESENTEN DICHOS ACTOS DE AUTORIZACIONES, DEBERÁN PRESENTARSE DENTRO DE LOS DIFECHOS DE REGISTRO, DENTRO DE LOS DIFECHOS SIGUIENTES A SU NOTIFICACIÓN, UNA RETIRO DE LO PREVETO POR EL NUMERAL DOS (2) DEL ARTICULO 21 DEL DECRETO 140 DE 1998.

CONTRA EL PRESENTE ACTO ADMINISTRATIVO, PROcede EL RECURSO DE REPOSICIÓN ANTE EL REGISTRADOR DE INSTRUMENTOS PÚBLICOS Y EN SUBSIDIO EL DE APELACIÓN ANTE LA SUBSECCIÓN DE APOYO JURIDICO REGISTRAL DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO DENTRO DE LOS DIFECHOS DE REGISTRO, DENTRO DE LOS DIFECHOS SIGUIENTES A SU NOTIFICACIÓN, UNA RETIRO DE LO PREVETO POR EL NUMERAL DOS (2) DEL ARTICULO 21 DEL DECRETO 140 DE 1998.

REGISTRADOR DE INSTRUMENTOS PÚBLICOS  
51856

FIN DE ESTE ACTO ADMINISTRATIVO

Imagen 13



Adicionalmente, después de revisar y estar de acuerdo con lo escrito en el formulario de nota devolutiva, al hacer clic en el botón “Aceptar” (ver imagen 4), el sistema permite imprimir el formulario con la firma correspondiente (ver imagen 15).

Imagen 15

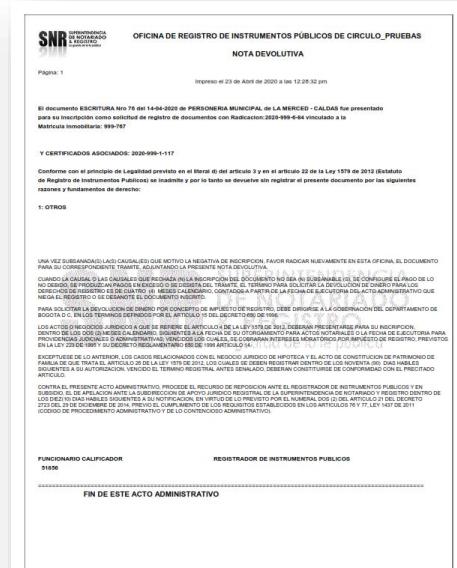
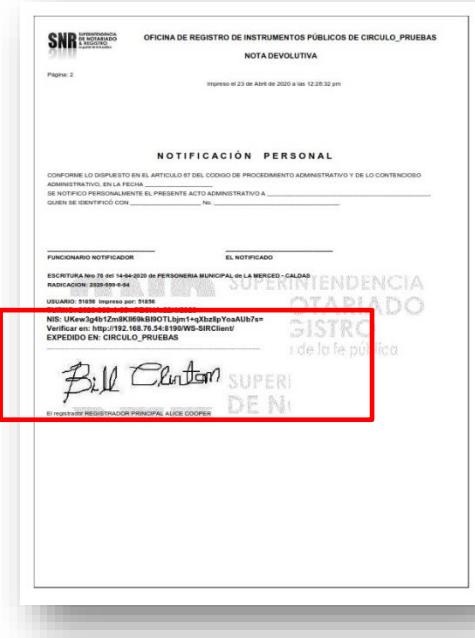


Imagen 16



9. En el Rol Responsable de Correcciones, en la fase “Revisar Aprobar” (Ver imagen 17), el sistema permite visualizar un botón llamado “Vista previa formulario” para mostrar el imprimible. Se visualiza en imagen 18.

Imagen 17

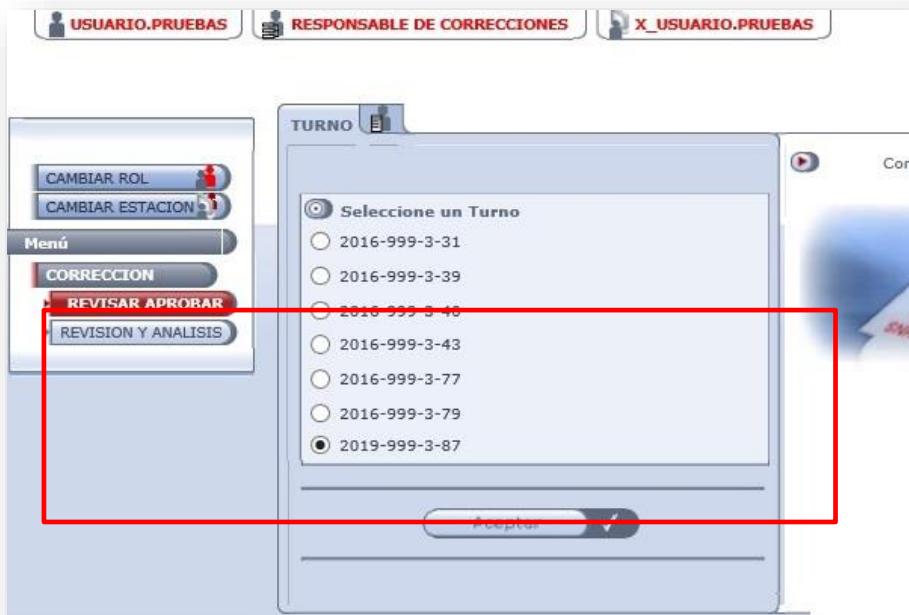


Imagen 18

Correo

Dirección de Correo

Derecho de Petición  
 NO

Corrección solicitada con anterioridad  
 NO

Interés Jurídico  
 NO

Datos Solicitante

Tipo de Identificación	CEDULA	Número	123456
Primer Apellido	OJO	Segundo Apellido	HNI
Nombre	GH	Teléfono	

Ver Cambios

Ver Alertas Consultar

Deshacer Cambios Aprobar y Guardar  Vista previa formulario Devolver Auxiliar

10. Si el turno es de un proceso de corrección, al hacer clic en el botón “Vista previa formulario”, el sistema debe permitir visualizar el formulario de corrección correspondiente. Se visualiza en imagen 19.

Imagen 19

SNR SISTEMAS DE NOTARIADO RENOVANDO EL PUEBLO

Página: 1 Impreso el 24 de Abril de 2020 a las 02:02:07 pm

Con el turno 2020-999-3-16 se corrigieron las siguientes matrículas:  
999-767

Nro Matrícula: 999-767  
CIRCULO DE REGISTRO: 999 CIRCULO PRUEBAS  
MUNICIPIO: PRUEBAS DEPARTAMENTO: PRUEBAS VEREDA: PRUEBAS TIPO PREDIO: RURAL  
FECHA APERTURA: 9/10/1990 RADICACION: 7206 CON ESCRITURA DE: 26/9/1990

=====

DIRECCION DEL INMUEBLE  
1) SIN DIRECCION.

SALVEDADES: (Información Anterior o Corregida)  
[51850]: Anotación Nro: 0 No. corrección: 1 Radicación: 2020-999-3-16 Fecha: 24/4/2020  
PRUEBAS PRUEBAS PRUEBAS PRUEBAS PRUEBAS PRUEBAS PRUEBAS PRUEBAS

ESTADIA DE NOTARIADO Y REGISTRO

FIN DE ESTE DOCUMENTO la guarda de la le publica  
El interesado debe comunicar cualquier falla o error en el registro de los documentos

(Fecha) | El registrador(a) |  
(Día | Mes | Año | Firma)


Adicionalmente, después de revisar y estar de acuerdo con lo escrito en el formulario de corrección, al hacer clic en el botón “Aprobar y Guardar” (ver imagen 20), el sistema permite generar la impresión del formulario con la firma correspondiente.

Imagen 20

The screenshot shows a software application window titled 'Correo'. The interface includes sections for 'Dirección de Correo', 'Derecho de Petición', 'Corrección solicitada con anterioridad', 'Interés Jurídico', and 'Datos Solicitante'. In the 'Datos Solicitante' section, there are fields for 'Tipo de Identificación' (CEDULA), 'Número' (123456), 'Primer Apellido' (OJO), 'Segundo Apellido' (HNI), and 'Nombre' (GH). At the bottom, there is a toolbar with buttons for 'Ver Alertas', 'Consultar', 'Deshacer Cambios', 'Aprobar y Guardar' (which is highlighted with a red box), 'Vista previa formulario', and 'Devolver Auxiliar'.

Imagen 22



11. En el Rol Registrador, después de crear una relación, al hacer clic en el botón “Finalizar” (ver imagen 23), el sistema permite visualizar generar la impresión de los formularios con la firma correspondiente.

Imagen 23



DESPLIEGUE 7.4.5

20/02/2020



#### RECIBO MAYOR VALOR- validación anotaciones temporales por número de Radicación

1. El sistema permite realizar la reimpresión de recibos por pantallas administrativas / reimpresión de recibos, para la reimpresión del recibo de mayor valor aplicado a un turno de registro se modifica el encabezado del recibo generado en este proceso, imagen 1.

Imagen 1: Recibo Reimpresión Mayor Valor

**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS**  
**CIRCULO\_PRUEBAS - NIT: 899999007-0**  
**RECIBO DE PAGO DE MAYOR VALOR**  
Impreso el 24 de Abril de 2019 a las 10:23:56 am

**62483260**

**No. RADICACIÓN: 2019-999-6-64**

NOMBRE DEL SOLICITANTE: ANDRES BELTRAN  
CERTIFICADO No.: 1234543256 del 24/4/2019 PERSONERIA de SOACHA

**MATRICULAS: 999-9324**

VALOR DERECHOS: \$100.000

**VALOR TOTAL: \$102000.0**

**FORMA PAGO:**

TARJETA DEBITO VALOR: \$ 102.000

Usuario: 51856

Esta es una reimpresión del recibo 62483260

2. Se modificará por líneas de código y se incluirán las validaciones necesarias sobre las anotaciones, en la fase de firma registro, específicamente validar por código que exista un numero de radicación y este coincida con el número de turno que se encuentra en trámite, asociado a la anotación temporal que se realizó sobre el folio.

Imagen 2.

Imagen 2, anotación temporal, # radicación

Anotación: Nº 3 del Folio #999-9087			
Radicación	2019-999-6-54	Del	04/4/2019
Doc	DOCUMENTO 0987654	Del	04/4/2019
Oficina de Origen	PERSONERIA	De	SOACHA
Valor		Estado	VALIDA
Especificación	HIPOTECA DE DERECHOS Y ACCIONES	Naturaleza Jurídica	0207
Comentario		Anotación Temporal	

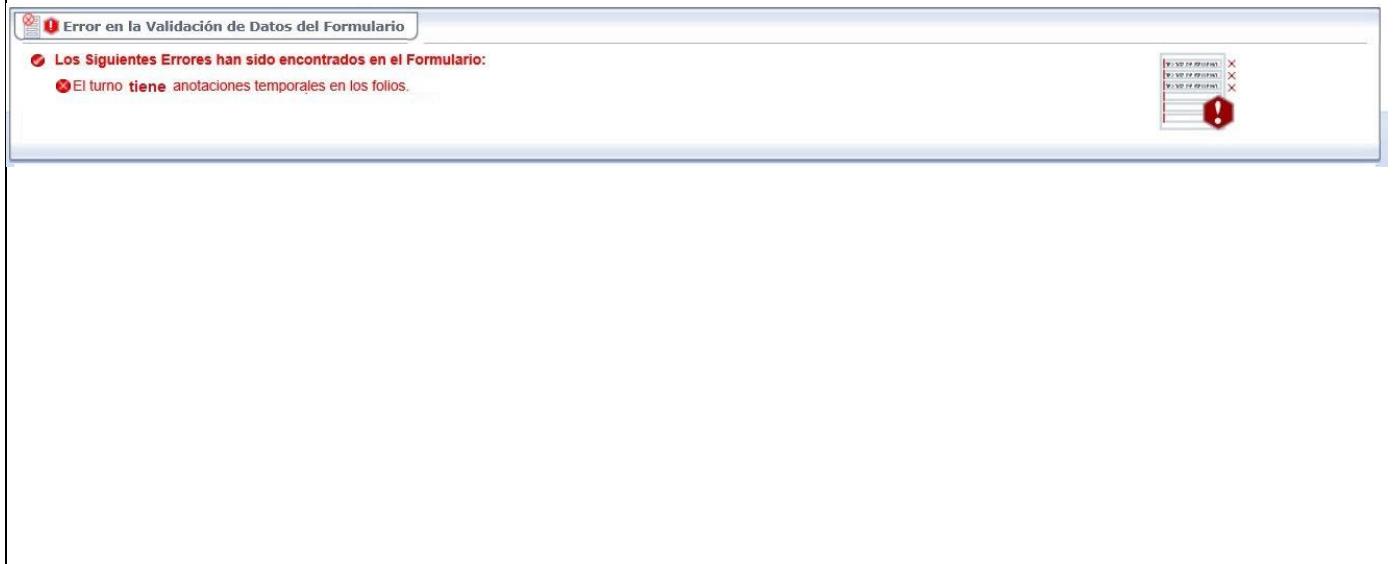
3. Se valida por código y se agrega una alerta que valide que; al pasar el turno de la fase; mesa de control registro a la fase; Registrador, este tenga una anotación temporal en alguno de los folios, siempre y cuando el turno se encuentre en estado Inscrito. Imagen 3.

Imagen 3, Alerta anotación temporal.



4. Se valida por código y se agrega una alerta cuando el turno se avance a la fase; certificados asociados y este no tenga las anotaciones en definitivo, no dejará imprimir los certificados ni avanzar el turno a la siguiente fase. Imagen 4.

Imagen 4, alerta anotación definitiva.



	<b>DESPLIEGUE 7.4.4</b> 24/10/2019	
<b>Solicitud 411379 - Suma de Certificados Asociados en recibo de Liquidación Reimpresión de los recibos de Liquidación</b>		

En el recibo de liquidación que genera el Sistema de Información Registral, el sistema permite realizar la suma del valor de los certificados asociados al total del valor de los derechos y conservación documental, se visualiza en imagen 1.

Imagen 1

**CONSTANCIA DE LIQUIDACIÓN**

Impreso el 6 de Septiembre de 2019 a las 01:38:49 pm

**No. DE LIQUIDACIÓN: L-2019-999-6-2772**

REGISTRADO POR: 51856

MATRICULA(S) 999-9747

SUBTIPO DE SOLICITUD: NORMAL

**VALOR DE DERECHOS DE REGISTRO: \$ 3.640.000**

**VALOR DE IMPUESTO: \$ 0**

**VALOR DE LA MULTA: \$ 0**

**CERTIFICADOS ASOCIADOS: \$ 16.800**

**Valor de conservación documental del 2%: \$ 72.800**

sumar

**VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS \$ 3.729.600**

**Total**

NOTA: ESTA CONSTANCIA TIENE VIGENCIA POR UN DÍA.

EL PROCESO DE REGISTRO EMPIEZA CON LA RADICACIÓN UNA VEZ EFECTUADO EL PAGO DE LOS DERECHOS

El sistema permite reimprimir el recibo de la última liquidación generada por el Rol Liquidador en el sistema desde la pantalla administrativa reimpresión de recibos por lo anterior se adiciona en la lista desplegable la opción “liquidación”. Se visualiza en imagen 2.

Imagen 2



Al seleccionar desde la lista desplegable la nueva opción liquidación, se carga la pantalla “Datos del turno para la reimpresión del recibo”, donde se evidencia el subtítulo “Liquidación” y se muestra el id de la última liquidación generada desde el sistema. Se visualiza en imagen 3.

Imagen 3



Se crea la lógica de reimpresión en el sistema, y en el imprimible se visualiza que es una reimpresión de una liquidación. Se visualiza en imagen 4.

Imagen 4

**CONSTANCIA DE LIQUIDACIÓN**  
Impreso el 6 de Septiembre de 2019 a las 01:38:49 pm  
**No. DE LIQUIDACIÓN: L-2019-999-6-2772**

REGISTRADO POR: 51856  
MATRICULA(S) 999-9747  
SUBTIPO DE SOLICITUD: NORMAL  
VALOR DE DERECHOS DE REGISTRO: \$ 3.640.000  
VALOR DE IMPUESTO: \$ 0  
VALOR DE LA MULTA: \$ 0  
CERTIFICADOS ASOCIADOS: \$ 16.800

Valor de conservación documental del 2%: \$ 72.800

**VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS \$ 3.729.600**

NOTA: ESTA CONSTANCIA TIENE VIGENCIA POR UN DÍA.

EL PROCESO DE REGISTRO EMPIEZA CON LA RADICACIÓN UNA VEZ EFECTUADO EL PAGO DE LOS DERECHOS

**Reimpresión del recibo de liquidación**



**DESPLIEGUE 7.4.3**

**29/08/2019**



**Solicitud Impresión número de Pin PSE en recibo**

El sistema permite generar recibos para sus diferentes procesos, los mismo ahora incluyen el número de PIN siempre y cuando este haya sido ingresado al momento de realizar la liquidación por medio de la pantalla liquidador.

**- Recibo de Registro**

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
CIRCULO\_PRUEBAS - NIT: 899999007-0  
SOLICITUD REGISTRO DOCUMENTOS  
Impreso el 29 de Agosto de 2018 a las 08:20:22 pm

**1236593054**

No. RADICACIÓN: **2018-999-6-524**

NOMBRE DEL SOLICITANTE: H

DOCUMENTO No.: 8885 del 29/8/2018 CONSULADO DE ESTADOS UNIDOS DE AMERICA de BOGOTA D. C.

**MATRICULAS: 999-8969**

ACTOS A REGISTRAR:

Tipo....	Código....	Cantidad....	Tipo Tarifa	Derecho	Impuesto
VENTA	1	400.000.000	N	\$	3.000.000

**TURNOS CERTIFICADOS ASOCIADOS: 1**

**VALOR MULTA: \$0**

**CANAL DE RECAUDO XXXXX**

**FORMA DE PAGO:**

CONSIGNACION\_SUPERGIROS: VALOR: \$ 1.000.000

CONSIG. BANCO: BANCO DE BOGOTA PIN: 8555855 FECHA CONSIG.: 29/08/2018 VALOR PAGADO:  
\$983.700 VALOR DOC.: \$1.500.000

Conservación documental del 2% \$ 60.000

- Recibo de Certificados

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
NIT: 899999007-0 CIRCULO\_PRUEBAS  
SOLICITUD CERTIFICADO DE TRADICION  
Impreso el 29 de Agosto de 2018 a las 08:20:30 pm

**1236593055**

No. RADICACIÓN: **2018-999-1-625**

Asociado al turno de registro: 2018-999-6-524

**MATRICULA: 999-8969**

NO SE EXPIDE DE INMEDIATO EL CERTIFICADO

NOMBRE DEL SOLICITANTE: H

CERTIFICADOS: 1

CANAL DE RECAUDO XXXXX

FORMA DE PAGO:

CONSIGNACIÓN\_SUPERGIROS: VALOR: \$ 1.000.000

CONSIG. BANCO: BANCO DE BOGOTA PIN: 8555855 FECHA CONSIG.: 29/08/2018 VALOR PAGADO:  
\$983.700 VALOR DOC.: \$1.500.000

**VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: \$ 16.300**

EL CERTIFICADO SE EXPIDE DE ACUERDO A LOS DATOS SUMINISTRADOS

Usuario: 51856

- Recibo pago Mayor Valor

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
CIRCULO\_PRUEBAS - NIT: 899999007-0  
RECIBO DE PAGO DE MAYOR VALOR  
Impreso el 22 de Abril de 2018 a las 05:04:20 am

**62483251**

No. RADICACIÓN: **2018-999-6-109**

NOMBRE DEL SOLICITANTE: HG SOLANO PRUEBAS SIR PRUEBAS V CC# 1111111  
TELÉFONO: 79261492

Canal de Recaudo: XXXXXXXXX

FORMA DE PAGO:

CONSIGNACION\_SUPERGIROS: VALOR: \$ 1.000.000  
CONSIG. BANCO: BANCO DE BOGOTÁ PIN: 8555855 FECHA CONSIG.: 29/08/2018 VALOR PAGADO:  
\$983.700 VALOR DOC.: \$1.500.000

Efectivo Valor: \$ 10.000

Valor Derechos: \$20.000

Valor Certificados: \$20.000

CONSERVACION DOCUMENTAL 2: \$ 400

VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS: \$ 20.400

JUSTIFICACION PAGO MAYOR VALOR:

PRUEBA 5 AM

Usuario: 51856

- Recibo pago Devoluciones



**RECIBO DE PAGO DEVOLUCIONES**  
**1236592202**

Impreso el 30 de Agosto de 2018 a las 10:25:03 am

No. RADICACIÓN: **2018-999-4-22**

DATOS SOLICITUD:

TURNO ASOCIADO: 2018-999-6-125

DESCRIPCIÓN: PRUEBAS

NOMBRE DEL SOLICITANTE: PRUEBAS PRUEBAS

Hago constar que recibo el dinero solicitado por concepto de devolución, de acuerdo con la resolución No. 25 del 27/08/2018, por medio de la forma de pago detallada a continuación,

CANAL DE RECAUDO XXXXXX

FORMA DE PAGO:

CONSIGNACION\_SUPERGIROS: VALOR: \$ 1.000.000  
CONSIG. BANCO: BANCO DE BOGOTÁ PIN: 8555855 FECHA CONSIG.: 29/08/2018 VALOR PAGADO:  
\$983.700 VALOR DOC.: \$1.500.000

Firma,

\_\_\_\_\_  
NOMBRE:  
C.C.

- Recibo Constancia de Solicitud de Devolución de Dinero

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
CIRCULO\_PRUEBAS

NIT:  
**CONSTANCIA DE SOLICITUD DE DEVOLUCIÓN DE DINERO**  
Impreso el 10 de Septiembre de 2018 a las 05:02:51 pm

No. RADICACIÓN: **2018-999-4-94**

**SOLICITANTE(S)**

NOMBRE: BELTRAN MORENO ANDRES

**DATOS SOLICITUD**

Turmo Asociado: 2018-999-6-529  
Descripción: PRUEBAAAAA

**DATOS PAGO INGRESADOS**

Canal de Recaudo xxxxxxxxxxxx  
Formas de pago

CONSIGNACION\_SUPERGIROS: VALOR: \$ 1.000.000  
CONSIG. BANCO: BANCO DE BOGOTA PIN: 8555855 FECHA CONSIG. 29/08/2018 VALOR PAGADO:  
\$983.700 VALORDOC.: \$1.500.000

TOTAL: \$1582700

**DOCUMENTOS ANEXADOS:**

RECIBO DE CAJA  
FOTOCOPIA DOCUMENTO IDENTIDAD SOLICITANTE

NUMERO DE FOLIOS: 1

- **Recibo de pago otros Recaudos “Fotocopias”**

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
CIRCULO\_PRUEBAS

NIT: 899999007-0

**62483341**

**Recibo de Pago de otros recaudos**

Impreso el 10 de Septiembre de 2018 a las 04:46:16 pm  
No. RADICACIÓN: **2018-999-5-21**

**Datos de Pago:**

Canal de Recaudo: XXXXXXXXXXXX

**FORMA DE PAGO:**

CONSIGNACION\_SUPERGIROS: VALOR: \$ 1.000.000  
CONSIG. BANCO: BANCO DE BOGOTA PIN: 8555855 FECHA CONSIG. 29/08/2018 VALOR PAGADO:  
\$983.700 VALORDOC.: \$1.500.000

VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: \$ 600

---

Solicitante: ANDRES BELTRAN MORENO  
Telefono: 21514548  
Radicado Por: 51856  
Usuario: 51856

- **Recibo de Certificados Nacionales**

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS

NIT: 899999007-0 CIRCULO\_PRUEBAS

SOLICITUD CERTIFICADO DE TRADICION

Impreso el 19 de Julio de 2019 a las 09:24:33 am

1236592790

No. RADICACIÓN: 2019-350-1-77149

TIPO DE CERTIFICADO: NACIONAL

MATRICULA: 350-8596

CERTIFICADO(S) SE EXPIDE(N) DE INMEDIATO

NOMBRE DEL SOLICITANTE: D S D

TELÉFONO: 8558

CERTIFICADOS: 1

FORMA DE PAGO:

CONSIGNACION\_SUPERGIROS: VALOR: \$ 1.000.000

CONSIG. BANCO: BANCO DE BOGOTA PIN: 8555855 FECHA CONSIG.: 29/08/2018 VALOR PAGADO:

\$983.700 VALOR DOC.: \$1.500.000

VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: \$ 16.800

EL CERTIFICADO SE EXPIDE DE ACUERDO A LOS DATOS SUMINISTRADOS

Usuario: 51856

- Recibo de Certificados Masivos

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS

NIT: 899999007-0 CIRCULO\_PRUEBAS

SOLICITUD CERTIFICADO DE TRADICION

Impreso el 19 de Julio de 2019 a las 09:21:25 am

1236592788

No. RADICACIÓN: 2019-999-1-221

MATRICULA: 999-8556

NO SE EXPIDE DE INMEDIATO EL CERTIFICADO

NOMBRE DEL SOLICITANTE: G DD G

CERTIFICADOS: 1

FORMA DE PAGO:

PAGO ASOCIADO AL TURNO: 2019-999-20-16

CONSIGNACION\_SUPERGIROS: VALOR: \$ 1.000.000

CONSIG. BANCO: BANCO DE BOGOTA PIN: 8555855 FECHA CONSIG.: 29/08/2018 VALOR PAGADO:  
\$983.700 VALOR DOC.: \$1.500.000

VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: \$ 16.800

EL CERTIFICADO SE EXPIDE DE ACUERDO A LOS DATOS SUMINISTRADOS

Usuario: 51856

- Recibo de Ampliación de Tradición

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS

NIT: 899999007-0 CIRCULO\_PRUEBAS

SOLICITUD CERTIFICADO DE TRADICION

Impreso el 19 de Julio de 2019 a las 09:19:08 am

**1236592786**

No. RADICACIÓN: **2019-999-1-219**

TIPO DE CERTIFICADO: AMPLIACION TRADICION

MATRICULA: **999-7974**

NO SE EXPIDE DE INMEDIATO EL CERTIFICADO

NOMBRE DEL SOLICITANTE: H H H

TELÉFONO: 747

CERTIFICADOS: 1

FORMA DE PAGO:

CONSIGNACION\_SUPERGIROS: VALOR: \$ 1.000.000

CONSIG. BANCO: BANCO DE BOGOTA PIN: 8555855 FECHA CONSIG.: 29/08/2018 VALOR PAGADO:  
\$983.700 VALOR DOC.: \$1.500.000

VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: **\$ 36.800**

EL CERTIFICADO SE EXPIDE DE ACUERDO A LOS DATOS SUMINISTRADOS

Usuario: 51856

- Recibo de Pertenencia

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS

NIT: 899999007-0 CIRCULO\_PRUEBAS

SOLICITUD CERTIFICADO DE TRADICION

Impreso el 19 de Julio de 2019 a las 09:17:24 am

**1236592785**

No. RADICACIÓN: **2019-999-1-218**

TIPO DE CERTIFICADO: PERTENENCIA

MATRICULA: **999-8797**

NO SE EXPIDE DE INMEDIATO EL CERTIFICADO

NOMBRE DEL SOLICITANTE: H H H

TELÉFONO: 78941

CERTIFICADOS: 1

FORMA DE PAGO:

CONSIGNACION\_SUPERGIROS: VALOR: \$ 1.000.000

CONSIG. BANCO: BANCO DE BOGOTA PIN: 8555855 FECHA CONSIG.: 29/08/2018 VALOR PAGADO:  
\$983.700 VALOR DOC.: \$1.500.000

VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: **\$ 36.400**

EL CERTIFICADO SE EXPIDE DE ACUERDO A LOS DATOS SUMINISTRADOS

Usuario: 51856

- Recibo de Certificado de Antiguo Sistema

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
NIT: 899999007-0 CIRCULO\_PRUEBAS  
SOLICITUD CERTIFICADO DE TRADICION  
Impreso el 19 de Julio de 2019 a las 09:15:08 am

1236592784

No. RADICACIÓN: 2019-999-1-217

TIPO DE CERTIFICADO: ANTIGUO SISTEMA

DATOS ANTIGUO SISTEMA:

LIBRO: Año: 2019 No: GJJ Pg: 1 Tipo: 1      TOMO: Año: 2019 No: 1 Pg: 1 Mun: 1  
COMENTARIO: PRUEBAS

NO SE EXPIDE DE INMEDIATO EL CERTIFICADO

NOMBRE DEL SOLICITANTE: 1 HFHF GHFHG

TELÉFONO: 71711

CERTIFICADOS: 1

FORMA DE PAGO:

CONSIGNACION\_SUPERGIROS: VALOR: \$ 1.000.000  
CONSIG. BANCO: BANCO DE BOGOTA PIN: 8555855 FECHA CONSIG.: 29/08/2018 VALOR PAGADO:  
\$983.700 VALOR DOC.: \$1.500.000

VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: \$ 16.800

EL CERTIFICADO SE EXPIDE DE ACUERDO A LOS DATOS SUMINISTRADOS

Usuario: 51856

### - Reimpresión de Recibos

Se solicita que, al momento de realizar la reimpresión de recibos desde la aplicación, los mismos se originen con la misma información planteada para la impresión de recibos descrita anteriormente, estos se obtienen a través de la siguiente ruta.

- Seleccionar el botón Administración, desde la pantalla principal del Sistema de Información Registral



- Seleccionar la opción Reimpresión de Recibos

## IMPRESION

- Administrador de Impresoras
- Reimpresión de Recibos
- Impresión Simple de Folio
- Reimpresión Formulario de Calificación
- Reimpresion de Consultas
- Reimpresin de Caratulas Reparto Notarial
- Reimpresin de Actas Reparto Notarial
- Reimpresion Formulario de Correcciones
- Reimpresión constancia de testamento

- Se despliega el listado de recibos a reimprimir.

Datos del turno para la reimpresión del recibo

Datos del Turno	
Proceso	<b>-SELECCIONE UNA OPCION-</b>
Re-Imprimir	CERTIFICADOS CERTIFICADOS MASIVOS CONSULTAS CORRECCIONES DEVOLUCION FOTOCOPIAS REGISTRO REPARTO NOTARIAL MINUTAS
Volver	<input type="checkbox"/> Reimprime recibo mayor valor



**DESPLIEGUE 7.4.2**

11/07/2019



### **OC 2302 Solicitud Suspensión Trámite de Registro**

Se crea en el sistema SIR dos fases nuevas, las cuales permiten enviar los turnos de registro desde el rol calificador al rol coordinador jurídico exigiendo documentar el motivo del envío, dichas fases están contempladas en los artículos 18 y 19 de la ley 1579 del 2012.

- “ARTICULO 18 SUSPENSIÓN DEL TRÁMITE DE REGISTRO A PREVENCIÓN”
- “ARTICULO 19 SUSPENSIÓN TEMPORAL DEL TRÁMITE DE REGISTRO”

Se crea una alerta en el sistema la cual notifique al coordinador jurídico la cantidad de turnos que tiene a punto de vencerse en cada una de las fases, pasado 30 días hábiles a partir de la llegada de los turnos en trámite de suspensión.

En el rol coordinador jurídico se desarrollar una opción que le permita a este devolver el turno al rol calificador con la respuesta indicada en cada una de las fases, igualmente para este mismo rol se implementa una opción que permita cargar la resolución y una sección donde se pueda documentar una nota informativa.

De otro lado se modifican los reportes a que haya lugar con el fin de establecer los turnos que están o han estado en las fases nuevas.

- **Rol calificador**

Se agregan las dos nuevas fases en el rol calificador en la sección “información folio / solicitar información de folios a otras dependencias”

La selección en la lista desplegable tiene las siguientes fases:

- Suspensión del trámite de registro a prevención
- Suspensión temporal del trámite de registro.

Ruta donde se realiza el ajuste, se visualiza en imagen 1 y 2

Imagen 1

Revisión encabezado

Datos Básicos

Tipo: 13 - EXTRACTO Número: 55685

Oficina de Procedencia

Código: 1 Nombre: EMPRESA SERVICIOS PI ALCANTARILLADO DE B

Departamento: BOGOTA D.C. Municipio: BOGOTA D.C.

Imagen 2

Consultar información de los folios

Solicitar información de folios a otras dependencias

Enviar el caso a :

SELECCIONE UNA OPCIÓN  
ANTIGUO SISTEMA  
CORRECCIONES

Se agregan las dos nuevas fases en la lista desplegable, para que el rol calificador pueda enviar el turno de registro al coordinador jurídico, se visualiza en imagen 3.

Imagen 3

Consultar información de los folios

Solicitar información de folios a otras dependencias

Enviar el caso a :

SUSPENSION DEL TRAMITE DE REGISTRO A PREVENCION  
SUSPENSION TEMPORAL DEL TRAMITE DE REGISTRO

- Nota informativa rol calificador

Se adiciona a la lista de notas informativas del rol calificador, una nueva nota de manera obligatoria para el envío del turno por suspensión del trámite de registro. Se visualiza en imagen 4 y 5

Imagen 4

Imagen 5

#### Condiciones

- El turno no puede ser enviado al coordinador jurídico sin haber documentado una nota informativa
- El sistema debe validar que el tipo de nota informativa escogida sea: “*envió a suspensión del trámite de registro*”, de escoger otra distinta el sistema debe restringir el envío del turno.
- Mensaje de selección aceptar/cancelar

Este mensaje se visualiza cuando el calificador da clic en el botón “enviar caso”, una vez haya seleccionado un trámite de suspensión de la lista desplegable, y haya agregado una nota informativa de envió a suspensión del trámite de registro, estructura del mensaje “*¿está seguro que desea reenviar el turno para que sea revisado en otras dependencias?*” se visualiza en imagen 6

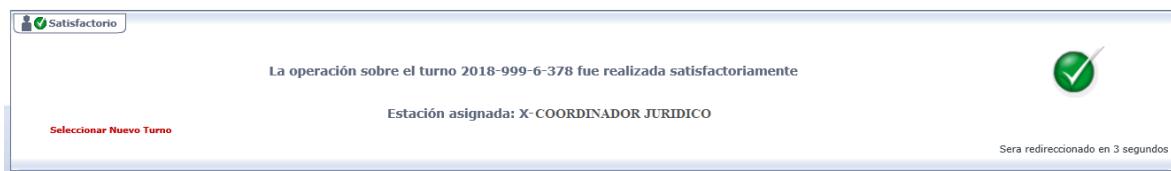
Imagen 6

The screenshot shows a software interface with a sidebar menu on the left containing options like 'Revisión y Análisis', 'Información Turno', 'Información Folio', 'Información Registro', 'Inscripción', 'Inscripción Parcial', 'Devolución', 'Digitador Masivo', 'Opciones', 'Formulario Temporal', and 'Notas Informativas'. The main area has a title 'Consultar información de los folios' with a grid of folio numbers: 999-1820, 999-1821, 999-9853, 999-9854, 999-9860, and 999-9861. Below this is a search bar for 'Consulta de folio por número de matrícula' with a 'Consultar' button. A modal dialog titled 'Mensaje de página web' asks '¿Está seguro que desea reenviar el turno para que sea revisado en otras dependencias?' with 'Aceptar' and 'Cancelar' buttons. At the bottom, there's a section 'Solicitar información de folios a otras dependencias' with a dropdown menu set to 'SUSPENSIÓN TEMPORAL DEL TRAMITE DE REGISTRO' and a 'Enviar Caso' button.

- Mensaje de confirmación

Este mensaje presenta cuando la operación de envió a suspensión se realizó satisfactoriamente, se visualiza en la imagen 7

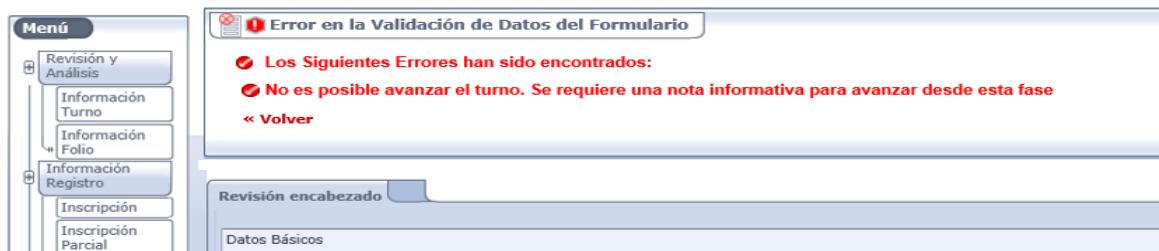
Imagen 7



#### Mensaje de restricción

Este mensaje se presenta si el funcionario no agrego una nota informativa o escogió una nota diferente a la del trámite de suspensión, el sistema debe restringir el cambio de fase del turno. Se visualiza en imagen 7

Imagen 7



- Rol coordinador jurídico

Se crea en el menú del rol coordinador jurídico dos nuevas opciones que permitan al usuario visualizar los turnos que tiene en suspensión por cada una de las fases antes mencionadas en el documento.

Las dos nuevas opciones son:

- Tramite de suspensión a prevención,
- Tramite suspensión temporal

- Las dos nuevas fases permiten listar los turnos que cada una tiene, de la misma manera que actualmente lo hace la opción “Aprobar Corrección calificación”. Que permite seleccionar un turno para trabajar sobre él.  
**Nota: para lo anterior tomar como referencia la navegación del rol Registrador.**
- Se crea para la pantalla de rol coordinador jurídico una alerta que la visualice al funcionario la cantidad de turnos que se encuentran en las fases mencionadas y que están próximos a vencer.

Los atributos de la pantalla Se visualizan en imagen 8.

Imagen 8



- **Pantalla trámite de suspensión a prevención**

Se desarrollar en el rol coordinador jurídico una pantalla para la opción “trámite de suspensión a prevención” la cual se abre al seleccionar un turno de la lista que se encuentre en esta fase.

La pantalla contiene los siguientes campos:

- Campo de texto nombrado como “descripción de la respuesta”, el cual debe permitir un mínimo de 20 caracteres.
- Campo adjunto de documento nombrado como “documento justificación”, el cual permitirá adjuntar al turno la resolución elaborada por el coordinador jurídico para la suspensión de este.
- Botón Agregar, el cual debe permitir adjuntar el documento de justificación y la descripción de la respuesta, una vez sea accionado el botón agregar este debe listar en la parte inferior de la pantalla sección “respuesta agregada” la respuesta que el coordinador jurídico asocio al turno.
- Sección respuesta agregada, esta debe permitir visualizar al funcionario los siguientes datos, “nombre de la fase, fecha, usuario, descripción”.  
Nota estos campos son solo de consulta y la fecha de respuesta debe ser automática tomada de la “fecha del sistema”
- Botón Regresar: este debe regresar a la pantalla anterior “rol coordinador jurídico”
- Botón Cancelar: este debe permitir cancelar los datos de respuesta y debe refrescar la página.
- Botón nota informativa: este debe permitir agregar una nota informativa de carácter obligatoria para documentar y llevar una traza del turno en el trámite de suspensión.

- Botón enviar respuesta: este debe permitir validar los datos capturados y enviar el turno con la respuesta al rol calificador.
- Botón de descarga o visualización del documento adjunto, este debe permitir al funcionario revisar el documento antes de enviar.

Los atributos de la pantalla se visualizan en las imágenes 9 y 10

Imagen 9

Imagen 10

➤ Notas informativas

Crear en el sistema dos notas informativas para capturar las observaciones del trámite de suspensión de registro, las notas deben ser obligatorias al momento de escoger en la lista desplegable “suspensión del trámite de registro” o “envió por error al trámite de suspensión” de lo contrario el aplicativo debe restringir cualquier otro tipo de nota. Se visualiza en imagen 11 y 12

Imagen 11

Agregar Nota Informativa

Notas Informativas

Nota

Escoger Tipo de Nota: XXX - SUSPENSIÓN DEL TRÁMITE DE REGISTRO

NOTA INFORMATIVA SUSPENSIÓN DEL TRÁMITE DE REGISTRO

Descripción: XXXXXXXXXXXXXXXX

Agregar, Regresar, Cancelar

REVISIÓN Y ANALISIS CORRECCION INTERNA

USUARIO.PRUEBAS

Imagen12

Agregar Nota Informativa

Notas Informativas

Nota

Escoger Tipo de Nota: XXX - ENVIÓ POR ERROR AL TRÁMITE DE SUSPENSIÓN

NOTA INFORMATIVA ENVÍO POR ERROR A SUSPENSIÓN DEL TRÁMITE DE REGISTRO

Descripción: XXXXXXXXXXXXXXXX

Agregar, Regresar, Cancelar

REVISIÓN Y ANALISIS CORRECCION INTERNA

USUARIO.PRUEBAS

- Condiciones para las notas informativas del rol coordinador jurídico:

La nota informativa “suspensión del trámite de registro” se envia en conjunto con el documento de justificación al momento de enviar el turno con la respuesta al rol calificador, de lo contrario el turno no puede ser enviado a dicho rol.

- Botón enviar respuesta

Este botón valida que el turno ya tenga una nota informativa, documento de justificación y una descripción de la respuesta, de no ser así el sistema debe restringir el envío del turno.

- Mensaje de selección aceptar/cancelar

Este mensaje se visualiza cuando se ha seleccionado un trámite de suspensión y se haya digitado un motivo de envío, estructura del mensaje “¿está seguro de que desea enviar la respuesta?” se visualiza en imagen 13

Imagen 13

➤ Mensaje de restricción

El aplicativo no deja avanzar el turno de fase sin que se agregue la descripción de la respuesta, sin que se haya creado una nota informativa o sin que se haya adjuntado un documento de justificación. Se visualiza en imagen 14.

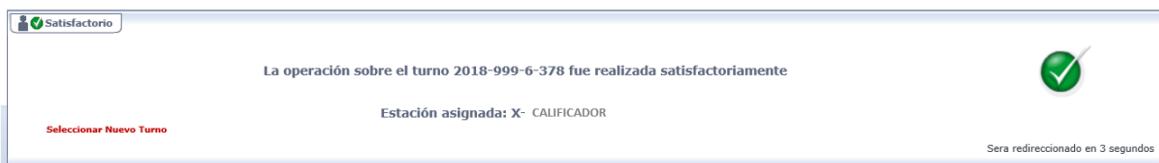
Imagen 14



➤ Mensaje de confirmación

Este mensaje se presenta cuando la operación de envió al rol calificador con la respuesta del trámite de suspensión se realizó satisfactoriamente, como se visualiza en la imagen 15.

Imagen 15



• Pantalla trámite de suspensión Temporal

Se desarrolla en el rol coordinador jurídico una pantalla para la opción “trámite de suspensión temporal” la cual deberá abrir al seleccionar un turno de la lista que se encuentre en esta fase.

La pantalla tiene lo siguiente:

- Campo de texto nombrado como “descripción de la respuesta”, el cual debe permitir un mínimo de 20 caracteres como otras funcionalidades ya existentes en el sistema.
- Campo adjunto de documento nombrado como “documento justificación”, el cual permitirá adjuntar al turno la resolución elaborada por el coordinador jurídico para la suspensión de este.

- Botón Agregar, el cual debe permitir adjuntar el documento de justificación y la descripción de la respuesta, una vez sea accionado el botón agregar este debe listar en la parte inferior de la pantalla sección “respuesta agregada” la respuesta que el coordinador jurídico asocio al turno.
- Sección respuesta agregada, esta debe permitir visualizar al funcionario los siguientes datos, “nombre de la fase, fecha, usuario, descripción”.  
Nota estos campos son solo de consulta y la fecha de respuesta debe ser automática “fecha del sistema”
- Botón Regresar: este debe regresar a la pantalla anterior “rol coordinador jurídico”
- Botón Cancelar: este debe permitir cancelar los datos de respuesta y debe refrescar la página.
- Botón nota informativa: este debe permitir agregar una nota de carácter obligatoria para documentar y llevar una traza del turno en el trámite de suspensión.
- Botón enviar respuesta: este debe permitir validar los datos capturados y enviar el turno con la respuesta al rol calificador.
- Botón de descarga o visualización del documento adjunto, este debe permitir al funcionario revisar el documento antes de enviar.

Los atributos de la pantalla se visualizan en las imágenes 16 y 17

Imagen 16

Imagen 17

➤ Notas informativas

Se crea en el sistema dos notas informativas para capturar las observaciones del trámite de suspensión de registro, las notas deben ser obligatorias al momento de escoger en la lista desplegable “suspensión del trámite de registro” o “envió por error al trámite de suspensión” de lo contrario el aplicativo debe restringir cualquier otro tipo de nota. Se visualiza en imagen 18 y 19.

Imagen 18

The screenshot shows a software interface titled 'Agregar Nota Informativa'. At the top, there's a button labeled 'Notas Informativas' with a red exclamation mark icon. Below it is a section titled 'Nota' with a sub-section 'Escoger Tipo de Nota' containing a dropdown menu with the value 'XXX - SUSPENSIÓN DEL TRÁMITE DE REGISTRO'. To the right of this dropdown is another dropdown menu with the value 'NOTA INFORMATIVA SUSPENSIÓN DEL TRÁMITE DE REGISTRO'. Below these dropdowns is a large text area labeled 'Descripción' containing the text 'XXXXXXXXXXXX'. At the bottom of the window are buttons for 'Agregar', 'Regresar', and 'Cancelar', along with a status bar that reads 'REVISIÓN Y ANALISIS CORRECCION INTERNA' and 'USUARIO.PRUEBAS'.

Imagen19

This screenshot is identical to Image 18, except the 'Escoger Tipo de Nota' dropdown is now set to 'XXX - ENVIÓ POR ERROR AL TRÁMITE DE SUSPENSIÓN'. The rest of the interface, including the note content and the status bar, remains the same.

- Condiciones para las notas informativas del rol coordinador jurídico:

La nota informativa “suspensión del trámite de registro” se envia en conjunto con el documento de justificación al momento de enviar el turno con la respuesta al rol calificado, de lo contrario el turno no puede ser enviado a dicho rol.

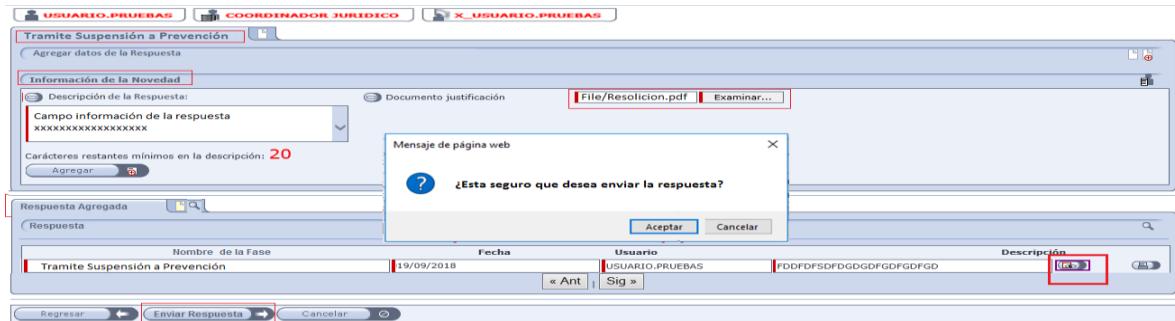
- Botón enviar respuesta

Este botón valida que el turno ya tenga una nota informativa, documento de justificación y una descripción de la respuesta, de no ser así el sistema debe restringir el envío del turno.

- Mensaje de selección aceptar/cancelar

Este mensaje se visualiza cuando se ha seleccionado un trámite de suspensión y se haya digitado un motivo de envío, estructura del mensaje “¿está seguro de que desea enviar la respuesta?” se visualiza en imagen 20.

Imagen 20



➤ Mensaje de restricción

El aplicativo no deja avanzar el turno de fase sin que se agregue la descripción de la respuesta, o sin que se haya adjuntado un documento de justificación. Se visualiza en imagen 21.

Imagen 21



➤ Mensaje de confirmación

Este mensaje se presenta cuando la operación de envió al rol calificador con la respuesta del trámite de suspensión se realizó satisfactoriamente, como se visualiza en la imagen 22.

Imagen 22



➤ Respuesta en el rol calificador

Se crea una nueva sesión en el rol calificador desde donde se descargue la respuesta del coordinador jurídico. Se visualiza en imagen 23.

Imagen 23

**Nota** en todas las secciones donde se documenten notas informativas adicionar las opciones de imprimir y consulta. Se visualiza en las imágenes 24 y 25

Imagen 24.

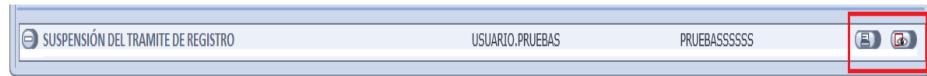


Imagen 25.

## EXCEPCIONES

En caso de que el turno de registro enviado a alguna de las fases se realiza por error, el sistema debe contemplar la inclusión de una nota informativa “Envío por error al trámite de suspensión” el turno se podrá enviar desde el coordinador jurídico sin necesidad de adjuntar el documento de justificación.

➤ Reporte G03 Informe de turnos en registro por fases

Se agrega al reporte g03 las dos nuevas fases de trámite de suspensión, “suspensión del trámite de registro a prevención y suspensión temporal del trámite de registro”, Se visualiza en imagen 26 y 27.

Imagen 26



**Superintendencia de Notariado y Registro**  
Ministerio del Interior y de Justicia  
República de Colombia  
[www.supernotariado.gov.co](http://www.supernotariado.gov.co)

**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS** CIRCULO\_PRUEBAS - 999

**INFORME DE TURNO DE REGISTRO POR FASES**

**REALIZADO POR:** USUARIO.PRUEBAS

Impreso el 01 de Noviembre de 2018 a las 12:05:38 PM

Página 1

**USUARIO.PRUEBAS**

CONFRONTACION	REPARTO	CALIFICACION	TESTAMENTOS	MESA CONTROL	FIRMA REGISTRADOR	CERTIFICADOS ASO	SUSPENSIÓN A PREVENCIÓN	SUSPENSIÓN TEMPORAL
2006-999-6-557	2017-999-6-121	2006-999-6-566	2016-999-6-107	2006-999-6-580	2012-999-6-376	2016-999-6-2	2017-999-6-121	2016-999-6-85
2006-999-6-621	2017-999-6-125	2016-999-6-10	2018-999-6-69	2006-999-6-665	2012-999-6-377	2016-999-6-5	2017-999-6-125	2016-999-6-86
2006-999-6-648	2017-999-6-126	2016-999-6-61	2018-999-6-133	2016-999-6-48	2016-999-6-23	2016-999-6-14	2017-999-6-126	2016-999-6-88
2016-999-6-105	2017-999-6-144	2016-999-6-79		2016-999-6-58	2016-999-6-51	2016-999-6-18	2017-999-6-144	2016-999-6-90
2016-999-6-376	2017-999-6-150	2016-999-6-85		2016-999-6-63	2016-999-6-67	2016-999-6-19	2017-999-6-150	
2017-999-6-44	2017-999-6-151	2016-999-6-86		2016-999-6-66	2016-999-6-73	2016-999-6-31	2017-999-6-151	
2017-999-6-164	2017-999-6-152	2016-999-6-88		2016-999-6-109	2016-999-6-78	2016-999-6-37	2017-999-6-152	
2018-999-6-13	2017-999-6-155	2016-999-6-90		2016-999-6-113	2016-999-6-278	2016-999-6-54	2017-999-6-155	
2018-999-6-24	2018-999-6-1	2016-999-6-100		2016-999-6-159	2016-999-6-360	2016-999-6-56	2018-999-6-1	
2018-999-6-25	2018-999-6-2	2016-999-6-102		2016-999-6-160	2017-999-6-142	2016-999-6-68	2018-999-6-2	

Imagen 27

TOTAL TURNO	24	26	132	3	48	10	59	10	4
<b>RESUMEN GENERAL</b>									
CONFRONTACION			24						
REPARTO			26						
CALIFICACION			132						
TESTAMENTOS			3						
MESA CONTROL			48						
FIRMA REGISTRADOR			10						
CERTIFICADOS ASOCIADOS			59						
SUSPENSIÓN DEL TRÁMITE DE REGISTRO A PREVENCIÓN			10						
SUSPENSION TEMPORAL DEL TRÁMITE DE REGISTRO			4						

➤ Reporte G02 Informe Estadístico de turnos por Calificador.

Se agrega al reporte g02 las dos nuevas fases de trámite de suspensión, “suspensión del trámite de registro a prevención y suspensión temporal del trámite de registro”, Se visualiza en imagen 28 y 29.

Imagen 28

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS BUCARAMANGA - 300  
 INFORME ESTADÍSTICO DE TURNOS POR CALIFICADOR DE 2018-10-22 A 2018-10-22

REALIZADO POR:USUARIO.PRUEBAS

Impreso el 22 de Noviembre de 2018 a las 16:23:55 PM

Página 1

CALIFICADOR	TURNOS REPARTIDOS	TURNOS REASIGNADOS, RESTITUIDOS Y/O REANOTADOS	TURNOS CALIFICADOS	TURNOS INSCRITOS	TURNOS DEVUeltos AL PUBLICO SIN REGISTRAR	TURNOS INSCRITOS PARCIALMENTE	TURNOS PENDIENTES (CALIFICACION)	TURNOS REDIRECCIONADOS A/S CORRECCIONES	TURNOS EN TRAMITE	SUSPENSIÓN TEMPORAL	SUSPENSIÓN A PREVENCIÓN
ANGIEE TATIANA OJEDA SALAZAR	0	16	15	13	2	0	1	0	0	0	0
ARIEL FERNAN RUEDA GOMEZ	25	1	23	21	1	1	0	0	1	1	1
ARLEY TORRES GARCIA	22	0	21	21	0	0	0	0	0	0	0
CLAUDIA RUEDA VILLALBA	24	2	22	19	3	0	0	0	0	0	0
GLORIA CRISTINA ORDÓNEZ RIVERO	25	0	23	22	1	0	2	0	0	0	0
JAIRO CAMARGO REY	25	0	21	19	2	0	0	0	0	0	0
JORGE ENRIQUE DIAZ CUIJA	0	1	1	1	0	0	0	0	0	0	0
LEYZA VALBUENA TORO	1	0	1	1	0	0	0	0	0	0	0
Luz Lizarazo Zarate	25	1	30	25	4	1	0	0	0	0	0
Luz Dari Angulo Vergara	0	1	5	5	0	0	0	0	0	0	0
Luz Marina Arguello	25	1	24	23	1	0	2	0	0	0	0
Maria Penalosa Rodriguez	26	0	28	25	1	2	0	0	0	0	0
NINOSKA BERNAL TOSCANO	23	0	13	13	0	0	0	0	0	0	0

Imagen 29

RESUMEN GENERAL	
TOTAL TURNOS REPARTIDOS	221
TOTAL TURNOS CALIFICADOS	227
TOTAL TURNOS REASIGNADOS, RESTITUIDOS Y/O REANOTADOS	23
TOTAL TURNOS INSCRITOS	208
TOTAL TURNOS DEVUeltos AL PUBLICO	15
TOTAL TURNOS INSCRITOS PARCIALMENTE	4
TOTAL TURNOS EN TRAMITE	1
TOTAL TURNOS REDIRECCIONADOS A/S Y CORRECCIONES	0
TOTAL TURNOS PENDIENTES POR CALIFICAR	5
TOTAL TURNOS SUSPENSIÓN TEMPORAL	1
TOTAL TURNOS SUSPENSIÓN A PREVENCIÓN	1

## NOTA:

TURNOS CALIFICADOS: Es la sumatoria de los turnos Devueltos al Publico, Inscritos e Inscritos Parcialmente.

TURNOS EN TRAMITE: Es la sumatoria de los turnos que se encuentran en las fases: Confrontación Correctiva, Digitación Calificaciones, Aprobar Corrección Calificación, Custodia Pagos Mayores, Notificar Ciudadano Pago Mayor, Registrar Pagos Mayores, Notificar Funcionario Pagos Mayores, Corrección Encabezado, Mesa Control y Firmar Registro.

TURNOS REASIGNADOS, RESTITUIDOS Y/O REANOTADOS: Es la sumatoria de los turnos que fueron reasignados, restituidos y/o reanotados.

Total Turnos Suspensión temporal y Total Suspensión a prevención son las fases donde se han enviado los turnos por motivo de suspensión del trámite de registro.

## OC 2338 Deshabilitar Reparto Notarial

El sistema permite realizar mediante el Rol RECEPTOR REPARTO NOTARIAL, La radicación de minutas aplicando diferentes categorías según el tipo de acto agregado para dicha radicación, se modifco para este proceso y se deshabilitan las diferentes categorías y solo queda habilitada la categoría SEPTIMA, con sus correspondientes actos:

- 48\_DECLARACION DE LA POSESION REGULAR.
- 49\_DECLARATORIA DE PRESCRIPCION ADQUISITIVA.

- Se desabilito por base de datos las diferentes categorías y solamente se dejó activa la categoría SEPTIMA.
- Por el Rol RECEPTOR REPARTO NOTARIAL, en la fase Reparto\_notarial\_minutas, en el apartado Acciones notariales; se desabilito de la lista desplegable los actos que no corresponden a la categoría SEPTIMA. Figura 1.

FIGURA 1.

	<div style="text-align: center;">    <b>DESPLIEGUE 7.4.1</b>  <b>06/06/2019</b> </div>	 <b>SUPERINTENDENCIA DE NOTARIADO &amp; REGISTRO</b> <i>La guarda de la fe pública</i>																																				
<b>OC 2325 Ajuste tarifas registrales 2019</b>																																						
<p>El Sistema de Información Registral SIR debe ajustarse teniendo en cuenta las nuevas tarifas registrales estipuladas en la resolución No. 6610 del 27 de mayo de 2019 para todos los procesos que impliquen un cobro en el sistema.</p> <p><b>ACTOS CON CUANTIA:</b></p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th colspan="4"><i>Actos con Cuantía 2019</i></th> </tr> <tr> <th><i>Rangos</i></th> <th><i>Inicio</i></th> <th><i>Final</i></th> <th><i>Tarifas</i></th> </tr> </thead> <tbody> <tr> <td>&lt;10 SMLMV</td> <td>-</td> <td>&lt; \$ 8.281.160</td> <td>\$ .36.400</td> </tr> <tr> <td>&gt;10 SMLMV; &lt;=150 SMLMV</td> <td>&gt;= \$ 8.281.161</td> <td>&lt;= \$ 107.655.080</td> <td>6,22 X MIL</td> </tr> <tr> <td>&gt;150 SMLMV; &lt;=260 SMLMV</td> <td>&gt;= \$ 107.655.081</td> <td>&lt;= \$ 215.310.160</td> <td>7,73 X MIL</td> </tr> <tr> <td>&gt;260 SMLMV; &lt;= 385 SMLMV</td> <td>&gt;= \$ 215.310.161</td> <td>&lt;= \$ 318.824.660</td> <td>8,61 X MIL</td> </tr> <tr> <td>&gt;385 SMLMV</td> <td>&gt;= \$ 318.824.661</td> <td></td> <td>9,10 X MIL</td> </tr> <tr> <td style="background-color: #cccccc;"><i>&lt; MENOR QUE</i></td> <td colspan="3" style="background-color: #cccccc;"><i>&gt; MAYOR QUE</i></td> </tr> <tr> <td style="background-color: #cccccc;"><i>&lt;= MENOR IGUAL QUE</i></td> <td colspan="3" style="background-color: #cccccc;"><i>&gt;= MAYOR IGUAL QUE</i></td> </tr> </tbody> </table> <p><b>ACTOS SIN CUANTÍA</b></p> <p>a) la suma de veinte mil trescientos pesos (\$ 20.300) por cada uno de los actos que por su naturaleza carezcan de cuantía en el documento de inscripción. Salvo los casos en esta resolución, también deberá cancelarse la suma de diez mil setecientos pesos (\$ 10, 700), por cada folio de matrícula adicional donde deba inscribirse el documento.</p> <p><b>ACTOS MINIMA CUANTIA:</b></p>			<i>Actos con Cuantía 2019</i>				<i>Rangos</i>	<i>Inicio</i>	<i>Final</i>	<i>Tarifas</i>	<10 SMLMV	-	< \$ 8.281.160	\$ .36.400	>10 SMLMV; <=150 SMLMV	>= \$ 8.281.161	<= \$ 107.655.080	6,22 X MIL	>150 SMLMV; <=260 SMLMV	>= \$ 107.655.081	<= \$ 215.310.160	7,73 X MIL	>260 SMLMV; <= 385 SMLMV	>= \$ 215.310.161	<= \$ 318.824.660	8,61 X MIL	>385 SMLMV	>= \$ 318.824.661		9,10 X MIL	<i>&lt; MENOR QUE</i>	<i>&gt; MAYOR QUE</i>			<i>&lt;= MENOR IGUAL QUE</i>	<i>&gt;= MAYOR IGUAL QUE</i>		
<i>Actos con Cuantía 2019</i>																																						
<i>Rangos</i>	<i>Inicio</i>	<i>Final</i>	<i>Tarifas</i>																																			
<10 SMLMV	-	< \$ 8.281.160	\$ .36.400																																			
>10 SMLMV; <=150 SMLMV	>= \$ 8.281.161	<= \$ 107.655.080	6,22 X MIL																																			
>150 SMLMV; <=260 SMLMV	>= \$ 107.655.081	<= \$ 215.310.160	7,73 X MIL																																			
>260 SMLMV; <= 385 SMLMV	>= \$ 215.310.161	<= \$ 318.824.660	8,61 X MIL																																			
>385 SMLMV	>= \$ 318.824.661		9,10 X MIL																																			
<i>&lt; MENOR QUE</i>	<i>&gt; MAYOR QUE</i>																																					
<i>&lt;= MENOR IGUAL QUE</i>	<i>&gt;= MAYOR IGUAL QUE</i>																																					

c) la suma de diez mil setecientos pesos (\$ 10.700) por cada folio de matricula que deba abrirse;

d) la suma de veinte mil trescientos pesos (\$ 20.300) por la inscripcion o revocatoria de testamentos.

**ARTICULO 11. Copias.** La copia de un documento inscrito, de resoluciones, de actuaciones administrativas, de inscripciones del antiguo sistema de registro, de instrumentos publicos que reposen en los archivos de la Entidad o de cualquier otro que se conserve en los archivos de las oficinas de registro de instrumentos publicos causara derechos asi:

- a) De docuemnts almacenados en medio magnetico la suma de mil doscientos pesos (\$1.200) por cada pagina reproducida;
- b) De documentos que reposen en los archivos fisicos de la respectiva oficina de registro, la suma de setecientos pesos(\$700) por cada pagina fotocopiada;
- c) La expedicion de copias virtuales a nivel nacional para la obtencion de registros y expedientes inmobiliarios que se encuentren organizados, digitalizados e indexados en la sede virtual nacional de copias de acuerdo con los lineamientos del Archivo General de la Nacion, que sean solicitadas y expedidas por canales virtuales o presenciales, causaran derechos por la suma de dos mil cien pesos (\$2.100).

**ARTICULO 12. Certificados.** Los certificados que según la ley corresponde expedir a los registradoes de instrumentos publicos, según el caso, causaran los derechos asi:

- a) Los certificados de tradicion que se expiden en las oficians de registro causaran derechos equivalentes a la suma de dieciseis mil ochocientos pesos (316.800).
- b) Los certificados de tradicion que correspondan a predios de mayor extension causaran derechos por la suma de treinta y seis mil cutrocientos pesos (\$36.400) cada uno. Entiendase por certificados de tradicion que correspondan a predios de mayor extebnsion, los que superen ciento cincuenta (150) anotaciones registrales;

En el caso de los certificados asociados a un turno de registro, si el folio tiene mas de 150 ciento cincuenta anotaciones (mayor extension), causaran igualmente derechos por la suma de treinta y seis mil cuatrocientos pesos (\$36.400) cada uno;

- c) Las certificaciones que según la ley corresponde expedir para adelantar procesos de pertenencia o de adjudicacion de bienes baldios urbanos o rurales (carencias registrales), causaran derechos por la suma de treinta y seis mil cuatrocientos pesos (\$36.400) cada uno;
- d) Los certificados contentivos de ampliacion a la tradicion de un inmueble por un lapso superior a los veinte (20) años, causaran derechos por la suma de treinta y seis mil ochocientos pesos (\$36.800) cada uno;



**DESPLIEGUE 7.4**

**09/05/2019**



**SUPERINTENDENCIA  
DE NOTARIADO  
& REGISTRO**  
La guardia de la fe pública

### **OC 2289 Canales de Recaudo y Formas de Pago**

Se solicita parametrizar en el Sistema Misional SIR los canales de recaudo y formas de pago que tiene disponible cada Oficina de Registro, y discriminar los datos que tiene que capturar el cajero por cada uno, en el momento de radicar turnos para la prestación de servicio.

Para el desarrollo se deben tener en cuenta los siguientes pasos:

- Se solicita que una vez el funcionario realice el proceso de solicitud de registro y se disponga a registrar el pago **“pantalla de registro de pago”** el sistema debe validar y visualizar los canales de recaudo que tiene asociada la ORIP en una lista desplegable.
- Una vez el funcionario seleccione un canal de recaudo el sistema debe validar y visualizar las formas de pago asociados al canal de recaudo en una lista desplegable.
- Al seleccionar una forma de pago el sistema debe validar y visualizar los campos de captura.

Lo anterior implica realizar desarrollos y modificaciones en las pantallas donde se capturan las formas de pago, los imprimibles del sistema, pantallas de consulta, pantallas administrativas, y los reportes donde se discriminen las formas de pago.

El sistema permitirá la parametrización tanto de canales, como formas de pago y administrar las entidades y cuentas bancarias de destino de los dineros que ingresan por cada uno.

### **PANTALLA ADMINISTRACIÓN DE FORMAS DE PAGO**

Se creó una pantalla de administración y creación de formas de pago, para el rol administrador nacional, la cual se debe abrir desde una nueva opción en el menú Administración / Pantallas Administrativas / Tarifas Pagos y Recibos, dicha pantalla debe tener las opciones agregar una forma de pago, activar e inactivar los datos de captura que debe tener cada forma de pago.

Desde esta pantalla se podrá crear las formas de pago en el sistema, además debe permitir activar o inactivar los datos de captura de dichas formas.

- En la nueva pantalla se podrá visualizar en una lista desplegable las formas de pago que se encuentran el sistema con sus respectivos campos de captura.

### Pantalla Creación forma de pago

### Pantalla selección forma de pago

Nombre	Estado	Activar	Inactivar
Banco / Franquicia	INACTIVO		
No Documento de Pago	ACTIVO		
No Aprobacion	INACTIVO		
No PIN	INACTIVO		
No NIR	INACTIVO		
No CUS	INACTIVO		
Valor Documento	ACTIVO		
Valor Pagado	ACTIVO		
Fecha Documento	ACTIVO		

## Pantalla selección forma de pago

The screenshot shows a software interface titled "Pantalla Administrativa" with a sub-tab "Consulta de Formas de Pago". The main area is titled "Resultados de la Consulta" and contains a table titled "Listado". The table has columns for "Nombre" (Name), "Estado" (State), and "Activar Inactivar" (Activate/Deactivate). The data rows are:

Nombre	Estado	Activar Inactivar
Banco / Franquicia	INACTIVO	[Buttons]
No Documento de Pago	ACTIVO	[Buttons]
No Aprobacion	INACTIVO	[Buttons]
No PIN	INACTIVO	[Buttons]
No NIR	INACTIVO	[Buttons]
No CUS	INACTIVO	[Buttons]
Valor Documento	ACTIVO	[Buttons]
Valor Pagado	ACTIVO	[Buttons]
Fecha Documento	ACTIVO	[Buttons]

- **Opción Administración formas de pago**

Se creo en la opción de menú Administración / Pantallas Administrativas / Tarifas Pagos y Recibos, para el rol administrador nacional una nueva opción que permita abrir la pantalla “Administración Formas de Pago”.



- **Pantalla Administración de cuentas bancarias por circulo**

Se creo una pantalla de administración y creación de cuentas bancarias por circulo, para el rol administrador nacional, la cual se debe abrir desde una nueva opción en el menú Administración / Pantallas Administrativas / Tarifas Pagos y Recibos, dicha pantalla debe tener las opciones agregar, activar e inactivar.

- Desde esta pantalla se podrá crear las cuentas bancarias y asociarlas a un circulo en el sistema, además debe permitir activar o no dicha cuenta para el circulo.

- En la nueva pantalla se podrá visualizar en una lista desplegable los círculos que se encuentran en el sistema.
- En la nueva pantalla se podrá visualizar en una lista desplegable las entidades bancarias que ya están asociadas al círculo seleccionado.
- En la nueva pantalla se tendrá un campo de captura donde se digitará el número de cuenta que se va a asociar a la entidad bancaria.
- En la nueva pantalla debe tener una sección de listado que permite visualizar las cuentas parametrizadas para el círculo.

Círculo	Entidad bancaria :	Número de Cuenta :	Estado	Activar Inactivar
BUCARAMANGA BUCARAMANGA	BANCOLOMBIA CAJA SOCIAL	031-988498-45 074003906	ACTIVO INACTIVO	 

- **Opción Administración de cuentas bancarias por círculo**

Se creo una opción de menú Administración / Pantallas Administrativas / Tarifas Pagos y Recibos, para el rol administrador nacional una nueva opción que permita abrir la pantalla “Administración de cuentas bancarias por Círculo”.



- **Pantalla Administración de canales de recaudo**

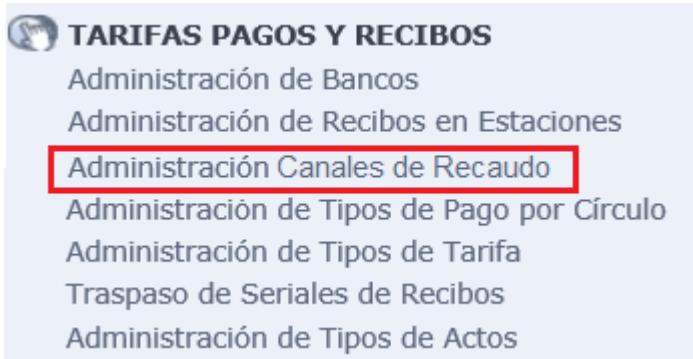
Se creo una pantalla de administración y creación de canales de recaudo, para el rol administrador nacional, la cual se debe abrir desde una nueva opción en el menú Administración / Pantallas Administrativas / Tarifas Pagos y Recibos, dicha pantalla debe tener las opciones agregar, activar e inactivar.

- Desde esta pantalla se podrá agregar los canales de recaudo al sistema, activar o inactivar dicho canal.
- La nueva pantalla tendrá un campo de captura donde se digitará el nombre del canal de recaudo que se va a agregar en el sistema.
- La nueva pantalla tendrá una sección del listado que permite visualizar los canales de recaudo creados en el sistema.

ID	Listado	Estado
01	CUENTA PRODUCTO	ACTIVO
02	LIQUIDADOR DE DERECHOS DE REGISTRO (VUR)	ACTIVO
05	SUPERGIROS	ACTIVO
06	RADICACION ELECTRONICA REL	ACTIVO
07	DATAFONO	ACTIVO
08	CAJA	INACTIVO

- **Opción Administración de canales de recaudo**

Se creó en el menú Administración / Pantallas Administrativas / Tarifas Pagos y Recibos una nueva opción que permita abrir la pantalla “Administración de canales de recaudo”.



- **Modificar opción Administración de tipos de pago por Circulo**

Se modifico en la opción de menú Administración / Pantallas Administrativas / Tarifas Pagos y Recibos, del rol Administrador Nacional, el texto de la opción administración de tipos de pago por circulo a “Administración de canales y formas de pago por circulo”.



## TARIFAS PAGOS Y RECIBOS

Administración de Bancos

Administración de Recibos en Estaciones

Administración Canales de Recaudo

Administración de Canales y Formas de Pago por Círculo

Administración de Tipos de Tarifa

Traspaso de Serials de Recibos

Administración de Tipos de Actos

- **Modificar pantalla “Configuración De Tipos de Pago Por Círculo”**

Se modifico la pantalla Configuración De Tipos de Pago Por Círculo a “configuración de canales y formas de pago por circulo”.

- Se agregó un campo de tipo desplegable que permita seleccionar los canales de recaudo existentes en el sistema.
- Se agregó un campo de tipo desplegable que permita seleccionar las entidades bancarias parametrizadas en el sistema.
- Se modificó el texto tipo de pago por forma de pago.
- Agregar canal de recaudo y entidad bancaria a la sección “listado”

The screenshot shows three main sections of the application interface:

- Top Section:** A header bar with the title "Pantalla Administrativa" and a sub-header "Configuración De Canales y Formas de pago por Círculo". Below it is a button labeled "Agregar Canal y Forma de pago".
- Middle Section (Agregar):** A form for adding new entries. It contains four dropdown menus labeled "Círculo Registrado", "Canal de Recaudo", "Entidad Bancaria", and "Forma de Pago", each with the placeholder text "-SELECCIONE UNA OPCIÓN-". Below the form is a "Agregar" button.
- Bottom Section (Listado):** Two tables for viewing existing configurations.
  - The first table has columns: "Círculo", "Canal de recaudo", "Forma de pago", and "Entidad Bancaria". It includes a "Listado" button and a "Eliminar" (Delete) button.
  - The second table is similar but lists two entries: "ARMENIA" and "LIQUIDADOR DE DERECHOS DE REGISTRO (VUR)" under "Canal de recaudo", "PAGO ELECTRONICO PSE" and "CONSIGNACION" under "Forma de pago", and "XXXXXXXXXXXX" and "BANCOLOMBIA" under "Entidad Bancaria". It also includes a "Listado" button and a "Eliminar" button.

- **Deshabilitar forma de pago en efectivo**

Se desabilito de la pantalla de registro de pago la opción “efectivo” que se encuentra por defecto en el sistema.

**Registrar Pago**

PAGO 0.823 Seg.

Liquidación

Valor derechos	\$ 3,000,000.00
Valor conservación documental	\$ 60,000.00
Valor impuestos	\$ 1.00
Valor multa	\$ 1.00
Valor certificados asociados	\$ 16,300.00
Valor otros impuestos	\$ 1.00
<b>Valor total ajustado a la centena por actos</b>	<b>\$ 3,076,300.00</b>

**Forma de Pago**

Efectivo	Valor Pagado \$ 3,076,300.00
----------	------------------------------

**Forma de Pago**

-SELECCIONE UNA OPCIÓN-

**Agregar Pago**

Registrar Pago Siguiente Editar Cancelar

### • Campo Canal de Recaudo

Se agrego en la pantalla registro de pago un nuevo campo “Canal de recaudo” que permita visualizar los canales asociados a cada ORIP.

**Registrar Pago**

PAGO 0.228 Seg.

Liquidación

Valor derechos	\$ 3,000,000.00
Valor conservación documental	\$ 60,000.00
Valor impuestos	\$ 1.00
Valor multa	\$ 1.00
Valor certificados asociados	\$ 16,300.00
Valor otros impuestos	\$ 1.00
<b>Valor total ajustado a la centena por actos</b>	<b>\$ 3,076,300.00</b>

**Canales de recaudo**

-SELECCIONE UNA OPCIÓN-

**Agregar Pago**

Registrar Pago Siguiente Editar Cancelar

### Formas de pago

Se realizó parametrización de las formas de pago para que al momento en que el funcionario seleccione el canal de recaudo asociado a la ORIP el sistema muestre las formas de pago por cada canal de recaudo, y a su vez el sistema muestre los campos de captura requeridos para la forma de pago seleccionada. Ver imagen:

**Registrar Pago**

PAGO 0.171 Seg.

Liquidación

Valor derechos	\$ 3,000,000.00
Valor conservación documental	\$ 60,000.00
Valor impuestos	\$ 1.00
Valor multa	\$ 1.00
Valor certificados asociados	\$ 16,300.00
Valor otros impuestos	\$ 1.00
<b>Valor total ajustado a la centena por actos</b>	<b>\$ 3,076,300.00</b>

**Canales de recaudo**

Cuenta producto	Forma de Pago
02 - CONSIGNACION	-SELECCIONE UNA OPCIÓN-

Banco/Franquicia

Nº Documento Pago

Fecha Documento

Valor Pagado \$

**Agregar Pago**

Registrar Pago Siguiente Editar Cancelar

- A continuación se detallan los campos de Canales de Recaudo y formas de pago agregados: Campos consignación

**Registrar Pago**

PAGO 0.171 Seg.

Liquidación

Valor derechos	\$ 3,000,000.00
Valor conservación documental	\$ 60,000.00
Valor impuestos	\$ .00
Valor multa	\$ .00
Valor certificados asociados	\$ 16,300.00
Valor otros impuestos	\$ .00
Valor total ajustado a la centena por actos	\$ 3,076,300.00

Canales de Recaudo

Cuenta producto	Forma de Pago
	02 - CONSIGNACION

Nueva o Banco/Franquicia  
Existe Saldó

Nº Documento Pago	Valor Documento	Fecha Documento	Valor Pagado \$
-SELECCIONE UNA OPCIÓN-			

Agregar Pago

- Campos Electrónico PSE

**Registrar Pago**

PAGO 3.299 Seg.

Liquidación

Valor derechos	\$ 3,000,000.00
Valor conservación documental	\$ 60,000.00
Valor impuestos	\$ .00
Valor multa	\$ .00
Valor certificados asociados	\$ 16,300.00
Valor otros impuestos	\$ .00
Valor total ajustado a la centena por actos	\$ 3,076,300.00

Canales de Recaudo

Cuenta producto	Forma de Pago	Banco/Franquicia
	09 - PAGO ELECTRONICO PSE	

Nº Aprobación

Fecha Documento

Valor Pagado \$

Agregar Pago

- Campos Tarjeta Débito

**Registrar Pago**

PAGO 3.299 Seg.

Liquidación

Valor derechos	\$ 3,000,000.00
Valor conservación documental	\$ 60,000.00
Valor impuestos	\$ .00
Valor multa	\$ .00
Valor certificados asociados	\$ 16,300.00
Valor otros impuestos	\$ .00
Valor total ajustado a la centena por actos	\$ 3,076,300.00

Canales de Recaudo

Cuenta producto	Forma de Pago	Banco/Franquicia
	12 - TARJETA DÉBITO	BANCO COLPATRIA - VISA

Nº Documento Pago

Nº Aprobación

Fecha Documento

Valor Pagado \$

Agregar Pago

- Pantalla ver detalles de turno

Se agregó en la pantalla ver detalles de turno el campo Canal de Recaudo.

Historial de fases del turno					
FASE	ATENDIDO POR	FECHA	RELACION FASE	RELACION SIGUIENTE	
SOLICITUD CONFRONTAR	USUARIO.PRUERAS	29/08/2018 08:37:49 PM			
		29/08/2018 08:37:55 PM			

Datos del solicitante

Nombre del Solicitante:	H	Telefono:	
Tipo de Documento del Solicitante:	SE	Número de Documento del Solicitante:	155502457

Pagos asociados

Valor derechos de registro:	3,000,000	Valor certificados asociados:	16,300
Valor de impuestos:	0	Valor de multa:	0
Otro impuesto:		Valor otro impuesto:	0
Valor Conservación Documental	60,000		

Canal de Recaudo

<b>CUENTA PRODUCTO</b>	
------------------------	--

Forma de Pago:	Banco/Franquicia:	Número Aprobación:	Fecha Documento:	Valor Pagado:	
PAGO ELECTRONICO PSE	BANCAMIA - CREDENCIAL	8589666	29/08/2018	76,300	
Forma de Pago:	Banco/Franquicia:	No. Documento Pago:	No. Aprobación:	Fecha Documento:	Valor Pagado:
TARJETA DEBITO	BANCO AGRARIO - CREDENCIAL	96696699	9699999	29/08/2018	1,000,000
Forma de Pago:	Banco/Franquicia:	No. Documento P			

2018-999-6-140	Eliminar turno anterior			
Turnos de Certificados Asociados				
2018-999-1-184				
Datos Asociados al Pago				
Canal de Recaudo				
<b>DATAFONO</b>				
()	Forma de Pago	TARJETA CREDITO	Valor	1,000,000.00
	Banco		Número	
	Saldo a favor			
()	Forma de Pago	TARJETA DEBITO	Valor	1,000,000.00
	Banco		Número	
	Saldo a favor			
Canal de Recaudo				
<b>CUENTA PRODUCTO</b>				
()	Forma de Pago	CONSIGNACIÓN	Valor	1,000,000.00
	Banco	01	Número	366699
	Saldo a favor	.00		

## • Pantalla Análisis

Se agregó en la pantalla de análisis, por el rol usuario Administrativo / Análisis, el campo Canal de Recaudo.

2018-999-6-140	Eliminar turno anterior			
Turnos de Certificados Asociados				
2018-999-1-184				
Datos Asociados al Pago				
Canal de Recaudo				
<b>DATAFONO</b>				
()	Forma de Pago	TARJETA CREDITO	Valor	1,000,000.00
	Banco		Número	
	Saldo a favor			
()	Forma de Pago	TARJETA DEBITO	Valor	1,000,000.00
	Banco		Número	
	Saldo a favor			
Canal de Recaudo				
<b>CUENTA PRODUCTO</b>				
()	Forma de Pago	CONSIGNACIÓN	Valor	1,000,000.00
	Banco	01	Número	366699
	Saldo a favor	.00		

## • Pantalla pago mayor valor

Se desactivó de la pantalla pago mayor valor la forma de pago en efectivo y se agregó el campo canal de recaudo de la misma manera en que se agregó en las ventanas de pago de registro.

- Pantalla pago certificados y fotocopias**

Se desactivó de la pantalla pago certificados y fotocopias la forma de pago en efectivo y se agregó el campo canal de recaudo de la misma manera en que se agregó en las ventanas de pago de registro.

- Recibos**

Se agregó en los imprimibles la línea donde discrimine el canal de recaudo, manteniendo la estructura de formas de pago.

- Recibo de registro**

**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS**  
**CIRCULO\_PRUEBAS - NIT: 899999007-0**  
**SOLICITUD REGISTRO DOCUMENTOS**  
 Impreso el 29 de Agosto de 2018 a las 08:20:22 pm

**1236593054**

No. RADICACIÓN: **2018-999-6-524**

NOMBRE DEL SOLICITANTE: H  
 DOCUMENTO No.: 8885 del 29/8/2018 CONSULADO DE ESTADOS UNIDOS DE AMERICA de BOGOTA D. C.

**MATRICULAS: 999-8969**

ACTOS A REGISTRAR:

Tipo....	Código....	Cuantía....	Tipo Tarifa	Derecho	Impuesto
VENTA	1	400.000.000	N	\$	3.000.000 \$0

**TURNOS CERTIFICADOS ASOCIADOS: 1**

VALOR MULTA: \$0

**CANAL DE RECAUDO XXXXX**

FORMA DE PAGO:

TARJETA CREDITO VALOR: \$ 1.000.000

TARJETA DEBITO VALOR: \$ 1.000.000

CONSIG. BANCO: BANCO DE BOGOTA Nro: 8555855 FECHA CONSIG.: 29/08/2018 VALOR PAGADO:

\$983.700 VALOR DOC.: \$1.500.000

PAGO ELECTRONICO PSE VALOR: \$ 76.300

VALOR DERECHOS: \$3.000.000

Conservación documental del 2% \$ 60.000

- **Recibo de certificados**

**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS**  
**NIT: 899999007-0 CIRCULO\_PRUEBAS**  
**SOLICITUD CERTIFICADO DE TRADICION**  
 Impreso el 29 de Agosto de 2018 a las 08:20:30 pm

**1236593055**

No. RADICACIÓN: **2018-999-1-625**

Asociado al turno de registro: 2018-999-6-524

**MATRICULA: 999-8969**

NO SE EXPIDE DE INMEDIATO EL CERTIFICADO

NOMBRE DEL SOLICITANTE: H

CERTIFICADOS: 1

**CANAL DE RECAUDO XXXXX**

FORMA DE PAGO:

CONSIG. BANCO: 01 Nro: 8555855 FECHA CONSIG.: 29/08/2018 VALOR PAGADO: \$16.300 VALOR DOC.: \$1.500.000

VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: \$ 16.300

EL CERTIFICADO SE EXPIDE DE ACUERDO A LOS DATOS SUMINISTRADOS

Usuario: 51856

- **Recibo pago Mayor valor**

**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS**  
**CIRCULO\_PRUEBAS - NIT: 899999007-0**  
**RECIBO DE PAGO DE MAYOR VALOR**  
 Impreso el 22 de Abril de 2018 a las 05:04:20 am

**62483251**

No. RADICACIÓN: **2018-999-6-109**

NOMBRE DEL SOLICITANTE: HG SOLANO PRUEBAS SIR PRUEBAS V CC# 1111111  
 TELÉFONO: 79261492

**Canal de Recaudo: XXXXXXXXX**

FORMA DE PAGO:

CONSIG. BANCO: BANCO COLPATRIA Nro: 123468 FECHA CONSIG.: 22/4/2018 VALOR: \$ 10.000

EFFECTIVO VALOR: \$ 10.000

VALOR DERECHOS: \$20.000

VALOR CERTIFICADOS: \$20.000

CONSERVACION DOCUMENTAL 2: \$ 400

VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS: \$ 20.400

JUSTIFICACION PAGO MAYOR VALOR:

PRUEBA 5 AM

Usuario: 51856

- Recibo Constancia de solicitud de devolución de dinero

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
CIRCULO\_PRUEBAS

NIT:  
CONSTANCIA DE SOLICITUD DE DEVOLUCIÓN DE DINERO  
Impreso el 10 de Septiembre de 2018 a las 05:02:51 pm

No. RADICACIÓN: 2018-999-4-94

SOLICITANTE(S)

NOMBRE: BELTRAN MORENO ANDRES

DATOS SOLICITUD

Turno Asociado: 2018-999-6-529

Descripción: PRUEBAAAAAA

DATOS PAGO INGRESADOS

Canal de Recaudo xxxxxxxxxx

Formas de pago

PAGO No. 1: EFECTIVO

Valor

\$1582700

TOTAL: \$1582700

DOCUMENTOS ANEXADOS:

RECIBO DE CAJA

FOTOCOPIA DOCUMENTO IDENTIDAD SOLICITANTE

NUMERO DE FOLIOS: 1

- Recibo de pago otros recaudos “Fotocopias”

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
CIRCULO\_PRUEBAS

NIT: 899999007-0

62483341

Recibo de Pago de otros recaudos

Impreso el 10 de Septiembre de 2018 a las 04:46:16 pm

No. RADICACION: 2018-999-5-21

Datos de Pago:

Canal de Recaudo: XXXXXXXXXX

FORMA DE PAGO:  
EFECTIVO

VALOR: \$ 600

VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: \$ 600

Solicitante: ANDRES BELTRAN MORENO  
Teléfono: 21514548  
Radicado Por: 51856  
Usuario: 51856

- Reportes

Se ajusto en el reporte 99 en la sesión totales quitando las líneas total cheque y total convenio.

<b>TOTALES</b>	
<b>Valor Conservacion Documental:</b>	
<b>TIPO PAGO</b>	<b>TOTAL</b>
<b>Total Efectivo:</b>	0
<b>Total Consignación:</b>	5,027,400
<b>Total Cheque:</b>	0
<b>Total Convenio:</b>	0
<b>Total Pago Electrónico PSE:</b>	152,600
<b>Total Tarjeta Crédito:</b>	2,000,000
<b>Total Tarjeta Débito:</b>	2,000,000
<b>Total Otros Pagos:</b>	0
<b>Total Oficina Vr:</b>	9,180,000
<b>Total Otras ORIPS Vr:</b>	0
<b>Total:</b>	9,180,000

Se realizaron los ajustes en los reportes 18, 19, 25, 99,180 agregando el campo canal de recaudo y eliminando la columna impuesto.



**DESPLIEGUE 7.3.3  
06/09/2018**



### **2283 CAMBIAR NUMERO DE COPIAS DE NOTAS DEVOLUTIVAS**

En el aplicativo SIR para el proceso de registro, se genera un turno, este al llegar a la fase de calificación y al realizarle una nota devolutiva; ya que no es procedente su inscripción, el sistema despliega una ventana donde se le agrega la nota devolutiva y se muestra el número de copias a imprimir dos (2) copias, al dar clic en el botón terminar el sistema imprime dos (2) copias de las notas devolutivas, como se ve en la figura 1 y 2.

Figura 1

The screenshot shows the SIR application interface. On the left, there is a vertical menu with several options: Revisión y Análisis, Información Turno, Información Folio, Información Registro, Inscripción, Parcial, Devolución (which is highlighted with a red box), Digitador Masivo, and Opciones. The main area has two sections: 'Revisión encabezado' and 'Revisión confrontación'. In the 'Revisión encabezado' section, there are fields for Datos Básicos (Tipo: 01 - ESCRITURA, Número: 1, Fecha: 22/01/2019), Oficina de Procedencia (Codigo: 1, Departamento: BOGOTA D C, Nombre: NOTARIA - NOTARIA PRIMERA, Municipio: BOGOTA D.C.), and a 'Corrección Encabezado' button. In the 'Revisión confrontación' section, there is a field for Envíar Confrontación (999-2132) and a 'Motivo Envío' dropdown. At the bottom, there is a 'Revisión finalización' button.

Figura 2

**Ingresar Nota Devolutiva**

Notas Devolutivas

**Nota**

Escoger Tipo Nota Devolutiva: 150 - \*\*\*\* AFECCION A VIVIENDA FAMILIAR \*\*\*\*

Escoger Subtipo Nota Devolutiva: 153 - EL NOTARIO OMITIO INDAGAR SOBRE LA AFECCION A VIVIENDA FAMILIAR

EL NOTARIO OMITIO INDAGAR SOBRE LA AFECCION A VIVIENDA FAMILIAR (LEY 258 DE 1996 Y LEY 854 DE 2003).

Agregar

**Descripción**

PRUEBAS

**Causales de Devolución**

FALTA PAGO DEL IMPUESTO DE REGISTRO (LEY 223/1995 Y DECRETO 650/96).  
EL NOTARIO OMITIO INDAGAR SOBRE LA AFECCION A VIVIENDA FAMILIAR (LEY 258 DE 1996 Y LEY 854 DE 2003).

Número de copias a imprimir: 2

Terminar Salir



**DESPLIEGUE 7.3.2**  
**04/07/2018**



### 2169 HABILITAR LA CONSULTA DE TURNOS ASOCIADOS

En la opción ver detalles del turno, cuando se consulta un turno siguiente, turno anterior o turnos asociados en el aplicativo SIR, el sistema permite realizar una consulta preliminar sobre el turno relacionado en el mismo formulario.

Teniendo en cuenta esto al realizar la consulta de turnos por la pantalla administrativa “Ver Detalles de Turnos”, se realizaron los siguientes ajustes:

1. Se inserta un botón de vista previa donde se visualiza la información del turno asociado de un turno consultado previamente.

2016-999-6-79	
---------------	--

Así, el usuario no tendrá que salir y volver a ingresar para consultar el turno asociado de los siguientes procesos registrales:

- ✓ Certificados (1)
  - Antiguo Sistema, Pertenencia, Ampliación Tradición, campos “Turno Anterior” y “Turno Siguiente”.

- Asociados, campo “Turnos Asociados”.
  - Inmediatos asociados a Masivos, campo “Turnos Asociados”.
- ✓ Correcciones (3), campo “Turno testamento”.
- ✓ Devoluciones (4), campo “Turno Anterior”.
- ✓ Registro (6), campos “Turno Anterior” y “Turno Siguiente”, turnos de certificado asociados
- ✓ Restitución Reparto Notarial (16), campo “Turno de reparto asociado”.
2. Para el proceso de Certificados Masivos en el campo “Turnos Asociados” se creo un botón, para cada turno de  certificado inmediato hijo donde al dar clic sobre él, se visualiza la información del turno asociado de un turno consultado previamente.
- ✓ Certificados Masivos (20), campo “Turnos Asociados”.

Se ajusta la pantalla de consulta de turnos a la cual se ingresa por la ruta: Pantallas administrativas/Manejo de turnos/ver detalle de turno y que, tras seleccionar el turno principal, en la sección donde se muestra los turnos asociados se crea un botón para cada turno asociado, que permita visualizar en una  nueva ventana el detalle de cada turno hijo asociado; como se muestra en la imagen:

Detalles de Turno 																					
Datos basicos de Turno																					
Número del Turno	2017-999-20-13																				
Fecha de Radicación	04/08/2017																				
Tipo de Certificado																					
Tipo de Tarifa																					
Número de Recibo	4																				
Valor Pagado	0																				
Valor por Pagar	0																				
Valor Devuelto	0																				
Valor por Devolver	0																				
Estado Actual	TURNO ENTREGADO.																				
Resolución																					
Relación Actual	NO ESTA RELACIONADO																				
Fase Relacion Actual	NO ESTA RELACIONADO																				
Turnos Asociados																					
<table border="1"> <tr><td>2017-999-1-160</td><td></td><td>2017-999-1-105</td><td></td></tr> <tr><td>2017-999-1-158</td><td></td><td>2017-999-1-108</td><td></td></tr> <tr><td>2017-999-20-12</td><td></td><td></td><td></td></tr> <tr><td>2017-999-1-158</td><td></td><td></td><td></td></tr> <tr><td>2017-999-1-159</td><td></td><td></td><td></td></tr> </table>		2017-999-1-160		2017-999-1-105		2017-999-1-158		2017-999-1-108		2017-999-20-12				2017-999-1-158				2017-999-1-159			
2017-999-1-160		2017-999-1-105																			
2017-999-1-158		2017-999-1-108																			
2017-999-20-12																					
2017-999-1-158																					
2017-999-1-159																					

Una vez el usuario de clic sobre el botón de cualquier turno asociado debe desplegarse la ventana con la información del turno asociado que fue seleccionado, En los casos en que el turno consultado tenga otros turnos asociados (turno hijos) y se quiere consultar uno de estos, el sistema debe mostrar la información del nuevo turno hijo consultado, La funcionalidad debe contar con un botón regresar que permita retornar a la pantalla principal, tal como se muestra en la imagen:

**Detalles de Turno**

**Datos básicos de Turno**

Número del Turno	2017-999-1-160
Fecha de Radicación	04/08/2017
Tipo de Certificado	EXENTO
Tipo de Tarifa	EXENTO
Número de Recibo	
Valor Pagado	0
Valor por Pagar	0
Valor Devuelto	0
Valor por Devolver	0
Estado Actual	TURNO ENTREGADO.
Resolución	EXPEDIDO
Relación Actual	NO ESTA RELACIONADO
Fase Relacion Actual	NO ESTA RELACIONADO

**Turnos Asociados** 2017-999-20-13

**Regresar a consulta de Turno**

**Regresar**

**Consulta de Detalles de Turnos**

**Datos del Turno**

<input checked="" type="radio"/> Número de Turno	2017-100-3-325
<input type="radio"/> Número de Matrícula	

**Buscar** **Volver**

Cuando se consulte un turno que a su vez tenga un turno asociado, y este a su vez tenga otro turno asociado, el sistema debe permitir navegar por estos turnos asociados mediante un botón “ver” y contar con un botón “Volver” que le permita al usuario regresar a la pantalla de consulta del turno inmediatamente anterior, de donde se lanzó la consulta, tal como se muestra en la imagen.

**Detalles de Turno**

**Datos básicos de Turno**

Número del Turno	2017-999-1-160
Fecha de Radicación	04/08/2017
Tipo de Certificado	EXENTO
Tipo de Tarifa	EXENTO
Número de Recibo	
Valor Pagado	0
Valor por Pagar	0
Valor Devuelto	0
Valor por Devolver	0
Estado Actual	TURNO ENTREGADO.
Resolución	EXPEDIDO
Relación Actual	NO ESTA RELACIONADO
Fase Relacion Actual	NO ESTA RELACIONADO

**Turnos Asociados** 2017-999-20-13

**Regresar a consulta de Turno**

**Regresar**

**Volver** **Consultar**

## 2196 CONSULTA TURNO CON TESTAMENTO

En el aplicativo SIR para todos los usuarios que tengan la opción de consulta de turno a través de pantallas administrativas por la opción de Ver detalle de turno, una vez se consulte el turno de testamento debe presentarse un botón “Ver detalle” ubicado después de la sección Matriculas Asociadas, que permita hacer la consulta en pantalla así (imagen1), es de aclarar que el botón se presenta solo para aquellos turnos de registro

que tengan como Subtipo de Solicitud - Testamento, datos que fueron ingresados en el proceso 6 - Registro . Una vez el usuario de un clic en este botón “Ver detalle” el aplicativo debe mostrar la misma información inscrita en la fase de calificación de testamento (Datos testador, Tomo , Número de anotación, fecha, revoca escritura y observaciones), como se ve en (imagen 2), si el texto es muy extenso se mostrara con un scroll.

Imagen 1:

Imagen 2:

Cuando el usuario desee ocultar la información desplegada, el aplicativo debe permitirle ocultar este campo cerrando la nueva ventana mediante el botón “ver detalles” o permitiéndole regresar.

Esta nueva ventana solo es de visualización no se permitir imprimir ni modificar.

## **2199 RESTRINGIR LA IMPRESION DE CERTIFICADOS ASOCIADOS CON TIPO DE CALIFICACION INSCRIPCION PARCIAL**

En el aplicativo SIR para el proceso de calificación se realiza inscripción sobre alguna de las matrículas, y a su vez se hace Nota Devolutiva en otra matricula según corresponda, como se muestra en las siguientes imágenes.

Imagen 1:

Para la matrícula 999-8585 se realiza inscripción en el folio de matrícula es decir se le agrega una o varias anotaciones.

Imagen 2:

Para la matrícula 999-9682, no se hace modificaciones es decir no se le agrega anotaciones, pero se agrega una nota devolutiva por la opción inscripción parcial.

Al avanzar el turno, en el rol certificados asociados se toma el turno y el sistema solo debe mostrar en pantalla la(s) matrícula(s) a la(s) que se le hizo inscripción y permitir imprimir los certificados, las matrículas a las que no se les hizo inscripción, es decir a las que se les hizo nota devolutiva, por la opción inscripción parcial, no se deben mostrar en pantalla, ni permitir imprimir los certificados, como se ve en la imagen 3.

Imagen 3:

The screenshot shows a software interface for managing associated certificates. At the top, there's a header 'Certificados Asociados'. Below it, a message box says 'Respuesta de Calificación' and 'INSCRITO PARCIALMENTE'. A table lists certificates with columns for 'Certificado' and 'Matrícula'. There are buttons for 'Aceptar' (Accept), 'Imprimir' (Print), and 'NOTAS'. The status bar at the bottom of the window displays the title '2197 CONSULTA FOLIO TODAS LAS PESTANAS'.

Al realizar la consulta de folios el sistema debe mostrar todas las pestañas de consulta de información de la matrícula mediante una ventana emergente, que tendrá la opción de cerrarse mediante el botón cerrar ventana,



La modificación tiene como alcance los siguientes botones:

Vista previa	
Revisar Folio	
Consultar Folio	

Los procesos y pantallas administrativas del aplicativo SIR donde se deberán realizar los cambios se detallan a continuación:

ROL	FASE	NOMBRE DEL BOTÓN
<b>Proceso: Registro (6)</b>		
DIGITADOR	DIGITACION CALIFICACIONES	
COORDINADOR JURIDICO	APROBAR CORRECCION CALIFICACION	Revisar Folio
<b>Proceso: Correcciones (3)</b>		
AUXILIAR DE CORRECCIONES	EJECUTAR CORRECCION SIMPLE Pestaña Consultas y/o Matriz Segregaciones	Consultar
AUXILIAR ACTUACIONES ADMINISTRATIVAS	EJECUTAR ACTUACIONES ADMINISTRATIVAS Pestaña Consultas y/o Matriz Segregaciones	Consultar
<b>Proceso: Devoluciones (4)</b>		
USUARIO ADMINISTRATIVO	ANALISIS RECURSOS DEVOLUCION RESOLUCION	Detalles de turno /Vista Previa
PAGO DEVOLUCION	PAGO DEVOLUCION	
<b>Pantallas Administrativas</b>		
VER DETALLES DE TURNOS	NÚMERO DE TURNO	Vista Previa
	NÚMERO DE MATRICULA	Consultar
	REABRIR FOLIOS	Consultar

### Proceso: Registro (6)

Sistema de Información Registral - [Desarrollo-7.2.0](#)  
Fecha Actual: 07/6/2018

The screenshot shows a software window titled 'Consulta - Folio'. At the top, there is a toolbar with several buttons: 'Detalles Folio', 'Matriz/Segregaciones' (which is highlighted with a red box), 'Gravamen', 'Salvedades', and 'Cancelaciones'. Below the toolbar is a sub-menu titled 'Ver folio consulta' with a small icon. At the bottom of the window, there is a status bar displaying a magnifying glass icon, the number '999-456', and other unidentifiable text.

### Proceso: Correcciones (3)

Sistema de Información Registral - [Desarrollo-7.2.0](#)  
Fecha Actual: 07/6/2018

The screenshot shows a software window titled 'Consulta - Folio'. At the top, there is a toolbar with several buttons: 'Detalles Folio', 'Matriz/Segregaciones' (which is highlighted with a red box), 'Gravamen', 'Salvedades', and 'Cancelaciones'. Below the toolbar is a sub-menu titled 'Ver folio consulta' with a small icon. At the bottom of the window, there is a status bar displaying a magnifying glass icon, the number '999-456', and other unidentifiable text.

### Proceso: Devoluciones (4)

Sistema de Información Registral - [Desarrollo-7.2.0](#)  
Fecha Actual: 07/6/2018

The screenshot shows a software window titled 'Consulta - Folio'. At the top, there is a toolbar with several buttons: 'Detalles Folio', 'Matriz/Segregaciones' (which is highlighted with a red box), 'Gravamen', 'Salvedades', and 'Cancelaciones'. Below the toolbar is a sub-menu titled 'Ver folio consulta' with a small icon. At the bottom of the window, there is a status bar displaying a magnifying glass icon, the number '999-456', and other unidentifiable text.

## Pantallas Administrativas

Sistema de Información Registral - [Desarrollo-7.2.0](#)  
Fecha Actual: 07/6/2018



**DESPLIEGUE 7.3.1**  
**21/06/2017**



### **2230 AJUSTE REPORTES DE TARIFAS REGISTRARLES 2018**

Para la generación de los reportes hechos en el aplicativo SIR se corrigió la información mostrada, incluye el cobro de conservación documental implementado en el anterior despliegue, los reportes afectados son:

**REPORTE 5: DIARIO MAYORES VALORES DEL DIA**



**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS**  
**DIARIO MAYORES VALORES DEL DIA**

**CIRCULO\_PRUEBAS - 999**  
**18-06-2018**

Impreso el 19 de Junio de 2018 a las 13:50:44 PM

Página 1

**MAYORES VALORES**

Radicacion	Valor	Efectivo	Cheque	Consignacion	T. Crédito	T. Debito	Derechos	T.Conservación Documental	Impuestos	Hora Pago
2018-999-6-426	\$ 86,700	\$ 86,700	\$ 0	\$ 0	\$ 0	\$ 0	\$ 85,000	\$ 1,700	\$ 0	07:27:46 PM
2018-999-6-428	\$ 85,700	\$ 85,700	\$ 0	\$ 0	\$ 0	\$ 0	\$ 84,000	\$ 1,700	\$ 0	07:28:00 PM
2018-999-6-435	\$ 516,100	\$ 516,100	\$ 0	\$ 0	\$ 0	\$ 0	\$ 506,000	\$ 10,100	\$ 0	07:39:50 PM
<b>Totales:</b>	<b>\$ 688,500</b>	<b>\$ 688,500</b>	<b>\$ 0</b>	<b>\$ 0</b>	<b>\$ 0</b>	<b>\$ 0</b>	<b>\$ 675,000</b>	<b>\$ 13,500</b>	<b>\$ 0</b>	
<b>Total Derechos Ajustados a la Centena:</b>		\$ 675,000								
<b>Total Impuestos:</b>		\$ 0								
<b>Total Conservación Documental:</b>		\$ 13,500								
<b>TOTAL MAYORES VALORES REGISTRO</b>		\$ 688,500								
<b>TOTAL MAYORES VALORES</b>		\$ 702,000								
<b>Totales:</b>	<b>\$ 688,500</b>	<b>\$ 688,500</b>	<b>\$ 0</b>	<b>\$ 0</b>	<b>\$ 0</b>	<b>\$ 0</b>	<b>\$ 675,000</b>	<b>\$ 13,500</b>	<b>\$ 0</b>	

**REPORTE 9: DIARIO RADICADOR DE DOCUMENTOS DEL DIA**



OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999  
DIARIO RADICADOR DE DOCUMENTOS DEL DÍA 18-06-2018

Impreso el 19 de Junio de 2018 a las 13:53:19 PM

REALIZADO POR: USUARIO.PRUEBAS

Página 1

Hora	Turno Ant	Turno	Acto	Nro Doc.	Fecha	Oficina	Ciudad	T.Conservación Documental	Total derechos ajustados	Recibo	Cajero
18:54:54		2018-099-8-418	VENTA,ACTUALIZACION DE NOMENCLATURA	(2) 215454	18-06-2018	JUZGADO PROMISCUO DEL CIRCUITO	SOACHA	\$ 75,400	\$ 3,769,700	1236592763 51856	
19:02:51		2018-099-8-419	VENTA,ARRENDAMIENTO	(2) 213123	18-06-2018	FISCALIA GENERAL DE LA NACION- FISCALIA SECCIONAL 03	SOACHA	\$ 81,700	\$ 4,085,300	1236592766 51856	
19:04:50		2018-099-8-420	VENTA	(1) 452752	18-06-2018	FISCALIA GENERAL DE LA NACION- FISCALIA SECCIONAL 03	SOACHA	\$ 30,000	\$ 1,500,000	1236592769 51856	
19:05:02		2018-099-8-421	VENTA,HIPOTECA,VENTA	(3) 253532	18-06-2018			\$ 108,700	\$ 5,433,600	1236592772 51856	
19:06:02		2018-099-8-422	VENTA,CANCELACION AFECTACION FAMILIAR	(2) 1234	18-06-2018	JUZGADO DE EJECUCION DE PENAS Y MEDIDAS DE SEGURIDAD	SOACHA	\$ 130,900	\$ 6,544,700	1236592774 51856	
19:06:09		2018-099-8-423	VENTA	(1) 7272	18-06-2018			\$ 80,000	\$ 3,000,000	1236592777 51856	
19:07:12		2018-099-8-424	VENTA,DONACION INTERNACIONAL	(2) 989776	18-06-2018	JUZGADO TERCERO CIVIL MUNICIPAL DE DESCONGESTION DE SUBA	SUBA	\$ 1,480,900	\$ 74,044,700	1236592779 51856	
19:08:25		2018-099-8-425	VENTA,CONSTITUCION SOCIEDAD COMERCIAL	(2) 1234	18-06-2018	JUZGADO PRIMERO CIVIL DEL CIRCUITO	SOACHA	\$ 54,700	\$ 2,735,300	1236592782 51856	
19:09:32		2018-099-8-426	VENTA,INSCRIPCION DE	(2) 1234	18-06-2018	OFICINA DE	LA DORADA	\$ 1,100	\$ 55,000	1236592785 51856	
			OFERTA			REGISTRO DE INSTRUMENTOS PUBLICOS DE LA DORADA					
19:10:48		2018-099-8-427	VENTA,ADMINISTRACION ANTICRETICA	(2) 123456	18-06-2018	FISCALIA GENERAL DE LA NACION- FISCALIA SECCIONAL 03	SOACHA	\$ 11,400	\$ 570,500	1236592788 51856	
19:12:00		2018-099-8-428	VENTA,INSCRIPCION FOLIO ADICIONAL	(2) 9876	18-06-2018	FISCALIA GENERAL DE LA NACION- FISCALIA SECCIONAL 03	SOACHA	\$ 148,300	\$ 7,412,900	1236592791 51856	
19:13:52	2018-099-8-421	2018-099-8-429	VENTA,HIPOTECA,HIPOTECA ,VENTA	(4) 253532	18-06-2018			\$ 112,500	\$ 5,625,000	1236592794 51856	
19:15:15	2018-099-8-423	2018-099-8-430	VENTA,HIPOTECA	(2) 7272	18-06-2018			\$ 52,500	\$ 2,625,000	1236592796 51856	
19:16:39		2018-099-8-431	ANULADO					\$ 0	\$ 0	1236592798 51856	
19:17:59		2018-099-8-432	ANULADO					\$ 0	\$ 0	1236592800 51856	
19:27:20		2018-099-8-433	HIPOTECA	(1) 22	18-06-2018			\$ 45,000	\$ 2,250,000	1236592802 51856	
19:29:25		2018-099-8-434	VENTA	(1) 2568	18-06-2018			\$ 45,000	\$ 2,250,000	1236592804 51856	
19:31:12		2018-099-8-435	VENTA	(1) 2552	18-06-2018			\$ 45,000	\$ 2,250,000	1236592806 51856	
20:17:16		2018-099-8-436	VENTA	(1) 345345	18-06-2018	FISCALIA GENERAL DE LA NACION- FISCALIA SECCIONAL 03	SOACHA	\$ 2,400	\$ 122,400	1236592811 51856	
21:50:35		2018-099-8-437	VENTA	(1) 1558	18-06-2018			\$ 60,000	\$ 3,000,000	1236592814 51856	
								Total derechos ajustados a la centena:	\$ 127,274,100		
								Total Turnos:	18		
								Total Conservación Documental:	\$ 2,545,500		
								Total Anulados:	2		

REPORTE 15 : INFORME CUENTAS POR PAGAR DEVOLUCIONES

**Superintendencia de Notariado y Registro**

Ministerio del Interior y de Justicia

República de Colombia

www.supernotariado.gov.co

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999

INFORME CUENTAS POR PAGAR DEVOLUCIONES, DESDE 18-06-2018 HASTA 18-06-2018

Impreso el 19 de Junio de 2018 a las 13:54:29 PM

Página 1

Concepto: REGISTRO

Radicación	Fecha Radicación	Turno Asociado	Num Consignación -cheque	Valor Derechos	Valor Impuestos	Valor Certificados	Pago Exceso	Valor a Pagar	T.Conservación Documental	Número Resolución	Fecha Resolución	Solicitantes
2018-999-4-80	18/06/2018	2018-999-5-436		\$ 102,000	\$ 0	\$ 0	\$ 0	\$ 102,000	\$ 2,500	5345345	19/06/2018	155502079, BELTRAN MORENO ANDRES;
<b>TOTAL REGISTRO:</b>				<b>\$ 102,000</b>	<b>\$ 0</b>	<b>\$ 0</b>	<b>\$ 0</b>	<b>\$ 102,000</b>				
<b>TOTALES</b>				<b>\$ 102,000</b>	<b>\$ 0</b>	<b>\$ 0</b>	<b>\$ 0</b>	<b>\$ 102,000</b>				

**REPORTE 16: INFORME CUANTAS POR PAGAR MAYORES VALORES****Superintendencia de Notariado y Registro**

Ministerio del Interior y de Justicia

República de Colombia

www.supernotariado.gov.co

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999

INFORME CUENTAS POR PAGAR MAYORES VALORES, DESDE 18-06-2018 HASTA 18-06-2018

Impreso el 19 de Junio de 2018 a las 13:55:32 PM

Página 1

Concepto: Registro

Radicación	Fecha	Liquidador	Vr. Derechos	T.Conservación Documental	Vr. Impuestos	Valor a Pagar
2018-999-6-425	18-06-2018 07:16:48 PM	51856	\$ 50,000	\$ 1,000	\$ 0	\$ 51,000
2018-999-6-427	18-06-2018 07:18:02 PM	51856	\$ 64,000	\$ 1,300	\$ 0	\$ 65,300
Total Registro			\$ 114,000	\$ 2,300	\$ 0	\$ 116,300

**RESUMEN GENERAL POR CONCEPTOS**

Concepto	Vr. Derechos	Vr. Impuestos	Valor a Pagar
REGISTRO	\$ 114,000	\$ 0	\$ 116,300
Tarifa Conservación Documental:	\$ 2,300	\$ 0	\$ 2,300
Total General:	\$ 65,000	\$ 0	\$ 0

**REPORTE 18: INFORME DE DEVOLUCIONES PAGADAS**



**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999**  
**INFORME DE DEVOLUCIONES PAGADAS (FORMA DE PAGO) , DESDE 18-06-2018 HASTA 18-06-2018**  
 Impreso el 19 de Junio de 2018 a las 13:57:10 PM

**REALIZADO POR:** USUARIO.PRUEBAS

Página 1

<b>Concepto: REGISTRO</b>											
Radicación	Fecha de Pago	Turmo Asociado	Valor Devuelto	Solicitantes	Derechos	Impuestos	Certificados	Número Resolución	Fecha Resolución	Tipo Pago	Num Doc
2018-999-4-75	18/06/2018	2018-999-6-419	\$4,199,600.00	2154, BELTRAN MORENO ANDRES;	\$4,167,000.00	\$0.00	\$32,600.00	78979	18/06/2018	EFFECT.	
2018-999-4-76	18/06/2018	2018-999-6-420	\$1,562,600.00	155502073, BELTRAN MORENO ANDRES;	\$1,530,000.00	\$0.00	\$32,600.00	98765	18/06/2018	EFFECT.	
2018-999-4-79	18/06/2018	2018-999-6-435	\$1,705,400.00	80844585, HG SOLANO;	\$1,689,100.00	\$0.00	\$16,300.00	4444	18/06/2018	EFFECT.	
<b>Totales REGISTRO:</b>				\$7,386,100.00		\$0.00	\$81,500.00				
<b>Totales:</b>				\$7,386,100.00		\$0.00	\$81,500.00				

**RESUMEN**

Total Efectivo:	7,467,600.00
Total Cheque:	0.00
Total Consignación:	0.00
Total Nota Débito:	0.00

**REPORTE 19: INFORME DE MAYORES VALORES**



**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999**  
**INFORME DE MAYORES VALORES PAGADOS, DEL DIA**  
 18-06-2018 HASTA 18-06-2018

Impreso el 19 de Junio de 2018 a las 13:58:09 PM

Página 1

<b>Concepto: Registro</b>											
Radicación	Fecha de Pago	Cajero	Recibo	Vr. Derechos	Vr.Conservación Documental	Vr. Impuestos	Vr. Certificados	Vr. Efectivo	Vr. Consig.	Vr. Cheque.	Vr. Pago
2018-999-6-426	18-06-2018 19:27:46	51856	62483332	\$ 85,000	\$ 1,700	\$ 0	\$ 66,700	\$ 66,700	\$ 0	\$ 0	\$ 66,700
2018-999-6-428	18-06-2018 19:28:00	51856	62483333	\$ 84,000	\$ 1,700	\$ 0	\$ 65,700	\$ 65,700	\$ 0	\$ 0	\$ 65,700
2018-999-6-435	18-06-2018 19:39:50	51856	62483334	\$ 506,000	\$ 10,100	\$ 0	\$ 516,100	\$ 516,100	\$ 0	\$ 0	\$ 516,100
<b>TOTAL Registro:</b>				\$ 675,000	\$ 13,500	\$ 0	\$ 688,500	\$ 688,500	\$ 0	\$ 0	\$ 688,500
<b>Concepto</b>											
<b>REGISTRO</b>				\$ 675,000	\$ 0	\$ 688,500	\$ 0	\$ 688,500	\$ 0	\$ 688,500	\$ 688,500
<b>Total:</b>				\$ 675,000	\$ 0	\$ 688,500	\$ 0	\$ 688,500	\$ 0	\$ 688,500	\$ 688,500
<b>Total Conservación Documental:</b>				\$ 13,500							
<b>Total General:</b>				702000							

**REPORTE 22: INFORME GENERAL DE MAYORES VALORES REGISTRADOS Y PAGADOS DEL**

DIA



### Superintendencia de Notariado y Registro

Ministerio del Interior y de Justicia

República de Colombia

www.supernotariado.gov.co

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999  
INFORME GENERAL DE MAYORES VALORES REGISTRADOS Y PAGADOS EL DÍA 18-06-2018  
Impreso el 19 de Junio de 2018 a las 13:59:36 PM

Página 1

Concepto	REGISTRO					
Radicación	Hora	Liquidador	Cajero	Recibo	Valor	Vr.Conservación Documental
2018-999-6-425	19:16:48	51856			\$ 51,000 Ingresado	\$ 1,000
2018-999-6-426	19:17:24	51856			\$ 86,700 Ingresado	\$ 1,700
2018-999-6-427	19:18:02	51856			\$ 65,300 Ingresado	\$ 1,300
2018-999-6-428	19:18:38	51856			\$ 85,700 Ingresado	\$ 1,700
2018-999-6-426	19:27:46	51856	51856	62483332	\$ 86,700 Cancelado	\$ 1,700
2018-999-6-428	19:28:00	51856	51856	62483333	\$ 85,700 Cancelado	\$ 1,700
2018-999-6-435	19:38:19	51856			\$ 516,100 Ingresado	-\$ 34,100
2018-999-6-435	19:39:50	51856	51856	62483334	\$ 516,100 Cancelado	-\$ 34,100
2018-999-6-419	19:40:30	51856			-\$ 4,199,600 Ingresado	\$ 0
2018-999-6-420	19:40:44	51856			-\$ 1,562,600 Ingresado	\$ 0
2018-999-6-422	19:41:00	51856			-\$ 6,708,200 Ingresado	\$ 0
2018-999-6-424	19:41:14	51856			-\$ 75,558,200 Ingresado	\$ 0
2018-999-6-435	19:53:12	51856			-\$ 1,705,400 Ingresado	-\$ 34,100
2018-999-6-436	20:22:26	51856			-\$ 134,600 Ingresado	\$ 0
Total Mayores Valores REGISTRO:					-\$ 88,375,300	-\$ 93,200
						-\$ 88,375,300
						-\$ 88,375,300
						-\$ 93,200
						-\$ 88,468,500

REPORTE 24 : INFORME MAYORES VALORES INGRESADOS



**Superintendencia de Notariado y Registro**  
 Ministerio del Interior y de Justicia  
 República de Colombia  
[www.supernotariado.gov.co](http://www.supernotariado.gov.co)

**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999**  
**INFORME MAYORES VALORES INGRESADOS DEL 18-06-2018 AL 18-06-2018**

Impreso el 19 de Junio de 2018 a las 14:00:43 PM

Página 1

**REALIZADO POR: USUARIO.PRUEBAS**

**Concepto: REGISTRO**

Radicación	Recibo	Valor	Derechos Impuestos	Solicitante	T.Conservacion Documental	Fecha Ing.	Usuar.	Cajero	Fecha
2018-999-6-426	62483332	\$ 86,700	\$ 85,000	\$ 0 ANDRES	\$ 1,700	18-06-2018 19:27:46	USUARIO. PRUEBAS	USUARIO.PRUEBAS	18-06-2018
2018-999-6-428	62483333	\$ 85,700	\$ 84,000	\$ 0 ANDRES	\$ 1,700	18-06-2018 19:28:00	USUARIO. PRUEBAS	USUARIO.PRUEBAS	18-06-2018
2018-999-6-435	62483334	\$ 516,100	\$ 506,000	\$ 0 HG SOLANO	\$ 10,100	18-06-2018 19:39:50	USUARIO. PRUEBAS	USUARIO.PRUEBAS	18-06-2018
2018-999-6-425		\$ 51,000		ANDRES	\$ 1,000		USUARIO. PRUEBAS	USUARIO.PRUEBAS	18-06-2018
2018-999-6-427		\$ 65,300		ANDRES	\$ 1,300		USUARIO. PRUEBAS	USUARIO.PRUEBAS	18-06-2018
<b>Totales REGISTRO:</b>		<b>\$ 804,800</b>	<b>\$ 675,000</b>	<b>\$ 0</b>		<b>\$ 15,800</b>			
<b>Totales Reporte:</b>		<b>\$ 804,800</b>	<b>\$ 675,000</b>	<b>\$ 0</b>		<b>\$ 15,800</b>			

**Resumen General**

Concepto	Liq. Con Pago	Liq. Sin Pago	Derechos Con Pago	Derechos Sin Pago	Imp. Con Pago	Imp. Sin Pago
<b>Totales:</b>						

**REPORTE 25: INFORME RECAUDOS DEL DIA POR USUARIO**



**Superintendencia de Notariado y Registro**  
 Ministerio del Interior y de Justicia  
 República de Colombia  
[www.supernotariado.gov.co](http://www.supernotariado.gov.co)

**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999**  
**INFORME RECAUDOS DEL DIA POR USUARIO 18-06-2018**

**REALIZADO POR: USUARIO.PRUEBAS**

Impreso el 19 de Junio de 2018 a las 14:01:46 PM

Página 1

**USUARIO: USUARIO.PRUEBAS**

Radicación	Recibo	Forma de Pago	No Doc	Banco	No. Aprobación	Vr.Docu.Pago	Vr. Aplicado	Diferencia		Hora Pago	Proc.
								Vr. Docu. Pago	Vr. Aplicado		
2018-999-6-418	1236592783	EFFECTIVO				\$ 3,877,700	\$ 3,845,100		\$ 32,600	6.54.50 PM	REG
2018-999-1-448	1236592784	EFFECTIVO				\$ 0	\$ 16,300	-\$ 16,300	-\$ 16,300	6.54.50 PM	CER-ASC
2018-999-1-449	1236592785	EFFECTIVO				\$ 0	\$ 16,300	-\$ 16,300	-\$ 16,300	6.54.50 PM	CER-ASC
2018-999-6-419	1236592786	EFFECTIVO				\$ 4,199,600	\$ 4,167,000		\$ 32,600	7.02.46 PM	REG
2018-999-1-450	1236592787	EFFECTIVO				\$ 0	\$ 16,300	-\$ 16,300	-\$ 16,300	7.02.46 PM	CER-ASC
2018-999-1-451	1236592788	EFFECTIVO				\$ 0	\$ 16,300	-\$ 16,300	-\$ 16,300	7.02.46 PM	CER-ASC
2018-999-6-420	1236592789	EFFECTIVO				\$ 1,562,600	\$ 1,530,000		\$ 32,600	7.04.48 PM	REG
2018-999-1-452	1236592770	EFFECTIVO				\$ 0	\$ 16,300	-\$ 16,300	-\$ 16,300	7.04.48 PM	CER-ASC
2018-999-1-453	1236592771	EFFECTIVO				\$ 0	\$ 16,300	-\$ 16,300	-\$ 16,300	7.04.48 PM	CER-ASC
2018-999-6-421	1236592772	EFFECTIVO				\$ 5,558,600	\$ 5,542,300		\$ 16,300	7.04.48 PM	REG
2018-999-1-454	1236592773	EFFECTIVO				\$ 0	\$ 16,300	-\$ 16,300	-\$ 16,300	7.04.48 PM	CER-ASC
2018-999-6-422	1236592774	EFFECTIVO				\$ 6,708,200	\$ 6,675,600		\$ 32,600	7.05.58 PM	REG



**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999**  
**INFORME RECAUDOS DEL DIA POR USUARIO 18-06-2018**

REALIZADO POR: USUARIO.PRUEBAS

Impreso el 19 de Junio de 2018 a las 14:01:46 PM

Página 2

USUARIO: USUARIO.PRUEBAS

Radicación	Recibo	Forma de Pago	No Doc	Banco	No. Aprobación	Vr.Docu.Pago	Vr. Aplicado	Diferencia		Hora Pago	Proc.	
								Vr. Docu. Pago	Vr. Aplicado			
2018-099-1462	1236592786	Efectivo				\$ 0	\$ 16,300		-\$ 16,300	7.09.29 PM	CER-ASC	
2018-099-1463	1236592787	Efectivo				\$ 0	\$ 16,300		-\$ 16,300	7.09.29 PM	CER-ASC	
2018-099-5-427	1236592788	Efectivo				\$ 614,500	\$ 581,900		\$ 32,600	7.10.44 PM	REG	
2018-099-1464	1236592789	Efectivo				\$ 0	\$ 16,300		-\$ 16,300	7.10.44 PM	CER-ASC	
2018-099-1465	1236592790	Efectivo				\$ 0	\$ 16,300		-\$ 16,300	7.10.44 PM	CER-ASC	
2018-099-5-428	1236592791	Efectivo				\$ 7,503,800	\$ 7,561,200		\$ 32,600	7.11.57 PM	REG	
2018-099-1466	1236592792	Efectivo				\$ 0	\$ 16,300		-\$ 16,300	7.11.57 PM	CER-ASC	
2018-099-1467	1236592793	Efectivo				\$ 0	\$ 16,300		-\$ 16,300	7.11.57 PM	CER-ASC	
2018-099-5-429	1236592794	Efectivo				\$ 5,737,500	\$ 5,737,500		\$ 0	7.13.48 PM	REG	
2018-099-1468	1236592795	Efectivo				\$ 0	\$ 0		\$ 0	7.13.48 PM	CER-MSV	
2018-099-5-430	1236592796	Efectivo				\$ 2,677,500	\$ 2,677,500		\$ 0	7.15.12 PM	REG	
2018-099-1469	1236592797	Efectivo				\$ 0	\$ 0		\$ 0	7.15.12 PM	CER-MSV	
2018-099-5-433	1236592802	Efectivo				\$ 2,311,300	\$ 2,295,000		\$ 16,300	7.27.16 PM	REG	
2018-099-1472	1236592803	Efectivo				\$ 0	\$ 16,300		-\$ 16,300	7.27.16 PM	CER-ASC	
2018-099-5-434	1236592804	CHEQUE	478747	36		\$ 2,000,000	\$ 2,000,000		\$ 0	7.29.22 PM	REG	
2018-099-5-434	1236592804	Efectivo				\$ 311,300	\$ 295,000		\$ 16,300	7.29.22 PM	REG	
2018-099-1473	1236592805	Efectivo				\$ 0	\$ 16,300		-\$ 16,300	7.29.22 PM	CER-ASC	
2018-099-5-435	1236592806	Efectivo				\$ 2,311,300	\$ 2,295,000		\$ 16,300	7.31.07 PM	REG	
2018-099-1474	1236592807	Efectivo				\$ 0	\$ 16,300		-\$ 16,300	7.31.07 PM	CER-ASC	
2018-099-4-75	1236592808	Efectivo					-\$ 4,199,600		\$ 0	-\$ 4,199,600	7.46.15 PM	DEV

**DETALLE DEL RECAUDO**

Total Efectivo:	\$ 136,383,200
Total Consignación:	\$ 0
Total Cheque:	\$ 2,000,000
Total Pago Electronico:	\$ 0
Total Tarjeta Credito:	\$ 0
Total Tarjeta Debito:	\$ 0
Total Convenio:	\$ 0
Total recaudo ORIP:	\$ 138,383,200
<b>Vr. total aplicado:</b>	<b>\$ 138,383,200</b>

**Total concepto de Conservacion \$ 975,200**

Resumen General por Bancos

BANCO	VALOR APPLICADO	VALOR
36 MEGABANCO		
CHEQUE	\$ 2,000,000	\$ 2,000,000

Resumen General por Bancos - Devoluciones

**Turnos Anulados**

Radicación	Recibo	Forma de Pago	No Doc	Banco	No. Aprobación	Vr.Docu.Pago	Vr. Aplicado	F. Radica.	F. Anula.	Hora Pago	Proc.
2018-999-6-431	1236592798	EFFECTIVO				\$ 2,311,300	\$ 2,295,000	18/06/2018	18/06/2018	19.16.37	REG
2018-999-1-470	1236592799	EFFECTIVO				\$ 0	\$ 16,300	18/06/2018	18/06/2018	19.16.37	CER
2018-999-6-432	1236592800	EFFECTIVO				\$ 6,518,800	\$ 6,502,500	18/06/2018	18/06/2018	19.17.55	REG
2018-999-1-471	1236592801	EFFECTIVO				\$ 0	\$ 16,300	18/06/2018	18/06/2018	19.17.55	CER
<b>Totales Turnos Anulados:</b>						<b>\$ 8,830,100</b>	<b>\$ 8,830,100</b>				

**REPORTE 99: INGRESOS DIARIOS POR DOCUMENTOS**



**Superintendencia de Notariado y Registro**

Ministerio del Interior y de Justicia

República de Colombia

[www.supernotariado.gov.co](http://www.supernotariado.gov.co)

**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999**

**INGRESOS DIARIOS POR DOCUMENTOS DEL: 18-06-2018 AL: 18-06-2018**

**REALIZADO POR: USUARIO.PRUEBAS**

Impreso el 19 de Junio de 2018 a las 14:02:52 PM

Página 1

**18 de Junio de 2018**

Turno	Recibo Solicitante	Acto	Tarifa	Cuantia/Can	Recaudo	T. Conservacion Documental
2018-999-6-418	1236592763 ANDRES	ACTUALIZACION DE NOMENCLATURA	N	1	\$ 19,700	\$ 0
2018-999-6-418	1236592763 ANDRES	VENTA	N	500,000,000	\$ 3,750,000	\$ 0
2018-999-6-419	1236592766 ANDRES	ARRENDAMIENTO	N	1	\$ 35,300	\$ 700
2018-999-6-419	1236592766 ANDRES	VENTA	N	540,000,000	\$ 4,050,000	\$ 81,000
2018-999-6-420	1236592769 ANDRES	VENTA	N	200,000,000	\$ 1,500,000	\$ 30,000
2018-999-6-421	1236592772 HG SOLANO	HIPOTECA	N	30,000,000	\$ 183,600	\$ 3,700
2018-999-6-421	1236592772 HG SOLANO	VENTA	N	700,000,000	\$ 5,250,000	\$ 105,000
2018-999-6-422	1236592774 ANDRES	CANCELACION AFECTACION FAMILIAR	N	1	\$ 19,700	\$ 400
2018-999-6-422	1236592774 ANDRES	VENTA	N	870,000,000	\$ 6,525,000	\$ 130,500
2018-999-6-423	1236592777 HG SOLANO	VENTA	N	400,000,000	\$ 3,000,000	\$ 60,000
2018-999-6-424	1236592779 ANDRES	DONACION INTERNACIONAL	N	1	\$ 19,700	\$ 400
2018-999-6-424	1236592779 ANDRES	VENTA	N	9,870,000,000	\$ 74,025,000	\$ 1,480,500
2018-999-6-425	1236592782 ANDRES	CONSTITUCION SOCIEDAD COMERCIAL	N	1	\$ 35,300	\$ 700

2018-999-6-428	1236592785 ANDRES	INSCRIPCION DE VENTA	N N	1 2,000,000	\$ 19,700 \$ 35,300	\$ 400 \$ 700
2018-999-6-428	1236592785 ANDRES	ADMINISTRACION ANTICRETICA	N	1	\$ 19,700	\$ 400
2018-999-6-427	1236592788 ANDRES	VENTA	N	90,000,000	\$ 550,800	\$ 11,000
2018-999-6-427	1236592788 ANDRES	INSCRIPCION FOLIO ADICIONAL	N	1	\$ 10,400	\$ 200
2018-999-6-428	1236592791 ANDRES	VENTA	N	987,000,000	\$ 7,402,500	\$ 148,100
2018-999-6-428	1236592791 ANDRES	HIPOTECA	N	450,000,000	\$ 3,375,000	\$ 67,500
2018-999-6-429	1236592794 HG SOLANO	VENTA	N	300,000,000	\$ 2,250,000	\$ 45,000
2018-999-6-429	1236592794 HG SOLANO	HIPOTECA	N	350,000,000	\$ 2,625,000	\$ 52,500
2018-999-6-430	1236592796 HG SOLANO	VENTA	N	0	\$ 0	\$ 0
2018-999-6-430	1236592802 HG SOLANO	HIPOTECA	N	300,000,000	\$ 2,250,000	\$ 45,000
2018-999-6-433	1236592804 HG SOLANO	VENTA	N	300,000,000	\$ 2,250,000	\$ 45,000
2018-999-6-434	1236592806 HG SOLANO	VENTA	N	300,000,000	\$ 2,250,000	\$ 45,000
2018-999-6-435	1236592811 ANDRES	VENTA	N	20,000,000	\$ 122,400	\$ 2,400
2018-999-6-436	1236592814 HG SOLANO	VENTA	N	400,000,000	\$ 3,000,000	\$ 60,000
437						

Total recaudo ajustado a la centena 18 de Junio de 2018: \$ 127,274,100 \$ 2,470,100

## TOTALES

**Valor Conservacion Documental:** \$ 2,470,100



**Superintendencia de Notariado y Registro**  
Ministerio del Interior y de Justicia  
República de Colombia  
[www.supernotariado.gov.co](http://www.supernotariado.gov.co)

**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999**  
**INGRESOS DIARIOS POR DOCUMENTOS DEL: 18-06-2018 AL: 18-06-2018**  
**REALIZADO POR: USUARIO.PRUEBAS**

Impreso el 19 de Junio de 2018 a las 14:02:52 PM

Página 2

TIPO PAGO	TOTAL
<b>Total Efectivo:</b>	127,819,600
<b>Total Consignación:</b>	0
<b>Total Cheque:</b>	2,000,000
<b>Total Convenio:</b>	0
<b>Total Pago Electrónico PSE:</b>	0
<b>Total Tarjeta Crédito:</b>	0
<b>Total Tarjeta Débito:</b>	0
<b>Total Otros Pagos:</b>	0
<b>Total Oficina Vr:</b>	129,819,600
<b>Total Otras ORIPS Vr:</b>	0
<b>Total:</b>	129,819,600



**DESPLIEGUE 7.3.0**  
**12/05/2017**



**2230 AJUSTE DE TARIFAS REGISTRARLES 2018**

El Sistema de Información Registral SIR debe ajustarse teniendo en cuenta las nuevas tarifas registrales estipuladas en la **resolución No. 2854 del 16 de marzo de 2018** para todos los procesos que impliquen un cobro en el sistema.

**ACTOS CON CUANTIA:**

ACTOS CON CUANTIA		
SMMLV INICIO	SMMLV FINAL	Tarifa
≥10	≤150	6.12 x 1.000
> 150	En adelante	7.50 x 1.000
≥ MAYOR O IGUAL QUE	≤ MENOR O IGUAL QUE	
> MAYOR QUE	< MENOR QUE	

**ACTOS SIN CUANTÍA** la suma de diecinueve mil setecientos pesos (\$ 19.700) por cada uno de los actos que por su naturaleza carezcan de cuantía en el documento de inscripción. Salvo los casos en esta resolución, también deberá cancelarse la suma de diez mil cuatrocientos pesos (\$ 10, 400), por cada folio de matrícula adicional donde deba inscribirse el documento.

**ACTOS MINIMA CUANTIA:** Los actos o negocios jurídicos con cuantía inferior a Diez salarios mínimos mensuales legales vigentes (< 10 SMMLV) tendrán una tarifa de treinta y cinco mil tresientos pesos (\$ 35.300).

Estas tarifas deben ser ajustadas en la aplicación SIR, las cuales se deben realizar por base de datos y además se deben ajustar los recibos con los cambios correspondientes a las nuevas tarifas y especificando el cobro del 2% por concepto de conservación documental.

Se debe modificar algunos reportes generados por la aplicación detallando la información del cobro del 2% por concepto de conservación documental, para el 2%, si el valor a cobrar por concepto de conservación documental termina en decimal el sistema redondeará dicho valor, como ya lo viene realizando la aplicación para cada uno de los actos.

Los reportes existentes en el sistema a modificar son el 05, 09, 15, 16, 18, 19, 22, 24, 25, 51, 99, 151 y 156 los cuales se detallan en el manual de SIR

## **Impresión Recibo de Registro**

Se solicita que todos los recibos que se generen por la aplicación SIR y que generen algun tipo de pago se modifiquen ajustandose a las nuevas tarifas, y ademas especificando el cobro del 2% el cual corresponde al cobro por concepto de conservación documental, esto aplica para los procesos de Registro de documentos, Devolucion de dinero y Pagos de mayor valor, para los demas procesos como Fotocopias, Consultas y Correcciones este cambio no se aplicara, los ajustes a las tarifas actuales se realiza por base de datos.

Para la actualizacion de las tarifas registrales los procesos afectados son:

- ❖ Certificados
- ❖ Fotocopias
- ❖ Registro
- ❖ Devolucion dinero
- ❖ Correciones ( solo aplica cuando se debe cobrar un mayor valor)

Tambien se aplicara para el subprocesso Mayor Valor.

Para el proceso de consultas este no se afectaria debido a que en las oficinas solo se realiza la consulta exenta.

A continuacion se presenta el modelo de los recibos a los que se le realizara el ajuste anteriormente mencionado:

- Recibo de liquidación En el proceso de liquidación adicionar en el recibo el total cobrado por concepto de conservación documental y este sumarlo al total de la liquidación .



OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
CIRCULO\_PRUEBAS  
NIT: 899999007-0

CONSTANCIA DE LIQUIDACIÓN

Impreso el 30 de Abril de 2018 a las 00:35:53 am  
No. DE LIQUIDACIÓN: L-2018-999-6-417

REGISTRADO POR: 51856  
MATRICULA(S) 999-2369  
SUBTIPO DE SOLICITUD: NORMAL  
VALOR DE DERECHOS DE REGISTRO: \$ 3.000.000  
VALOR DE IMPUESTO: \$ 0  
VALOR DE LA MULTA: \$ 0  
CERTIFICADOS ASOCIADOS: \$ 16.300

Valor de conservación documental del 2%: 60000.0

VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS \$ 3.060.000

NOTA: ESTA CONSTANCIA TIENE VIGENCIA POR UN DÍA.  
EL PROCESO DE REGISTRO EMPIEZA CON LA RADICACIÓN UNA VEZ EFECTUADO EL PAGO DE LOS DERECHOS

- RECIBO SOLICITUD REGISTRO DE DOCUMENTOS, En el proceso de registro es necesario agregar la línea que denote lo cobrado por concepto de conservación documental y que a su vez este sea sumado al total de la conservación. Se visualiza en imagen 2

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
CIRCULO\_PRUEBAS - NIT: 899999007-0  
SOLICITUD REGISTRO DOCUMENTOS  
Impreso el 3 de Mayo de 2018 a las 03:12:16 pm

**1236592520**

No. RADICACIÓN: **2018-999-6-302**

NOMBRE DEL SOLICITANTE: HG SOLANO  
OFICIO No.: 1 del 3/5/2018 BANCO CENTRAL HIPOTECARIO LIQUIDADO de BOGOTA D. C.

**MATRICULAS: 999-1336**

ACTOS A REGISTRAR:

Tipo....	Código....	Cuantía....	Tipo Tarifa	Derecho	Impuesto
VENTA	1	\$ 400.000.000	N	\$ 2.280.000	\$ 0
<b>SUBTOTAL</b>		\$ 2.280.000			
Conservacion documental del 2%		\$ 45.600			
<b>TOTAL</b>		\$ 2.280.000			

**TURNOS CERTIFICADOS ASOCIADOS: 1**

**VALOR MULTA: \$0**

**FORMA DE PAGO:**

Efectivo VALOR PAGADO: \$2.280.000 VALOR DOC : \$2.341.900

VALOR DERECHOS: \$2.280.000

**VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS: \$ 2,362,600**

Usuario: 51856

En el recibo de certificado asociado validar que el valor del documento sea el mismo que el del recibo del registro

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
NIT: 899999007-0 CIRCULO\_PRUEBAS  
SOLICITUD CERTIFICADO DE TRADICION  
Impreso el 3 de Mayo de 2018 a las 03:12:19 pm

1236592521

No. RADICACIÓN: 2018-999-1-333

Asociado al turno de registro: 2018-999-6-302

MATRICULA: 999-1336

NO SE EXPIDE DE INMEDIATO EL CERTIFICADO

NOMBRE DEL SOLICITANTE: HG SOLANO

CERTIFICADOS: 1

FORMA DE PAGO:

EFFECTIVO VALOR PAGADO.: \$16.300 VALOR DOC.: \$ 2,362,600

VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: \$ 16.300

EL CERTIFICADO SE EXPIDE DE ACUERDO A LOS DATOS SUMINISTRADOS

Usuario: 51856

- Recibo mayor valor: En el proceso de pago mayor valor adicionar en el recibo el total cobrado por concepto de conservación documental y este sumarlo al valor total a pagar

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
CIRCULO\_PRUEBAS - NIT: 899999007-0  
RECIBO DE PAGO DE MAYOR VALOR  
Impreso el 28 de Abril de 2018 a las 01:42:34 pm

62483273

No. RADICACIÓN: 2018-999-6-238

NOMBRE DEL SOLICITANTE: HG SOLANO

FORMA DE PAGO:

EFFECTIVO VALOR:... \$ 25,000

VALOR DERECHOS: \$ 25,000

VALOR CERTIFICADOS: \$0

CONSERVACION DOCUMENTAL 2%: \$ 400

VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS: \$ 25,400

JUSTIFICACION PAGO MAYOR VALOR:

VALOR MAYOR

Usuario: 51856

En el proceso de corrección pago mayor valor, adicionar en el recibo el total cobrado por concepto de conservación documental y este sumarlo al valor total a pagar.

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
CIRCULO\_PRUEBAS

NIT: 899999007-0

RECIBO DE PAGO DE MAYORES VALORES

Impreso el 2 de Mayo de 2018 a las 03:14:29 pm

**62483283**

No. RADICACIÓN: **2018-999-3-5**

NOMBRE DEL SOLICITANTE: BELTRAN MORENO ANDRES CC# 2454365  
TELÉFONO: 8764675

FORMA DE PAGO:

Efectivo Valor: \$ 100.000

Valor Derechos: \$100.000

**CONSERVACION DOCUMENTAL 2: \$ 2, 000**

**VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: \$ 102. 000**

- Recibo Nueva Entrada adicionar en el total cobrado por concepto de conservación documental y este sumarlo al valor total a pagar

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS

CIRCULO\_PRUEBAS - NIT: 899999007-0

SOLICITUD REGISTRO DOCUMENTOS

Impreso el 27 de Abril de 2018 a las 09:59:06 am

**1236592341**

No. RADICACIÓN: **2018-999-6-224**

NOMBRE DEL SOLICITANTE: HG SOLANO

SUCESION No.: 1 del 27/4/2018 P.L. DE BOSA/CAÑAS ARBOLEDA DAVID ED de BOSA

MATRICULAS: 999-1236

ACTOS A REGISTRAR:

Tipo....	Código....	Cantidad....	Tipo Tarifa	Derecho	Impuesto
VENTA	1	\$ 400.000.000	N	\$ 3.000.000	\$ 0
<b>SUBTOTAL</b>		\$ 3.000.000			
Conservacion documental del 2%		\$ 60.000			
<b>TOTAL</b>		\$ 3.060.000			

**TURNOS CERTIFICADOS ASOCIADOS: 1**

VALOR MULTA: \$0

FORMA DE PAGO:

Efectivo Valor Pagado: \$3.000.000 Valor Doc: \$3.076.300

Valor Derechos: \$3.000.000

Valor Total a Pagar Ajustado a la Centena por Actos: \$ **3.060.000**

Usuario: 51856

- En el recibo de certificado asociado validar que el valor del documento sea el mismo que el del recibo generado en la nueva entrada

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
 NIT: 899999007-0 CIRCULO\_PRUEBAS  
 SOLICITUD CERTIFICADO DE TRADICION  
 Impreso el 27 de Abril de 2018 a las 09:59:08 am

**1236592342**

No. RADICACIÓN: **2018-999-1-233**

Asociado al turno de registro: 2018-999-6-224

MATRICULA: **999-1236**

NO SE EXPIDE DE INMEDIATO EL CERTIFICADO

NOMBRE DEL SOLICITANTE: HG SOLANO

CERTIFICADOS: 1

FORMA DE PAGO:

Efectivo VALOR PAGADO.: \$16.300 VALOR DOC.: \$3.076.300

VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: **\$ 16.300**

EL CERTIFICADO SE EXPIDE DE ACUERDO A LOS DATOS SUMINISTRADOS

Usuario: 51856

- Modificar el pie de página del recibo de liquidación de documentos a ser fotocopiados, cambiando el número de resolución de 0450 del 20 de enero de 2017, a la nueva resolución con el número 2854 del 16 de marzo de 2018

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS  
 CIRCULO\_PRUEBAS  
 NIT: 899999007-0

#### Liquidacion de Documentos a ser Fotocopiados

Impreso el 17 de Abril de 2018 a las 00:47:56 am

No. RADICACIÓN: **2018-999-5-3**

---

##### Documentos Asociados a esta solicitud:

- 1.
- Tipo de Documento: DEMANDA  
 Número de copias: 1  
 Descripción: PRUEBA TARIFAS  
 Tipo de fotocopia: NORMAL  
 Num Hojas: 1

---

Total Juegos de Copias: 1  
 Valor Liquidado Ajustado a la Centena por Actos: \$ 600

---

Solicitante: ANDRES BELTRAN MORENO  
 Telefono: 545147421  
 Radicado Por: 51856

Según Resolución 2854 del 16 de marzo de 2018, se cobran las siguientes tarifas:

- Documentos almacenados en medio óptico o microfilmado: \$1.100,00 por cada página reproducida.
- Documentos que reposen en los archivos físicos de la respectiva oficina de registro: \$600,00 por cada página fotocopiada.

De acuerdo con la Resolución 5123 del 2.000. Artículo 1. Parágrafo y Resolución 4907 de 2009. Artículo 1. "Se entenderá que el interesado desiste de su petición cuando transcurridos dos meses contados a partir de la solicitud no acredite el pago respectivo de lo cual el Registrador dejaría expresa constancia en memorial peticionario de las

- **Modificar pantalla Administrativa ver detalles de Turno**

Se requiere modificar la pantalla administrativa ver detalles de turno, agregando un campo donde muestre al usuario el valor cobrado por concepto de conservación documental, cuando el turno posea cobro por mayor valor este también debe visualizar el cobro por concepto de conservación documental.

Actos Asociados				
Tipo Acto:	Cantidad:	Valor:	Valor Conservación Documental:	
ACLARACION	1	19,700	400	
CESION (SIN CUANTIA PARA IMPETO)	1	19,700	400	
LIQUIDACION SOCIEDAD COMERCIAL	1	35,300	700	
ENGLOBE	1	19,700	400	
VENTA	2	35,300	700	

Certificados Asociados				
TURNO	TIPO	VALOR		
2018-999-1-94	MEDIO	\$,200		
2018-999-1-95	MEDIO	\$,200		

Pagos asociados				
Valor derechos de registro:	129,700	Valor certificados asociados:	16,400	
Valor de impuestos:	0	Valor de multa:	0	
Otro impuesto:		Valor otro impuesto:	0	
Valor Conservación Documental	2,600			

Valor Pagado:  
EFECTIVO 129,700

Pago Mayor Valor				
Razón del pago:	PRUEBA 20-04-2018 TARDE	Valor certificados asociados:	0	
Valor derechos de registro:	19,700	Valor pagado:	19,700	
Otro impuesto:	0			
Valor Conservación Documental	1,100			

Forma de Pago: Banco/Franquicia: No. Documento Pago: Valor Documento: Fecha Documento: Valor Pagado:  
CHEQUE BANCO AGRARIO 809765234 35,300 20/04/2018 35,300

- **Modificar pantalla preliquidar**

Pantalla preliquidación esta opción se encuentra por el rol Liquidador/ botón preliquidar, en esta pantalla se requiere adicionar el concepto de conservación documental y debe ser sumado al total de la liquidación

Datos Básicos																							
Tipo	97	SUCESION	<input checked="" type="checkbox"/>	Número	57878	Fecha	2/5/2018																
Actos																							
#	Id	Tipo	Cuenta, cantidad o importe	Derechos registro	Impuesto																		
1		VENTA	400,000,000	2,280,000	NO APlica																		
Nuevo Acto																							
<table border="1"> <tr> <td>Acto</td> <td>Tipo</td> <td>Descripción</td> <td>Tarifa</td> </tr> <tr> <td>Tipo:</td> <td></td> <td>SELECCIONE UNA OPCION-</td> <td></td> </tr> <tr> <td>Cuenta, autovalor o cantidad</td> <td></td> <td>No hay datos PORCENTUAL</td> <td></td> </tr> <tr> <td>Impuesto?</td> <td><input type="radio"/> NORMAL</td> <td></td> <td></td> </tr> </table>								Acto	Tipo	Descripción	Tarifa	Tipo:		SELECCIONE UNA OPCION-		Cuenta, autovalor o cantidad		No hay datos PORCENTUAL		Impuesto?	<input type="radio"/> NORMAL		
Acto	Tipo	Descripción	Tarifa																				
Tipo:		SELECCIONE UNA OPCION-																					
Cuenta, autovalor o cantidad		No hay datos PORCENTUAL																					
Impuesto?	<input type="radio"/> NORMAL																						
Otro Impuesto																							
<table border="1"> <tr> <td>Valor</td> <td>Descripción</td> </tr> <tr> <td></td> <td></td> </tr> </table>								Valor	Descripción														
Valor	Descripción																						
Certificados asociados																							
# de certificados	Tipo tarifa																						
0	EXENTO																						
0	MEDIO																						
1	NORMAL																						
<table border="1"> <tr> <td>Liquidar</td> <td>Regresar</td> </tr> <tr> <td>\$ 2,280,000,00</td> <td></td> </tr> <tr> <td>\$ .00</td> <td></td> </tr> <tr> <td>\$ .00</td> <td></td> </tr> <tr> <td>\$ .00</td> <td></td> </tr> <tr> <td>\$ 45,600</td> <td></td> </tr> <tr> <td colspan="2">Valor total</td> </tr> <tr> <td colspan="2">\$ 2,325,600</td> </tr> </table>								Liquidar	Regresar	\$ 2,280,000,00		\$ .00		\$ .00		\$ .00		\$ 45,600		Valor total		\$ 2,325,600	
Liquidar	Regresar																						
\$ 2,280,000,00																							
\$ .00																							
\$ .00																							
\$ .00																							
\$ 45,600																							
Valor total																							
\$ 2,325,600																							

- Modificar pantalla de pago cuando se trae con liquidación**

Modificar la pantalla de pago cuando esta tiene ya una liquidación para denotar lo cobrado por concepto de conservación documental y a su vez sumar este concepto al total a pagar.

Concepto	Valor
Valor derechos	\$ 2,280,000.00
Valor conservación documental	\$ 45,600.00
Valor impuestos	\$ 0.00
Valor multa	\$ 0.00
Valor certificados asociados	\$ 16,300.00
Valor otros impuestos	\$ 0.00
Valor total ajustado a la centena por actos	\$ 2,341,900.00

- Modificar pantalla de pago cuando el turno se realiza desde la fase de registro**

Modificar la pantalla cuando se realiza un registro sin una liquidación previa, agregando un nuevo campo que denote lo cobrado por concepto de conservación documental y que as u ves este valor sea sumado al total del registro.

Concepto	Valor
Valor derechos	\$ 2,280,000.00
Valor conservación documental	\$ 45,600.00
Valor impuestos	\$ 0.00
Valor multa	\$ 0.00
Valor certificados asociados	\$ 16,300.00
Valor otros impuestos	\$ 0.00
Valor total ajustado a la centena por actos	\$ 2,341,900.00

- Modificar pantalla de calificación de un turno**

Modificar la pantalla de calificación de un turno, agregando un campo que denote el valor total cobrado por concepto de conservación documental

Información Motivo Envío

Revisión liquidación

#	Id	Tipo	Cuenta, cantidad o autoavaluo	Derechos registro	Impuesto
1	18	AFFECTACION A VIVIENDA FAMILIAR	1	19,700	NO APlica
2	44	RESTITUCION FIDUCIA (SIN CUANTIA PARA IMPTO)	1	35,300	NO APlica

Otros impuestos  
No hay otros impuestos asociados

Valor Conservación Documental: 1,100

Total liquidaciones : 55,000

- **Modificar pantalla custodia mayor valor**

Modificar la pantalla de custodia mayor valor, agregando un campo que denote el valor total cobrado por concepto de conservación documental y sumando este al valor total de la liquidación

http://192.168.76.57:7778/SNR/turno.do

SNR :: Superintendencia de ...

Custodia pagos mayores

Matrícula Inmobiliaria de la Propiedad

Turno Anterior

¿ Está asociado a un turno anterior ?  
Número del Turno Anterior

Datos Solicitante

Tipo de Identificación CEDULA Número 80844585  
Primer Apellido HG SOLANO Segundo Apellido

Detalle de la liquidación

Valor por Derechos	\$ 25,000.00
Valor por Certificados	\$ .00
Valor impuestos	\$ .00
Valor Conservación Documental	\$ 400.00
Razón del pago	134 PRUEBAS

Valor de la liquidación  
25,400.00

- **Pantalla de entrega mayor valor**

Modificar el campo “valor de la liquidación” en la pantalla de entrega mayor valor, para que en este valor se totalice lo cobrado por concepto de conservación documental

- **Pantalla de cajero mayor valor**

Modificar el campo “valor de la liquidación” en la pantalla de entrega mayor valor, para que en este valor se totalice lo cobrado por concepto de conservación documental

- **Pantalla pago mayor valor**

Modificar en la sesión pago liquidación el campo en la pantalla de pago mayor valor, para que en este valor se totalice lo cobrado por concepto de conservación documental.

- Pantalla custodia mayores valores caso corrección**

Modificar la pantalla para el caso de corrección de mayor valor cuando se encuentre en el rol registrador en la fase custodia mayores valores.

- Pantalla entrega mayor valor caso corrección**

Modificar el campo “valor de la liquidación” en la pantalla de entrega mayor valor, para que en este valor se totalice lo cobrado por concepto de conservación documental, cuando el caso sea de corrección de mayor valor.

Pago Mayor Valor

Notificar Ciudadano - Pago Mayor Valor

Matrícula Inmobiliaria de la Propiedad

Matrículas

Matrícula: 999-234

Turno Anterior

¿ Está asociado a un turno anterior ?  
Número del Turno Anterior

Datos Solicitante

Tipo de Identificación  
Primer Apellido: ANDRES  
Nombre: SECUENCIA  
Número: 155501130  
Segundo Apellido:

Valor de la liquidación  
Nota : Total mas la conservacion documetal  
\$ 25,400.00

Aceptar

- Pantalla pago mayor valor caso corrección**

Modificar en la sesión pago liquidación el campo en la pantalla de pago mayor valor, para que en este valor se totalice lo cobrado por concepto de conservación documental, cuando el caso sea de corrección de mayor valor.

Registrar Pago

Pago por mayor valor

Haga click para maximizar

PAGO

Liquidación  
Valor: \$ 25,400.00  
Nota : Total mas la conservacion documetal

Forma de Pago  
-SELECCIONE UNA OPCION-

Registrar Pago Cancelar Agregar Pago

- Pantalla pago nueva entrada**

Modificar la pantalla de pago cuando al turno se le realiza una nueva entrada.

Liquidación: 2018-999-6-460

Registrar Pago

PAGO 0.698 Seg.

Liquidación

Valor derechos	\$ 2,280,000.00
Valor impuestos	\$ .00
Valor multa	\$ .00
Valor certificados asociados	\$ 16,300.00
Valor otros impuestos	\$ .00
<b>Valor de conservación documental</b>	<b>\$ 45,600.00</b>
Valor total ajustado a la centena por actos	\$ 2,341,900.00

Forma de Pago

SELECCIONE UNA OPCIÓN

Registrar Pago Siguientes Editar Cancelar Agregar Pago

- **Modificar pantalla proceso de devolución**

Modificar la pantalla rol receptor de devoluciones agregando el campo de lo cobrado por concepto de conservación documental.

Solicitud de Devolución de Dineros 0.427 Seg.

Turno asociado

Turno sobre el cual se solicita la devolución 2018-999-6-86 Agregar Turno

Turnos de Certificados Asociados 2018-999-1-92

Datos Asociados al Pago

Forma de Pago	Efectivo	Valor	81,700.00
Banco		Número	
Saldo a favor	.00	Valor de conservación documental	<b>1,300</b>

Forma de Pago	Efectivo	Valor	10,000.00
Banco		Número	
Saldo a favor	.00	Valor de conservación documental	<b>200</b>

Proceso de devoluciones – Rol usuario administrativo – Fase Análisis

Valores totales

Valor a devolver por derechos	\$ 129,700.00
Valor a devolver por derechos Mayor Valor	\$ 55,000.00
Saldo a Favor	\$ .00
Valor a devolver por certificados asociados	\$ 16,400.00
Valor a devolver por certificados Mayor Valor	\$ .00
<b>Valor a devolver por Conservación Documental</b>	<b>\$ .00</b>
Valor a devolver por impuestos	\$ .00
Valor a devolver por impuestos Mayor Valor	\$ .00

Proceso

devoluciones- rol usuario administrativo-fase resolución

Mayor Valor

- Saldo a Favor \$ .00
- Valor a devolver por certificados asociados \$ 16,300.00
- Valor a devolver por certificados asociados Mayor Valor \$ .00
- Valor a devolver por impuestos \$ .00
- Valor a devolver por impuestos Mayor Valor \$ .00
- Valor a devolver por Conservación Documental \$ .00

Aceptar Devolver a Análisis Cerrar

NOTAS

proceso devoluciones- rol usuario administrativo-fase recursos devolución

Valores a devolver

- Valor a devolver por derechos \$ 19,700.00
- Valor a devolver por derechos Mayor Valor \$ .00
- Valor a devolver por Conservación Documental \$ .00
- Saldo a Favor \$ .00
- Valor a devolver por certificados asociados \$ .00
- Valor a devolver por certificados asociados Mayor Valor \$ .00
- Valor a devolver por impuestos \$ .00
- Valor a devolver por impuestos Mayor Valor \$ .00

Aceptar Devolver a Resolución Cerrar

proceso devoluciones- rol pago devolución

Valores a devolver

- Valor a devolver por derechos \$ 19,700.00
- Valor a devolver por derechos Mayor Valor \$ .00
- Valor a devolver por Conservación Documental \$ .00
- Valor a devolver por certificados asociados \$ .00
- Valor a devolver por certificados asociados Mayor Valor \$ .00
- Valor a devolver por impuestos \$ .00
- Valor a devolver por impuestos Mayor Valor \$ .00

Resoluciones

Número	123	Fecha	19/04/2018	Fecha Ejecutoría	20/04/2018
--------	-----	-------	------------	------------------	------------

Historial de Recursos del Turno

- creación nuevo reporte consolidado conservación documental**

Se solicita crear un nuevo reporte que contenga el consolidado de conservación documental, el cual debe ser ubicado en la lista de reportes con el nombre de: “Reporte 180 Consolidado Conservación Documental”.

- Modificar la pantalla lista de reportes agregando la descripción del nuevo reporte

Reporte E04: Estadísticas de correcciones en Trámite  
 Reporte 176: Traslado masivo de folios a este círculo  
**Reporte 178: Informe de Turnos Corrección Encabezado**  
 Reporte 180: Consolidado Conservación Documental



- Crear la pantalla de consulta desde donde se escogerá el rango de fecha y el círculo para generar el reporte

**Reporte 180**

### Consolidado de conservación documental

Fecha Inicio: 
  
 Fecha Fin: 
  
 Círculo Registral :

**Regresar**

**Observar**

- Crear el Pdf donde se detallará lo cobrado por concepto de conservación documental.

Con la estructura planteada en la siguiente imagen.

**Superintendencia de Notariado y Registro**  
 Ministerio del Interior y de Justicia  
 República de Colombia  
[www.supernotariado.gov.co](http://www.supernotariado.gov.co)

**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS**  
**CONSOLIDADO DE CONSERVACIÓN DOCUMENTAL**

BUCARAMANGA - 300  
 01-03-2017 HASTA 17-03-2017

Impreso el 10 de Abril de 2018 a las 10:35:41 AM

Página 1

Concepto:	Registro		
Radicación	Fecha de Pago	Vr. Derechos	Total Conservación Documental
2017-300-6-3975	02-03-2017 10:17:47	\$ 208,000	\$ 4,200
2017-300-6-1310	02-03-2017 13:00:14	\$ 30,000	\$ 600
2017-300-6-6051	06-03-2017 15:12:55	\$ 34,000	\$ 0
2017-300-6-8139	07-03-2017 06:41:59	\$ 33,200	\$ 0
2017-300-6-3745	07-03-2017 11:38:05	\$ 19,000	\$ 0
2017-300-6-5385	07-03-2017 14:31:31	\$ 19,000	\$ 0
2017-300-6-7850	07-03-2017 14:47:33	\$ 19,000	\$ 0
2017-300-6-788	08-03-2017 15:21:48	\$ 10,600	\$ 0
2017-300-6-6226	08-03-2017 15:22:24	\$ 10,000	\$ 200
2017-300-6-7962	15-03-2017 15:10:27	\$ 447,500	\$ 0
2017-300-6-8107	16-03-2017 15:27:32	\$ 2,000	\$ 0
2017-300-6-6796	16-03-2017 15:27:53	\$ 19,000	\$ 0
2017-300-6-7264	16-03-2017 15:28:07	\$ 11,200	\$ 0
2017-300-6-7823	16-03-2017 15:28:22	\$ 19,000	\$ 0
2017-300-6-9528	16-03-2017 15:33:47	\$ 184,400	\$ 0
2017-300-6-9783	17-03-2017 10:11:55	\$ 19,000	\$ 0
2017-300-6-9408	17-03-2017 11:11:36	\$ 19,000	\$ 0
<b>TOTAL Registro:</b>		<b>\$ 1,103,900</b>	<b>\$ 5,000</b>



## DESPLIEGUE 7.2.0 09/11/2017



### 2154. JUSTIFICAR ACTIVACION-INACTIVACION DE ROLES Y USUARIOS

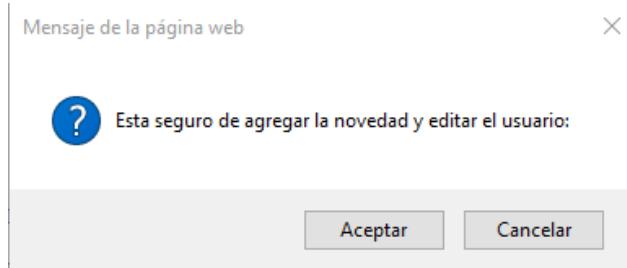
Para los roles de Administrador Regional y Administrador Nacional por el menú de pantallas administrativas a través de la opción de Administración de Usuarios y Administración de Perfiles, el sistema deberá permitir realizar los cambios que a continuación se mencionan por cada una de las opciones:

- **Administración de Usuarios**

En la pantalla administrativa “Administración de Usuarios” cada vez que se requiera hacer una novedad a un usuario, cuando se dé un clic en alguno de los botones Activar, Inactivar, o Editar se desplegará un formulario como lo ilustra la siguiente pantalla:

En la parte inferior de la pantalla se mostrará los siguientes botones con la siguiente funcionalidad Botón “Agregar”: Agregará la nota informativa seleccionada con su respectiva descripción. Este estará inhabilitado hasta que se cumplan las condiciones de validación del botón las cuales se mencionan más adelante. Una vez se dé clic en este botón para las opciones de Activar e Inactivar se presentará en pantalla un mensaje de alerta así:

- Está seguro de agregar la novedad y editar el usuario:



Botón “Regresar”: Retorna a la interfaz “Pantalla Administrativa Consulta de Usuarios”

Botón “Cancelar”: Retorna a la interfaz “Administración”.

Adicionalmente en esta opción se desplegará en la parte inferior de la pantalla el listado de las notas que ya haya tenido el funcionario que se está editando las cuales se presentarán en orden descendente de las más reciente a la más antigua, con la siguiente estructura:

Resultados de la Consulta					
Listado					
Rango Fecha, desde:		Hasta:	Buscar		
Descargar En Excel					
Nombre de Novedad	Fecha	Usuario	Tipo Nota	Descripción	
ACTIVAR-INACTIVAR ROL	04/10/2017	USUARIO.PRUEBAS	Nombramiento	SE ACIVA ROL POR NOMBRAMIENTO	- [ ]
ACTIVAR-INACTIVAR ROL	04/10/2017	USUARIO.PRUEBAS	Incapacidad	SE INCAPACIA POR AL RAZON	- [ ]

El sistema presentará en pantalla las últimas 20 notas, en caso de tener más el sistema generará un hipervínculo para navegar en las siguientes páginas, adicionalmente permitirá tener la consulta por un rango de fecha.

Adicionalmente estas consultas de notas podrán exportarse a Excel en caso de que se requiera.

Esta opción también tendrá dos botones al final de cada nota informativa “Imprimir” y “Ver adjunto”.

-Al dar clic en el botón “Imprimir” el sistema generará la impresión de la nota.

-Al dar clic en el botón “Ver adjunto” el sistema descargará el adjunto que se cargó en la nota.

- **Administración de Perfiles**

En la pantalla administrativa “Administración de Perfiles” asignado al rol registrador, administración regional y al administrador nacional, se realizaron los siguientes ajustes:

Tras seleccionar el círculo y el usuario a quien se le hará la modificación actuamente se presenta esta pantalla donde se lista todos los roles asociados al usuario que se consultó:

Roles	Estaciones	Estado	Dar Acceso	Eliminar	Cambiar Estado
ABOGADO DIRECCION REGISTRO	X-JUAN.DELACOSA	ACTIVA		<input type="checkbox"/>	
ABOGADO DIRECCION REGISTRO	X_USUARIO.PRUEBAS	ACTIVA		<input type="checkbox"/>	
ACTUACIONES ADMINISTRATIVAS	X_USUARIO.PRUEBAS	ACTIVA		<input type="checkbox"/>	
ADMINISTRADOR IMPRESION	X_USUARIO.PRUEBAS	ACTIVA		<input type="checkbox"/>	
ADMINISTRADOR REGIONAL	X_USUARIO.PRUEBAS	ACTIVA		<input type="checkbox"/>	

Allí se presenta los siguientes campos a los que se le deberán hacer los siguientes cambios según el tipo de novedad:

Cambiar estado / Eliminar

Se presentarán controles de lista de chequeo (para poder seleccionar uno a varios roles) y agregando los botones de acción (Cambio de estado y eliminar) en la parte superior.

Roles	Estaciones	Estado	Dar Acceso	Eliminar	Cambiar Estado
ABOGADO DIRECCION REGISTRO	X-CALIDADTLBM	ACTIVA		<input type="checkbox"/>	
ACTUACIONES ADMINISTRATIVAS	X-CALIDADTLBM	ACTIVA		<input type="checkbox"/>	
ADMINISTRADOR IMPRESION	X-CALIDADTLBM	ACTIVA		<input type="checkbox"/>	
ADMINISTRADOR NACIONAL	X-CALIDADTLBM	ACTIVA		<input type="checkbox"/>	
ADMINISTRADOR REGIONAL	X-CALIDADTLBM	ACTIVA		<input type="checkbox"/>	
AMPLIACION TRADICION	X-CALIDADTLBM	ACTIVA		<input type="checkbox"/>	
ANALISTA REPARTO NOTARIAL	X-CALIDADTLBM	ACTIVA		<input type="checkbox"/>	
AUDITOR	X-CALIDADTLBM	ACTIVA		<input type="checkbox"/>	
AUXILIAR ACTUACIONES ADMINISTRATIVAS	X-CALIDADTLBM	ACTIVA		<input type="checkbox"/>	

Los botones quedarán como una lista de chequeo, que al ser seleccionado uno o más, el cambio se aplique para todos los seleccionados.

En el formulario donde se están desplegando los roles en la parte inferior se debe agregar el formulario para notas informativas

-El sistema deberá presentar en pantalla las últimos 20 notas, en caso de tener más de este número el sistema deberá generar un hipervínculo para navegar en las siguientes páginas, adicionalmente deberá permitir tener la consulta por un rango de fecha De y Hasta, con el formato DDMMAAAA en lo posible al dar un clic en el campo se despliegue el calendario.

Resultados de la Consulta					
Listado					
Rango Fecha, desde:	Hasta:	Buscar			
Descargar En Excel					
Nombre de Novedad	Fecha	Usuario	Tipo Nota	Descripción	
ACTIVAR-INACTIVAR ROL	04/10/2017	USUARIO.PRUEBAS	Nombramiento	SE ACTIVA ROL POR NOMBRAMIENTO	-
ACTIVAR-INACTIVAR ROL	04/10/2017	USUARIO.PRUEBAS	Incapacidad	SE INCAPACIA POR AL RAZON	-

-Si el estado del rol es Activo y si se selecciona en la lista de chequeo este debe cambiarse a Inactivo; si está en inactivo este deberá cambiar a Activo.

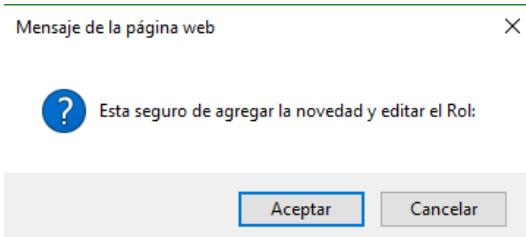
-En el caso en que se haya seleccionado para eliminar, deberá eliminarse el rol o los roles seleccionados.

-El sistema valida que no se elimine o se cambie el estado de un mismo o de diferentes roles al mismo tiempo, es decir si se selecciona un rol en cambiar estado no deberá permitir seleccionar en ningún rol la opción de eliminar, por lo que deberá bloquear toda la columna de la lista de chequeo de eliminar o viceversa, esto con el fin que se elimine o cambie un estado cuando no requería hacerlo.

-En la lista de chequeo se debe permitir seleccionar o deseleccionar un rol con un clic.

- En la lista de chequeo se selecciona o deselecciona un rol con un clic.

-Una vez el sistema efectúe las validaciones tal como hoy en día se efectúan, tras dar clic en el botón Agregar de la nota informativa, se deberá presentar el siguiente mensaje de alerta, Está seguro de agregar la novedad y editar el rol.



- Una vez el usuario de clic en el botón Aceptar deberá quedar el cambio de los roles o de eliminarlos si esta fue la selección, así como quedar guardada la nota informativa en la tabla asignada para tal fin.
- Las notas informativas estarán disponibles para la consulta cada vez que ingrese por este proceso por lo que en la parte inferior de la pantalla deberá existir un campo llamado notas informativa.
- En esta opción se desplegará el listado de las notas que ya haya tenido el funcionario que se está editando las cuales se deberán presentar en orden descendente de las más reciente a la más antigua, con la siguiente estructura:

Resultados de la Consulta																													
Listado																													
Rango Fecha, desde:		Hasta:	Buscar																										
Descargar En Excel																													
<table border="1"> <thead> <tr> <th>Nombre de Novedad</th><th>Fecha</th><th>Usuario</th><th>Tipo Nota</th><th>Descripción</th><th></th></tr> </thead> <tbody> <tr> <td>ACTIVAR-INACTIVAR ROL</td><td>17/10/2017</td><td>USUARIO.PRUEBAS</td><td>Incapacidad</td><td>PRUEBA DE CAMBIO DE ESTADO CCCDDDDDDDD</td><td>[Imprimir] [Ver adjunto]</td></tr> <tr> <td>ACTIVAR-INACTIVAR ROL</td><td>04/10/2017</td><td>USUARIO.PRUEBAS</td><td>Nombramiento</td><td>SE ACTIVA ROL POR NOMBRAMIENTO</td><td>[Imprimir] [Ver adjunto]</td></tr> <tr> <td>ACTIVAR-INACTIVAR ROL</td><td>04/10/2017</td><td>USUARIO.PRUEBAS</td><td>Incapacidad</td><td>SE INCAPACIA POR AL RAZON</td><td>[Imprimir] [Ver adjunto]</td></tr> </tbody> </table>						Nombre de Novedad	Fecha	Usuario	Tipo Nota	Descripción		ACTIVAR-INACTIVAR ROL	17/10/2017	USUARIO.PRUEBAS	Incapacidad	PRUEBA DE CAMBIO DE ESTADO CCCDDDDDDDD	[Imprimir] [Ver adjunto]	ACTIVAR-INACTIVAR ROL	04/10/2017	USUARIO.PRUEBAS	Nombramiento	SE ACTIVA ROL POR NOMBRAMIENTO	[Imprimir] [Ver adjunto]	ACTIVAR-INACTIVAR ROL	04/10/2017	USUARIO.PRUEBAS	Incapacidad	SE INCAPACIA POR AL RAZON	[Imprimir] [Ver adjunto]
Nombre de Novedad	Fecha	Usuario	Tipo Nota	Descripción																									
ACTIVAR-INACTIVAR ROL	17/10/2017	USUARIO.PRUEBAS	Incapacidad	PRUEBA DE CAMBIO DE ESTADO CCCDDDDDDDD	[Imprimir] [Ver adjunto]																								
ACTIVAR-INACTIVAR ROL	04/10/2017	USUARIO.PRUEBAS	Nombramiento	SE ACTIVA ROL POR NOMBRAMIENTO	[Imprimir] [Ver adjunto]																								
ACTIVAR-INACTIVAR ROL	04/10/2017	USUARIO.PRUEBAS	Incapacidad	SE INCAPACIA POR AL RAZON	[Imprimir] [Ver adjunto]																								
<p align="center">« Ant   1   Sigue »</p>																													

El sistema presentara en pantalla las últimos 20 notas, en caso de tener más de este número el sistema deberá generar un hipervínculo para navegar en las siguientes páginas, adicionalmente deberá permitir tener la consulta por un rango de fecha De y Hasta, con el formato DDMMA AAAA en lo posible al dar un clic en el campo se despliegue el calendario.

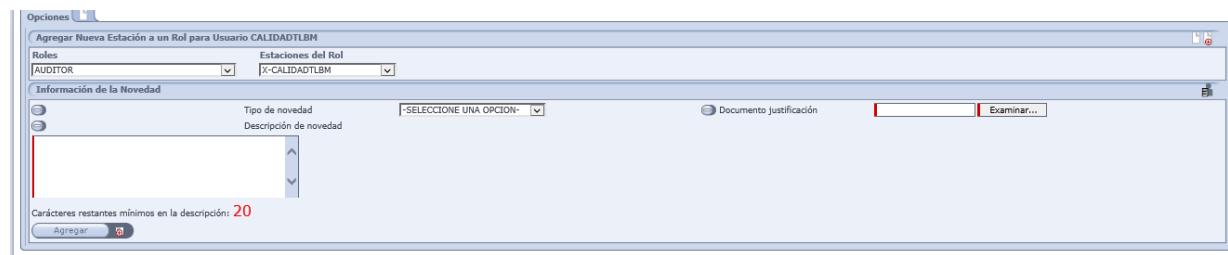
-Adicionalmente esta consulta de notas tendrá la opción de exportarse a Excel independiente el número de notas que tenga el funcionario.

Esta opción también tendrá dos botones al final de cada nota informativa “Imprimir” y “Ver adjunto”.

- Al dar clic en el botón “Imprimir” el sistema generará la impresión de la nota.
- Al dar clic en el botón “Ver adjunto” el sistema permitirá descargar el adjunto que se cargó en la nota.

## Agregar ROL

En los casos en los que a través de la administración de usuarios se requiera agregar un rol, cuando el usuario de un clic en el botón agregar tal como se presenta en la siguiente imagen, el sistema habilitara el formulario de la nota.



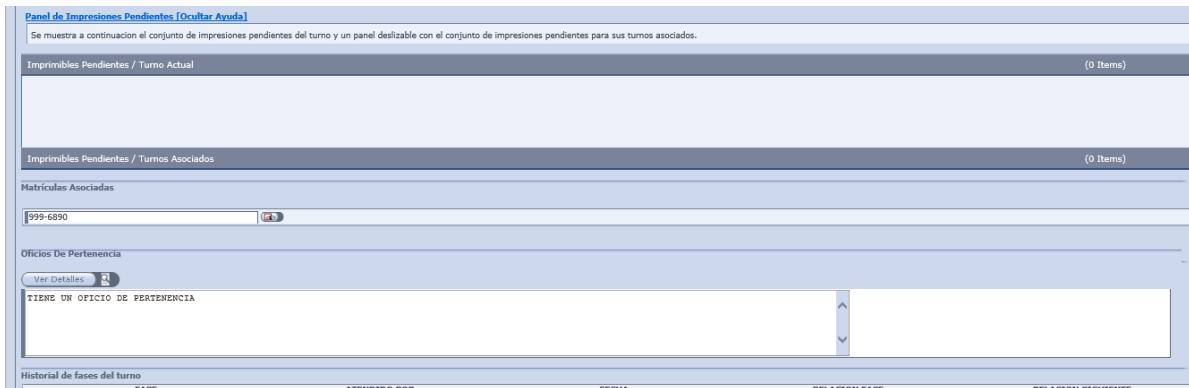
- En la parte inferior del formulario aparecerá la tabla de histórico de notificaciones creadas con formato de paginado.
- El sistema presentara en pantalla las últimos 20 notas, en caso de tener más de este número el sistema deberá generar un hipervínculo para navegar en las siguientes páginas, adicionalmente deberá permitir tener la consulta por un rango de fecha De y Hasta, con el formato DDMMA AAA en lo posible al dar un clic en el campo se despliegue el calendario.



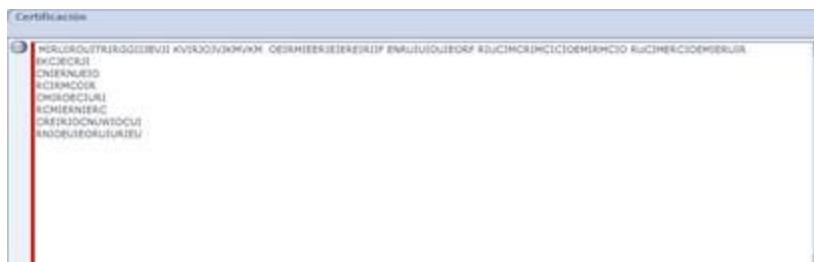
Nombre de Novedad	Fecha	Usuario	Tipo Nota	Descripción
ACTIVAR-INACTIVAR ROL	17/10/2017	USUARIO-PRUEBAS	Incapacidad	PRUEBA DE CAMBIO DE ESTADO CCCCCDDDDDDDD
ACTIVAR-INACTIVAR ROL	04/10/2017	USUARIO-PRUEBAS	Nombramiento	SE ACTIVA ROL POR NOMBRAMIENTO
ACTIVAR-INACTIVAR ROL	04/10/2017	USUARIO-PRUEBAS	Incapacidad	SE INCAPACIA POR AL RAZON

## 2155. Oficio pertenencia

En el aplicativo SIR para todos los usuarios que tengan la opción de consulta de turno a través de pantallas administrativas por la opción Ver detalle de turno, una vez se consulte el turno y si éste tiene asociado un certificado tipo pertenencia presentara un botón que permita hacer la consulta en pantalla así.



Una vez el usuario de un clic en este botón “Oficio de Pertenencia (Ver/Ocultar)” deberá desplegarse el campo con la información que se ingresó en la fase Oficio de Pertenencia. Este texto no es editable es solo lectura.



Una vez el usuario deseé ocultar la información del certificado, deberá permitirle ocultar este campo dando un clic nuevamente en el botón “Oficio de Pertenencia (Ver/Ocultar)”

Nota: En caso de que la información del texto de pertenencia sea más amplia que la ventana donde se muestra se debe incluir con un scroll lateral derecho que permita la completa navegación.



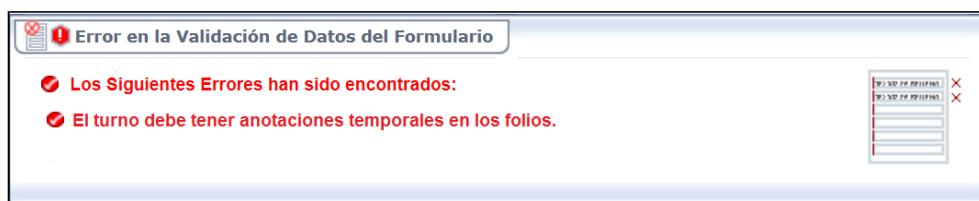
## DESPLIEGUE 7.1.0 07/04/2017



### 2099.VALIDACION.AVANCE.TURNOS.FIRMAR.REGISTRO

Para turnos con resolución inscrito e inscrito parcialmente, cuando en la fase de firmar registro se dé clic en el botón aceptar, el sistema deberá verificar que los folios asociados al turno en trámite contengan datos temporales y así dar continuidad con el proceso, avanzándolo a la fase de certificados asociados.

En caso de que los folios no tengan datos temporales, el aplicativo deberá mostrar en la misma ventana donde se está realizando la operación el siguiente mensaje de validación: El turno debe tener anotaciones temporales en los folios.



## DESPLIEGUE 7.0.0 04/02/2017

### 2079.INACTIVAR.PROCESO.CONSULTAS.INDIVIDUALES

De acuerdo a lo definido en la Circular No 199 del 25 de Enero del 2017, se relaciona a continuación las modificaciones para el proceso de consultas;

- Se inactivará del menú del rol, el rol cajero consultas para todas las estaciones que lo tengan asignado en el aplicativo SIR, proceso por el cual se radicaban las solicitudes de búsqueda de información almacenados en el sistema.
- En la opción de pantallas administrativas – administración de perfiles, al momento de agregar una nueva estación a un rol, en los campos Roles del Sistema y Roles, no se listará el rol cajero consultas.
- Basados en el cambio que se efectuará para el proceso de consultas tendrá como alcance, efectuar el ajuste en los reportes 25: Informe recaudos del día (tiracaja) y 94: Informe recaudos de consultas del día xx realizados por yy, con el fin de que en el campo usuario se listen los usuarios activos e inactivos del sistema.

-Así mismo se llevará acabo la depuración del rol usuario operativo consultas, ya que el registrador será quien lo deberá tener asignado y este a su vez será el responsable de delegar este rol a algún funcionario de la ORIP bajo su responsabilidad.

## **DESPLIEGUE 6.0.0**

**25/01/2017**

### **2062.TARIFAS.REGISTRALES.2017**

Basados en las tarifas definidas en la Resolución No 0450 de fecha 20 de Enero del 2017, se requiere realizar en el aplicativo SIR los siguientes ajustes:

- Proceso de Registro

- Para el proceso de registro, en la liquidación de los actos o negocios jurídicos que por su naturaleza tienen cuantía, se les aplicará la tarifa que corresponda, de conformidad con la siguiente tabla:

ACTOS CON CUANTIA		
SMLV INICIO	SMLV FINAL	TARIFA POR 1000
>= 10 SMMLV	<= 150 SMMLV	4.71 x 1.000
> 150 SMMLV	<= 300 SMMLV	5.30 x 1000
> 300 SMMLV	<= 500 SMMLV	5.60 x 1.000
> 500 SMMLV		5.70 x 1.000

- Los actos o negocios jurídicos con cuantía inferior a 25 SMMLV tendrán una tarifa de sesenta mil pesos (34.000).
  - Cuando se inscriba un documento con anterioridad al 31 de diciembre de 1990 y la liquidación contenga los actos: 1.Venta, 3. Remates, 4.Sucesiones, 6.Permuta, 36. Dación de pago y demás asignados por la SNR, el sistema causará derechos registrales por valor de diecisiete mil seiscientos pesos (\$ 18.600), como se define en el artículo 14 de la Resolución No 0450 del 20 de Enero del 2017.

- Proceso de Fotocopias

Se requiere que el imprimible del proceso de Fotocopias “Liquidación de Documentos a ser Fotocopiados”, refleje la normatividad correcta por la cual se establecen las tarifas aplicadas para la expedición de documentos que reposan en los archivos de la Oficina de

Registro.

En el párrafo de la normatividad, el texto a actualizar es:

“Según Resolución 0450 del 20 de Enero de 2017, se cobran las siguientes tarifas:

- Documentos almacenados en medio óptico o microfilmado \$ 1.100.00 por cada página reproducida.
- Documentos que reposen en los archivos físicos de la respectiva oficina de registro \$600.00 por cada página fotocopiada.

• Proceso Certificados

Los certificados que según la ley corresponde expedir a los registradores de instrumentos públicos, causarán derechos así:

- Los certificados de tradición y libertad que correspondan a predios de mayor extensión, causarán derechos por la suma de cuarenta mil pesos (\$34.000) cada uno, haciendo referencia a los certificados que cuenten con un número mayor o igual a 150 anotaciones.
- Cuando un certificado sea de tipo pertenencia, ampliación de tradición o masivo y tenga asociado un predio de mayor extensión, para la liquidación predominará el valor por este concepto, es decir \$34.000 sin afectar el flujo del proceso.
- Para los certificados asociados a registro y se agregue una matrícula de mayor extensión el sistema liquidará el valor definido por este concepto, es decir \$34.000 sin afectar el flujo del proceso.

Así mismo para realizar la actualización de las tarifas en el aplicativo SIR por los demás conceptos, se efectuará con la información que anualmente envía la SNR donde se relaciona los ítems y los valores que están sujetos a cambios discriminados en la Resolución 0450 del 20 de Enero del 2017.

**DESPLIEGUE 5.2.3**

**18/11/2016**

**2028.AJUSTE.REPORTES.166.167.G02**

De acuerdo al análisis realizado a la funcionalidad de los reportes 166, 167 y G02, se determina

realizar los siguientes cambios y ajustes:

- Reporte 166: Documentos Repartidos Mensualmente: Se debe tener en cuenta para la generación del reporte los usuarios con el rol calificador en estado activo e inactivo, trasladados y/o tuvieron asignado este rol.
- Reporte 167: Documentos Pendientes por Calificar por cada Abogado: En los parámetros de generación se modificará el texto de los ítems “Fecha Inicial Radicación” por “Fecha Inicial Asignación” y “Fecha Final Radicación” por “Fecha Final Asignación”, como se muestra a continuación en la imagen.

The screenshot shows a software window titled 'Reporte 167' with the sub-header 'Reporte Documentos Pendientes Por Calificar'. It contains three input fields with icons: 'Fecha Inicial Asignación' (27/09/2016), 'Fecha Final Asignación' (27/09/2016), and 'Círculo Origen' (dropdown menu showing '-SELECCIONE UNA OPCION-'). Below the inputs are two buttons: 'Regresar' and 'Observar'.

Así mismo se agregará el título “TURNOS ASIGNADOS POR REPARTO” a la sección que relaciona los usuarios con el rol calificador en estado activo e inactivo y tienen turnos pendientes por calificar, igualmente se creará la sección “TURNOS REASIGNADOS, REANOTADOS Y RESTITUIDOS” donde se listarán los usuarios y el número de turnos que fueron asignados por estos conceptos, los cuales están pendientes por calificar.



**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS**

**DOCUMENTOS PENDIENTE POR CALIFICAR**

Impreso el 08 de Septiembre de 2016 a las 11:34:53 AM

**REALIZADO POR:** JOSE.GARCIA

**DESDE: 01-08-2016 HASTA: 08-09-2016**

Página 1

**TURNOS ASIGNADOS POR REPARTO**

Círculo	Usuario	September-16	Total Usuario
BUCARAMANGA - 300	ESPERANZA.RINCON	1	1
	GLADYS.PINILLA	20	20
	MARIA.PENALOZA	1	1
	Total Usuarios	22	22
	Total Mes	22	22

**TURNOS REASIGNADOS, REANOTADOS Y RESTITUIDOS**

Círculo	Usuario	September-16	Total Usuario
BUCARAMANGA - 300	ESPERANZA.RINCON	1	1
	GLADYS PINILLA	20	20
	MARIA.PENALOZA	1	1
	Total Usuarios	22	22
	Total Mes	22	22

- Reporte G02: Informe Estadístico de turnos por Calificador: Se deberá efectuar cambios en los parámetros de generación, eliminando el ítem usuario. Ver Imagen.

Reporte G02

Informe de Turnos por Calificador

Fecha Inicial Reparto:

Fecha Final Reparto:

Círculo Registral:

**Regresar** **Observar**

En las columnas que contienen la información y el conteo de los turnos es necesario efectuar los siguientes cambios:

- Calificador: Se deberá listar los usuarios con este rol en estado activo e inactivo, trasladados y/o tuvieron asignado este rol en algún momento.
- Turnos Repartidos: Correspondrá al conteo de los turnos que fueron asignados por reparto normal a través del rol jefe operativo.
- Turnos Reasignados, Restituidos y Reanotados: Correspondrá al conteo de los turnos que fueron reanotados, restituidos y reasignados por la opción de pantallas administrativa para cada criterio. Igualmente se tendrá en cuenta los turnos reasignados por el sistema, que son los turnos relacionados a un calificador diferente al calificador inicial, ya que el usuario se encuentra con este rol inactivo y debido a su estado no podrá recibir turnos para continuar nuevamente con el trámite.
- Turnos Pendientes (Calificación): Deberá contar los turnos que se encuentren en la fase de calificación.
- Turnos Redireccionados a A.S Correcciones: Se tendrá en cuenta para el conteo, los turnos que hayan sido redireccionados desde calificación y se encuentren en las siguientes fases: Revisión Inicial, Hoja de Ruta, Revisión Final, Revisión y Análisis, Ejecutar Corrección Simple y Revisar Aprobar
- Turnos en Tramite: El conteo pertenecerá a los turnos que se encuentran en las siguientes fases: Confrontación correctiva, Digitación calificaciones, Aprobar corrección calificación, Custodia pagos mayores, Notificar ciudadano pago mayor, Registrar pagos mayores, Notificar funcionario pagos mayores, Corrección encabezado, Mesa control y Firmar registro.
- Sección Resumen General: Debajo del “TOTAL TURNOS REPARTIDOS”, se agregará el ítem “TOTAL TURNOS REASIGNADOS, RESTITUIDOS Y REANOTADOS ” relacionando el conteo por estos criterios.
- En la sección “NOTA” se modificará el segundo párrafo cambiando el texto “TURNOS EN TRAMITE; Es la sumatoria de los turnos que se encuentran en la fase Mesa de control y

Firma registro.” por “TURNOS EN TRAMITE; Es la sumatoria de los turnos que se encuentran en las fases: Confrontación correctiva, Digitación calificaciones, Aprobar corrección calificación, Custodia pagos mayores, Notificar ciudadano pago mayor, Registrar pagos mayores, Notificar funcionario pagos mayores, Corrección encabezado, Mesa control y Firmar registro.

Así mismo se agregará otro párrafo donde se indicará el texto “TURNOS REASIGNADOS, RESTITUIDOS Y/O REANOTADOS: Es la sumatoria de los turnos que fueron reasignados, restituidos y/o reanotados.”

Superintendencia de Notariado y Registro Ministerio del Interior y de Justicia República de Colombia www.senpermnotario.gov.co		OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS BUCARAMANGA - 300								
		INFORME ESTADÍSTICO DE TURNOS POR CALIFICADOR DE								
		2016-08-01 A 2016-09-08								
		REALIZADO POR USUARIO PRUEBAS								
Impreso el 26 de Septiembre de 2016 a las 14:33:00 PM										
Página 1										
CALIFICADOR	TURNOS REPARTIDOS	TURNOS REASIGNADOS RESTITUIDOS REANOTADOS	TURNOS CALIFICADOS	TURNOS INSCRITOS	TURNOS DEVUeltos AL PUBLICO SIN REGISTRAR	TURNOS INSCRITOS PARCIALMENTE	TURNOS PENDIENTES (CALIFICACION)	TURNOS REDIRECCIONADOS A.S CORRECCIONES	TURNOS EN TRAMITE	
ANATARAZA NA CAICEDO	12	0	10	7	1	2	1	0	1	
ESPERANZA RINCON DE SALCEDO	1	17	0	0	0	0	1	0	0	
GLADYS PINILLA ARISMENDY	59	0	0	0	0	0	59	0	0	
<b>RESUMEN GENERAL</b>										
<b>TOTAL TURNOS REPARTIDOS</b>	<b>72</b>									
<b>TOTAL TURNOS CALIFICADOS</b>	<b>10</b>									
<b>TOTAL TURNOS REASIGNADOS, RESTITUIDOS Y/O REANOTADOS</b>	<b>17</b>									
<b>TOTAL TURNOS INSCRITOS</b>	<b>7</b>									
<b>TOTAL TURNOS DEVUeltos AL PUBLICO</b>	<b>1</b>									
<b>TOTAL TURNOS INSCRITOS PARCIALMENTE</b>	<b>2</b>									
<b>TOTAL TURNOS EN TRAMITE</b>	<b>1</b>									
<b>TOTAL TURNOS REDIRECCIONADOS A.S Y CORRECCIONES</b>	<b>0</b>									
<b>TOTAL TURNOS PENDIENTES POR CALIFICAR</b>	<b>61</b>									
<b>NOTA:</b> TURNOS CALIFICADOS: Es la sumatoria de los turnos Devueltos al público, inscritos e Inscritos Parcialmente.										
TURNOS EN TRAMITE: Es la sumatoria de los turnos que se encuentran en las fases: Confrontación Correctiva, Digitación Calificaciones, Aprobar Corrección Calificación, Custodia Pagos Mayores, Notificar Ciudadano Pago Mayor, Registrar Pagos Mayores, Notificar Funcionario Pagos Mayores, Corrección Encabezado, Mesa Control Y Firmar Registro.										
TURNOS REASIGNADOS, RESTITUIDOS Y/O REANOTADOS: Es la sumatoria de los turnos que fueron reasignados, restituidos y/o reanotados.										

## DESPLIEGUE 5.2.2

09/09/2016

### 2015.REPORTE ESTADISTICO.POR NATURALEZA.JURIDICA

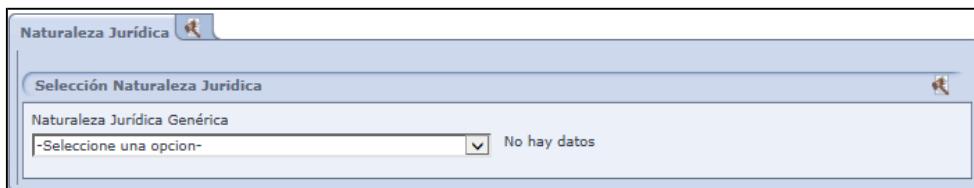
Se requiere crear un reporte que permita listar las matrículas que tengan relacionada la naturaleza jurídica según la necesidad de la información a generar, a continuación se establecen las condiciones a tener en cuenta para su diseño:

- El reporte será nombrado como Reporte 179: REPORTE ESTADISTICO POR NATURALEZA JURIDICA, deberá ser asignado al rol Reportes Nacional y su generación será en formato Excel.
- Los parámetros de generación del reporte serán basados en los siguientes campos;

- Fecha Inicial: Corresponde a la fecha de inicio en el rango de la consulta para la información solicitada, el formato de la fecha estará definida por dd/mm/aaaa, así mismo en este campo irá el icono del calendario y al dar clic abrirá una ventana emergente que permitirá al usuario seleccionar la fecha.
- Fecha Final: Corresponde a la fecha de terminación en el rango de la consulta para la información solicitada, el formato de la fecha estará definida por dd/mm/aaaa así mismo en este campo irá el icono del calendario y al dar clic abrirá una ventana emergente que permitirá al usuario seleccionar la fecha.

Cuando se escoja las fechas (inicial – final) y se genere el reporte, en el detalle relacionará las anotaciones que tengan información con la fecha de radicación del rango de fechas seleccionado en los parámetros fecha inicial y fecha final.

- Círculo Registral: En este campo se listarán las ORIPs (Círculo Registral – Departamento) que se encuentran en el aplicativo SIR y la opción TODOS.
- Naturaleza Jurídica: Este ítem estará compuesto por dos campos (código – descripción) que permitirán al usuario digitar el código o seleccionar la naturaleza jurídica. En el caso de elegir la opción para escoger el dato, se deberá dar clic en el icono situado al lado del campo de texto, el sistema tendrá que desplegar una ventana emergente denominada “Naturaleza Jurídica” donde mostrará inicialmente el campo Naturaleza Jurídica Genérica y contendrá el listado de grupos por naturalezas jurídicas.



Después de escogerse esta información se visualizará otro campo en blanco y cuando se presione clic en la flecha , se desplegará en una lista la última versión de las naturalezas jurídicas que están asociadas a esta categoría.

Naturaleza Jurídica

Selección Naturaleza Jurídica

Naturaleza Jurídica Genérica

FALSA TRADICION

- Seleccione una opción:

- ADJUDICACION LIQUIDACION SOCIEDAD CONYUGAL DERECHOS Y ACCIONES - 0615
- ADJUDICACION SUCESION DERECHOS Y ACCIONES - 0601
- ADJUDICACION SUCESION GANANCIALES - 0602
- AFFECTACION A VIVIENDA FAMILIAR SOBRE MEJORAS EN PREDIO AJENO PAR. ART. 5 LEY 258/96 - 0603
- COMPENSACION INMUEBLE EN PROCESO DE RESTITUCION LITERAL J) ART. 91 Y ART. 97 LEY 1448 DE 2011 - 0621
- COPRAVENTA DE COSA AJENA - 0604
- COPRAVENTA DERECHOS GANANCIALES - 0605
- COPRAVENTA DERECHOS Y ACCIONES - 0607
- COPRAVENTA MEJORAS EN SUELLO AJENO CON ANTECEDENTE REGISTRAL - 0616
- COPRAVENTA POSESION CON ANTECEDENTE REGISTRAL - 0608
- DACION EN PAGO DE DERECHOS Y ACCIONES - 0617
- DECLARACION MEJORAS EN PREDIO AJENO PAR. ART. 5 LEY 258/96 - 0609
- DEclaracion de posesion regular art 1 ley 1183 de 2008 - 0619
- DONACION DERECHOS Y ACCIONES - 0610
- ENACION GANANCIALES - 0611
- FALSA TRADICION - 0600
- PATRIMONIO DE FAMILIA SOBRE MEJORAS EN PREDIO AJENO PAR. ART. 5 LEY 258/96 - 0612
- REMADE DERECHOS Y ACCIONES - 0613
- REMADE GANANCIALES - 0614
- RESTITUCION MATERIAL AL POSEEDOR LITERAL H) ART. 91 LEY 1448 DE 2011 - 0620
- TRANSFERENCIA DE CUERPO CIERTO TENIENDO SOLO DERECHOS DE CUOTA CON ANTECEDENTES REGISTRALES - 0605
- TRANSFERENCIA DE POSESION CON ANTECEDENTE REGISTRAL - 0618

Para finalizar con la selección de la naturaleza jurídica se dará clic en el botón agregar, el sistema cerrará la ventana emergente y relacionará el dato en el campo de texto.

- Tipo Predio: En este campo se relacionarán las opciones de TODOS, RURAL, URBANO Y SIN INFORMACIÓN.

Reporte 179

Reporte 179: Estadística Declaración Judicial de Pertenencia

Fecha Inicial: 18/08/2016

Fecha Final: 18/08/2016

Círculo Registral: SELECCIONE UNA OPCIÓN

Naturaleza Jurídica: Código Descripción

Tipo de Predio: SELECCIONE UNA OPCIÓN

Regresar Observar

- En el Reporte 179: REPORTE.ESTADISTICO.POR.NATURALEZA.JURIDICA se deberá mostrar en el encabezado la información correspondiente al logo definido por la entidad, nombre de la ORIP y código del círculo, nombre del reporte, rango de fechas de generación (Inicial – Final), el usuario que realizó la solicitud de generación del reporte, fecha, hora de impresión del reporte y el número de página.

Así mismo mostrará una nota que contiene el siguiente texto: "NOTA: LA INFORMACIÓN RELACIONADA, CORRESPONDE A LAS ANOTACIONES INSCRITAS CON FECHA DE RADICACION EN ESTADO DEFINITIVO"



OFICINA DE REGISTRO DE INSTRUMENTOS  
Reporte Estadístico por Naturaleza Jurídica

Impreso el 05 de Septiembre de 2016 a las 15:26:57 PM

REALIZADO POR: USUARIO.PRUEBAS DESDE: 02-09-2016

ORIP: BUCARAMANGA - 300

Página 1

HASTA 02-09-2016

NOTA: LA INFORMACIÓN RELACIONADA, CORRESPONDE A LAS ANOTACIONES INSCRITAS CON FECHA DE RADICACIÓN EN ESTADO DEFINITIVO

- En el detalle del reporte se mostrará las siguientes columnas:

- Naturaleza Jurídica: En este campo se relacionará el nombre de la naturaleza jurídica seleccionada en los parámetros de generación del reporte.
- Tipo de Predio: En este campo se relacionará el tipo de predio seleccionado en los parámetros de generación del reporte.
- Departamento: Corresponde al nombre del departamento donde pertenece la ORIP.
- Municipio: Corresponde al nombre del Municipio donde pertenece la ORIP.
- ORIP: Es el nombre del círculo registral seleccionado en los parámetros de generación del reporte.
- Matrícula: Número de la matrícula que contiene la naturaleza jurídica seleccionada en los parámetros de generación del reporte.
- Radicación: Es el número del turno que dio origen a la anotación o la información que se ingresa en este campo al momento de crear o modificar la anotación de una matrícula. En el caso de que no se digite información, en el reporte se mostrará el campo en blanco.
- Anotación: Es el número de la anotación que tiene relacionada la matrícula con la naturaleza jurídica seleccionada en los parámetros de generación del reporte.

NATURALEZA JURÍDICA	COMPROVANTIA (0125)	TIPO PREDIO:	URBANO		
DEPARTAMENTO	MUNICIPIO	ORIP	MATRÍCULA	RADICACIÓN	ANOTACIÓN
VALLE	BUENAVENTURA	BUENAVENTURA	372-20719	2015-372-6-184	3
VALLE	BUENAVENTURA	BUENAVENTURA	372-20718	2015-372-6-184	3
VALLE	BUENAVENTURA	BUENAVENTURA	372-13870	2015-372-6-206	3
VALLE	BUENAVENTURA	BUENAVENTURA	372-53700	2014-372-6-2788	1
Subtotal por ORIP BUENAVENTURA:		4			
VALLE	SEVILLA	SEVILLA	382-19355	2015-382-6-109	3
VALLE	CAICEDONIA	SEVILLA	382-21779	2014-382-6-2193	7
VALLE	CAICEDONIA	SEVILLA	382-23968	2015-382-6-114	3
VALLE	CAICEDONIA	SEVILLA	382-23925	2015-382-6-52	3
VALLE	SEVILLA	SEVILLA	382-23589	2015-382-6-126	4
VALLE	SEVILLA	SEVILLA	382-13981	2015-382-6-125	3
VALLE	SEVILLA	SEVILLA	382-18679	2015-382-6-149	7
VALLE	SEVILLA	SEVILLA	382-15327	2015-382-6-80	6
VALLE	SEVILLA	SEVILLA	382-5621	2015-382-6-104	4
Subtotal por ORIP SEVILLA:		9			
Total por Departamento VALLE:		13			

- En la parte final del detalle del reporte 179, se visualizará la sección de Subtotal por ORIP y el Total por Departamento, donde se relacionará el conteo por cada concepto.

## DESPLIEGUE 5.2.1

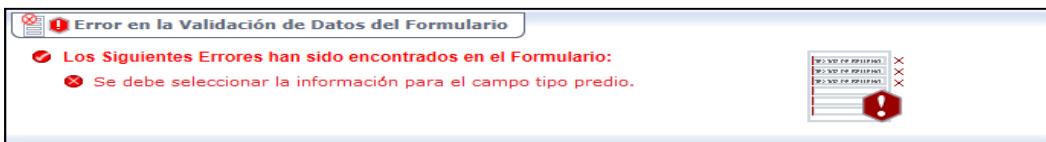
12/08/2016

### 1607.AJUSTE.BANCO.FRANQUICIA.FORMAS.PAGO

Para los procesos registrales que recaudan dinero actualmente en el aplicativo SIR al momento de hacer la radicación de un turno, para la forma de pago: Pago Electrónico PSE, en el campo “Banco/Franquicia” para la entidad financiera Bancolombia (Código 07) se relacionan las franquicias American Express, Credencial, Débito Maestro, Master Card y Visa; adicional a esto cuando se agrega un número de aprobación ya usado para esta misma forma de pago el sistema está generando el siguiente error de validación “Pago Invalido : El numero de aprobacion XXXXXXXX ingresado ya fue utilizado.”, y no permite continuar con el pago de la liquidación del turno.

En el aplicativo SIR en los procesos registrales que recaudan dinero para las formas de pago que manejen franquicia como lo son: Pago Electrónico PSE, Tarjeta crédito y Tarjeta débito, se requiere configurar para el identificador 07 correspondiente a Bancolombia con el id\_tipo\_franquicia 7, una vez realizada esta relación en el campo Banco/franquicia el sistema listará el nombre de Bancolombia. Teniendo como referencia el ajuste realizado para el banco/franquicia de Banco Occidente, como se muestra a continuación;

Así mismo cuando se agregue un número de aprobación ya usado para la forma de pago: Pago Electrónico PSE, el sistema generará el siguiente error de validación “Pago Invalido: El número de aprobación XXXXXXXX ingresado ya fue utilizado en el turno XXXX-XXX-X-XX.”, y no permitirá continuar con el pago de la liquidación del turno.



#### 1736.NO.DUPLICIDAD.PROCESO.DIGITAR.MASIVO.AL.REALIZAR.COPY

Se ha evidenciado en el sistema que al realizar un copiado masivo de linderos en un rango de folios en la sección “Agregar Linderos” con la acción de “Copiar a” para el proceso de registro en la fase Digitación Calificaciones y para el proceso de Correcciones en la fase Ejecutar Corrección Simple, botón “Digitación” la información copiada se duplica en cada matrícula.

En la fase Digitación Calificaciones para el proceso de Registro y en la fase Ejecutar Corrección Simple, en el botón “Digitación” para el proceso de Correcciones será eliminada la sección “Agregar Linderos” la cual permitía hacer el copiado de linderos de forma masiva para un rango de folios, visualizándose el formulario de la siguiente manera:

Fase Digitación Calificaciones:

**Superintendencia de Notariado y Registro**  
 Ministerio del Interior y de Justicia  
 República de Colombia

Libertad y Orden

Sistema de Información  
 Registral - Producción -  
 Versión 3.1.0  
 Fecha Actual: 06/5/2016

Turno: 2016-999-6-100

Menú Salir

folios

Matrículas  
 Revisar Folio

Agregar Complementación

DE  
 Número de Folio 999- | Cargar Complementación | Construir complementación

Complementación

A  
 Acción Rango de Folios  
 -SELECCIONE UNA OPCIÓN- Desde: -SELECCIONE UNA OPCIÓN- Hasta: -SELECCIONE UNA OPCIÓN- Copiar Complementación

Agregar Dirección

Tenga en cuenta las siguientes recomendaciones:  
 La dirección que se cargue o se escriba en los siguientes campos, se concatenará con la información del lote que se ingresó en el momento de la segregación.  
 La información del lote que se ingrese en el englobe, no se concatenará con la dirección que se cargue o se escriba, sólo se adiciona al nuevo folio.

DE  
 Número de Folio 999- | Cargar Dirección |

Dirección  
 -SELECCIONE UNA OPCIÓN- | -SELECCIONE UNA OPCIÓN- |

Especificación

A  
 Acción Rango de Folios  
 -SELECCIONE UNA OPCIÓN- Desde: -SELECCIONE UNA OPCIÓN- Hasta: -SELECCIONE UNA OPCIÓN- Copiar Dirección

OPCIONES DIGITACIÓN  
 Aceptar ✓

NOTAS

Notas Informativas

Fase Ejecutar Corrección Simple, botón “Digitación”

Para ingresar información de linderos en una matrícula en los procesos de registro y correcciones en las fases que se mencionarán a continuación solamente se podrá realizar de la siguiente manera:

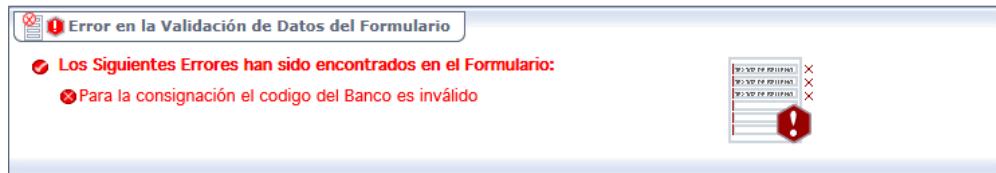
- Fase de Digitación Calificaciones: se deberá ingresar al botón Revisar Folio, en la sección “Descripción: Cabida y Linderos” en la cual se agregará la información correspondiente por este concepto y se guardará dando clic en el botón Guardar Edición Cabida y Linderos.
- Fase de Ejecutar Corrección Simple: se deberá ingresar al botón Editar Folio, en la sección Datos Folio en el campo “Cabida y Linderos”, siempre y cuando en la fase Revisión y Análisis haya sido asignado el permiso de Cabida y Linderos al turno.

## **1954.VALIDACION.BANCO.FRANQUICIA.CONSIGNACION**

Actualmente para la liquidación de turnos registrales que recaudan dinero el aplicativo SIR permite agregar la forma de pago Consignación sin información del Banco/Franquicia; adicionalmente para las formas de pago Pago Electrónico PSE, Tarjeta Crédito y Tarjeta Débito cuando el campo Banco/Franquicia no contiene información el sistema genera una excepción.

Se requiere que en una liquidación de turnos para cualquier proceso y subprocesso regstral que implique recaudo de dinero tales como: Certificados, Certificados masivos, Consultas, Fotocopias, Registro, Devoluciones y Mayor Valor al agregar las formas de pago Consignación, Pago Electrónico PSE, Tarjeta Crédito y Tarjeta Débito sin información en el campo Banco/Franquicia dependiendo la forma de pago seleccionada el sistema generará un mensaje de validación donde notifique lo siguiente:

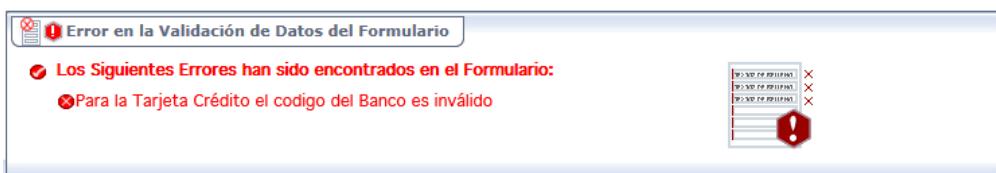
- Consignación



- Pago Electrónico PSE



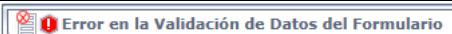
- Tarjeta Crédito



- Tarjeta Débito



Adicional a esto en dado caso de no seleccionar una forma de pago y se vaya a agregar la forma de pago, el sistema deberá notificar el siguiente mensaje de validación:



## 1972.CORRECCION.DE.TESTAMENTOS.ACTO.REVOCATORIA.DE. TESTAMENTO

Ante la incidencia presentada en el formulario de Inscripción de testamento con turno 2016-290-6- 7396, el aplicativo SIR, por regla de negocio (**un documento con acto de testamento debe radicarse con acto de liquidación de testamento**). Esto implica que el aplicativo no permite visualizar el formulario en el proceso de correcciones para corregir la información.

Lo anterior se aclara, toda vez que la oficina de registro de instrumentos públicos de Armenia, en el proceso de radicación de escritura 1332 del 25-04-2016 de la notaría primera de Armenia – Quindío, tomó como acto de liquidación Revocatoria de testamento, debido a la regla de negocio mencionada anteriormente el sistema no permite realizar dicha acción.

Por lo cual se solicita que el operador – ETB, realice lo pertinente para que la ORIP de Armenia, pueda ingresar al proceso de correcciones, en la fase ejecutar con el fin de corregir la información del campo “Observación”, contenida en la constancia de inscripción de testamentos.

Se requiere que en el proceso de correcciones, en la fase de solicitud en el campo Turno testamentos, el sistema permita asociar turno de testamento que contenga acto de revocatoria de testamento, conservando las demás validaciones que existen actualmente para esta sección, tales como: Acto Testamento, Resolución Inscrito, Estado Desanotado y Subtipo de solicitud Testamento.

Así mismo se debe modificar el mensaje de validación en el caso de que se intente ingresar un turno de testamento sin las condiciones requeridas por el aplicativo para dicha acción, como se muestra a continuación:



## DESPLIEGUE 5.2.0

07/07/2016

### 1156 USUARIOS\_ROLE\_INACTIVOS

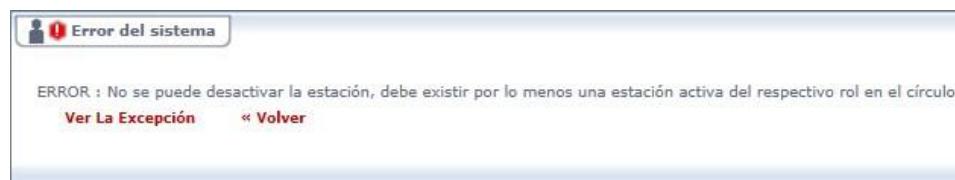
Actualmente el aplicativo SIR permite asignar turnos a roles con estado inactivo, cuando dichos roles están asignados a usuarios activos.

Cuando una estación tiene un rol asignado en estado inactivo, el sistema no debe permitir la asignación de turnos a estos roles. Con base en lo anterior se debe tener en cuenta las siguientes validaciones;

- Si el rol se encuentra en estado inactivo y tiene turnos asignados el sistema debe permitir trabajarlos y avanzarlos a la siguiente fase. Esto quiere decir que cuando se inactiva un rol a un usuario, los turnos que ya habían sido asignados a éste, el sistema debe permitir avanzarlos.
- En la opción de reasignación de turnos en el listado de funcionarios que tiene asignados los turnos, en el campo funcionario se deben listar todos los usuarios sin tener en cuenta el estado del rol (Activo – Inactivo).
- En la opción de reasignación de turnos en el listado de funcionarios a quien desea reasignarle los turnos del campo funcionario, el sistema solo debe listar los usuarios que tienen el rol activo.
- En la opción de reanotación de turnos en el campo de Funcionario Calificador, el sistema solo debe listar los usuarios que tienen el rol calificador activo.
- En la opción de Restitución de turnos en el campo lista de calificadores, el sistema solo debe listar los usuarios que tienen el rol calificador activo.
- En la opción Administración de Perfiles, cuando se vaya a cambiar de estado (inactivar) o eliminar un rol asociado a una estación, el sistema deberá validar si existe al menos una estación activa asignada a ese rol.

Pantalla Administrativa   Información del Usuario					
Perfil del Usuario USUARIO.PRUEBAS1		Estaciones		Estado	Dar Acceso
Roles	Estaciones	Estado	Dar Acceso	Eliminar	Cambiar Estado
ACTUACIONES ADMINISTRATIVAS	X-USUARIO.PRUEBAS1	ACTIVA	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
AUXILIAR DE CORRECCIONES	ESTACION EJEC.CORR 999	ACTIVA	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
CAJERO	ESTACION X-CAUERO.999	ACTIVA	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

En caso contrario, el aplicativo mostrará el siguiente mensaje “ERROR: No se puede desactivar la estación, debe existir por lo menos una estación activa del respectivo rol en el círculo.”, como se muestra a continuación:



Para los procesos que realicen redirección o tengan la opción de devolver a una fase anterior, el sistema no debe tener en cuenta el usuario con el rol inactivo para la nueva asignación del turno; los

procesos que se deben contemplar para esta validación son;

- Certificados
  - Ampliación tradición
    - \*Fase revisión inicial – Opción enviar a mayor valor.
    - \* Fase ampliación Tradición – Opción devolver a revisión inicial.
  - Rol Encargado Antiguo Sistema
    - \* Fase revisión inicial – Opción enviar a mayor valor.
- Correcciones
  - Rol Responsable de Correcciones
    - \* Fase revisión y análisis – Redireccionar Antiguo sistema, mayor valor y actuaciones administrativas.
    - \* Fase revisar y aprobar – Opción devolver auxiliar y deshacer cambios.
  - Rol Auxiliar correcciones
    - \* Fase ejecutar corrección simple – Opción devolver a revisión.
  - Rol Actuaciones administrativas
    - \*Fase actuaciones administrativas – Opción corrección simple, redireccionar antiguo sistema o mayor valor.
- Devoluciones
  - Rol Usuario administrativo
    - \* Fase resolución – Opción devolver análisis.
    - \* Fase recursos devolución – Opción devolver a resolución.
  - Rol Pago Devolución
    - \* Fase pago devolución – Opción devolver a resolución o devolver a recursos.
- Registro
  - Rol Calificador.
    - \*Fase calificación.
      - Opción información turno (corrección encabezado, enviar a confrontación).
      - Opción información folio enviar caso antiguo sistema.
  - Rol Mesa control registro.
    - \* Fase Mesa control.
    - Opción devolver a testamento.
    - Opción calificación.
  - Rol Coordinador Jurídico
    - \* Fase aprobar corrección calificación
    - Opción enviar a calificación o mesa de control.
  - Rol Registrador
    - \*Fase Firmar registro – Opción calificación.

- Antiguo Sistema

- Rol encargado antiguo sistema

\*Fase revisión final – Opción regresar a hoja de ruta

- Rol buscador antiguo sistema

\*Fase hoja de ruta – Opción enviar caso rechazar, más documentos o revisión inicial.

#### **1247.ASIGNACION.SOLICITANTE.CORRECCION.INTERNIA**

En el aplicativo SIR al enviar a correcciones un turno de registro desde la fase calificación el sistema genera un turno de corrección interna, donde el nombre del solicitante de éste turno es tomado de los datos de la solicitud del turno de registro, registrando los datos del ciudadano.

En el proceso de registro en la fase de calificación al momento de enviar un turno a una corrección interna el sistema solicita ingresar la nota informativa que determina el motivo por el cual se delega la calificación a la corrección, esta nota informativa debe visualizarse en la consulta del turno por Pantalla Administrativa - Ver Detalles de Turnos, tanto en el turno de registro, como en el turno de corrección interna.

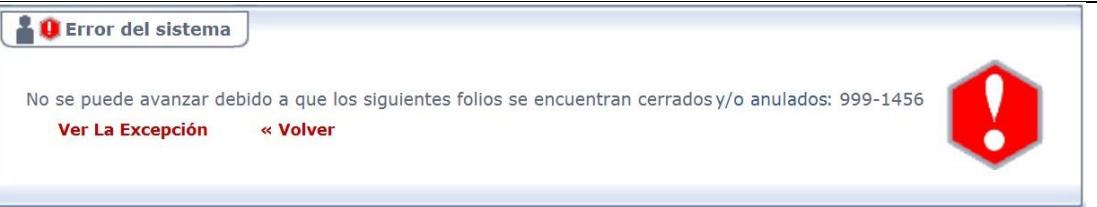
Así mismo en la sección datos del solicitante del turno de corrección derivado que se visualiza en el formulario de revisión y análisis y en la consulta de ver detalles de turnos, la información de los campos nombre del solicitante, tipo de documento del solicitante y número de documento del solicitante corresponderá al calificador del turno de registro que realizó la solicitud.

#### **1248.ASOCIAR.FOLIOS.CERRADOS.FASE.REVISION.ANALISIS**

Actualmente los folios en estado cerrado asociados a turnos de corrección que se encuentran en la fase de actuaciones administrativas, cuando el turno es enviado a la fase revisión y análisis el sistema permite eliminarlos, sin embargo al intentar agregarlos nuevamente al turno el sistema niega la acción generando la excepción que los folios se encuentran en estado cerrado. Por lo tanto se requiere ajustar esta funcionalidad para dar cumplimiento a Ley 1579 del 2012 Capítulo XVI, parágrafo del Artículo 67, donde cita “En los eventos en que la matrícula inmobiliaria se encuentre sometida a un trámite de actuación administrativa o judicial o de cualquiera otra índole, se expedirá el certificado de tradición y libertad, con la correspondiente nota de esta situación”.

Cuando un turno de correcciones este en la fase de revisión y análisis el sistema deberá permitir asociar folios en estado cerrado y anulado, basados en estas condiciones se deben tener en cuenta las siguientes validaciones:

1. El sistema debe permitir el redireccionamiento del turno a la fase de actuaciones administrativas para continuar con su respectivo trámite.
2. En el caso de que se intente avanzar los turnos de corrección con folios asociados en estado cerrado y anulado a la fase de Ejecutar Corrección Simple, el sistema mostrará el mensaje restrictivo “No se puede avanzar debido a que los siguientes folios se encuentran cerrados y/o anulados: XXX-XXXX”.



## 1249.AJUSTE.REPORTE.39.INFORME.DOCUMENTOS.CORRECCION

Actualmente en el aplicativo SIR el reporte 39: Informe de documentos en corrección, en la sección totales generales no se está realizando correctamente la sumatoria de los turnos que se relacionan en el detalle del reporte.

Se requiere ajustar el reporte 39: Informe de documentos en corrección, en la sección totales generales los siguientes ítems:

- Correcciones aprobadas desanotadas: En esta parte del reporte se debe tener en cuenta para la sumatoria, todos los turnos con resolución aprobada y que se encuentren en las fases de corrección desanotada, entrega corrección y finalizado.
- Correcciones en Tramite: En esta parte del reporte se debe tener en cuenta para la sumatoria, todos los turnos con resolución aprobada o negada y que se encuentren en las fases de revisión y análisis, ejecutar corrección simple, revisar aprobar, mesa control correcciones, actuaciones administrativas, ejecutar actuaciones administrativas, custodia pagos mayores, notificar funcionario pagos mayores, revisión inicial, hoja de ruta y revisión final.
- Correcciones Negadas Desanotadas: En esta parte del reporte se debe tener en cuenta para la sumatoria, todos los turnos con resolución negada y que estos se encuentren en las fases de corrección desanotada, entrega corrección y finalizado.

El total de correcciones radicadas debe corresponder a la sumatoria de los turnos relacionados en los ítems anteriormente descritos.

## 1252.REPORTE.TURNOS.CORRECCION.ENCABEZADO

Con el fin de que las Oficinas de Registro puedan realizar control y seguimiento de los turnos a los cuales se les realiza corrección de encabezado en el proceso registro, es necesario crear un reporte que muestre esta información.

Se requiere crear el reporte donde se relacionen los turnos enviados desde el proceso de calificación a corrección de encabezado en el aplicativo SIR.

El reporte será nombrado como Reporte 178: Informe de Turnos Corrección Encabezado y deberá ser asignado a los roles registrador, administrador regional, administrador nacional, coordinador jurídico, consulta reportes nacional y consulta nacional.

Los parámetros de generación del reporte serán basados en los siguientes campos;

- Fecha Inicial: Corresponde a la fecha de inicio en el rango de la consulta para la información solicitada, el formato de la fecha estará definida por dd/mm/aaaa, así mismo en este campo irá el icono del calendario y al dar clic abrirá una ventana emergente que permitirá al usuario seleccionar la fecha.

- Fecha Final: Corresponde a la fecha de terminación en el rango de la consulta para la información solicitada, el formato de la fecha estará definida por dd/mm/aaaa así mismo en este campo irá el icono del calendario y al dar clic abrirá una ventana emergente que permitirá al usuario seleccionar la fecha.
- Usuario: Este campo pertenece a los usuarios que tengan asignado el rol calificador, se listará la opción de todos como también el nombre de los calificados.
- Círculo Registral: En este campo solo se listará la ORIP a la cual pertenece el usuario que está realizando la consulta, en el caso de que el usuario tenga asignado el rol de consulta nacional se listarán todas las Orips del aplicativo SIR.

En el Reporte 178: Informe de Turnos Corrección Encabezado se deberá mostrar en el encabezado la información correspondiente al logo definido por la entidad, nombre y código del círculo, nombre del reporte, rango de fechas de generación (Inicial – Final), el usuario que realizó la solicitud de generación del reporte, fecha y hora de impresión del reporte y el número de página como se muestra a continuación.



En el detalle del reporte se deberá mostrar las siguientes columnas:

**USUARIO: JOSE.GARCIA**

TURNO	FECHA DE ENVÍO A CORRECCIÓN	FECHA DE MODIFICACIÓN	DATOS MODIFICADOS	USUARIO QUE SOLICITA LA CORRECCIÓN	USUARIO QUE REALIZA LA CORRECCIÓN
2015-300-6-14617	21/12/2015	22/12/2015	TIPO: ESCRITURA	JOSE.GARCIA	JOSE.GARCIA

- Usuario: Es el nombre del calificador del cual se está realizando la consulta.
- Turno: Pertenece al turno de registro enviado a corrección de encabezado.
- Fecha de envío a corrección: Fecha en la que se envió el turno a corrección de encabezado.
- Fecha de Modificación: Fecha en la que se realizó la corrección.
- Datos Modificados: Pertenece a la información corregida de los campos tipo de documento, número de documento, fecha del documento, oficina de procedencia ya sea nacional o internacional, departamento, municipio, tipo de oficina y Número de Oficina, que deberá ser relacionada en esta columna.
- Usuario que solicita la corrección: Esta columna hace referencia al nombre del calificador que envió el turno de registro a corrección de encabezado.
- Usuario que realiza la corrección: Corresponde al usuario que se le asignó el turno de registro y realizó la corrección de encabezado.

Así mismo el reporte debe contener una sección llamada Resumen General con los siguientes ítems:

RESUMEN GENERAL	
TOTAL TURNOS ENVIADOS A CORRECCIÓN:	1
TOTAL TURNOS MODIFICADOS:	1

- Total turnos enviados a corrección: Es el total de turnos de registro enviados a corrección de encabezado.
- Total turnos modificados: Es el total de turnos de registro enviados a corrección de encabezado modificados.

**1478.AJUSTE TRASLADO SIR - FOLIO**

Actualmente en el aplicativo SIR cuando se realiza el traslado de matrículas de SIR a FOLIO, al efectuar la consulta del folio se observa que la información no es consistente.

Se requiere que al efectuarse el proceso de traslado de folios ya sea masivo o individual del aplicativo SIR al sistema FOLIO, estas matriculas no se listan para confirmación del traslado.



Así mismo al realizarse la consulta por ver detalles del folio de las matriculas trasladadas la información del campo estado del folio se debe relacionar la Orip destino a la cual se transfirió el folio.



Igualmente se debe ajustar la sección información relevante del traslado del folio mostrando los datos consistentes en los siguientes campos;

- Círculo Origen
- Numero matricula Origen
- Información del fundamento:
  - Círculo (Origen/Destino)
  - Tipo de fundamento
  - Numero de fundamento
  - Fecha del fundamento

Información Relevante de Traslado Folio			
Círculo Origen:	Numero Matricula Origen:	Círculo Destino:	
BOGOTA ZONA SUR	505-649753		
Círculo	Tipo de Fundamento	Numero de Fundamento	Fecha
ORIGEN	RESOLUCION	1	08/05/2015
DESTINO	RESOLUCION	1	17/05/2015

Realizando pruebas en el ambiente de producción para realizar la especificación del requerimiento se tomó del círculo de pruebas el folio 999-7475 el cual se trasladó a la oficina Bogotá Norte

The screenshot shows a MySQL query editor window. The top bar has various icons. Below it, a tab labeled "Hoja de Trabajo" contains the SQL query: "select \* from sir\_ne\_traslado where id\_matricula\_origen = '999-7475';". The results are displayed in a table titled "Resultado de la Consulta" with one row:

ID_TRASLADO	ID_MATRICULA_ORIGEN	ID_MATRICULA_DESTINO	FECHA_TRASLADO	RESOLUCION_ORIGEN
1 461	999-7475	FOLIO50N	02/06/2011 16:14:50 (null)	

Se procedió a realizar la consulta por ver detalles de folio de la matrícula 999-7475.

The screenshot shows the "Sistema de Información Registral - Producción - Versión 4.2.1" interface. The title bar includes the logo of the Superintendencia de Notariado y Registro, the Ministry of Interior and Justice of Colombia, and the motto "Libertad y Orden". Below the title bar, the date "Fecha Actual: 28/9/2015" is displayed. The main content area is titled "Pantalla Administrativa" and "Consulta de Detalles de Folios". It features a search form with a field for "Número del folio" containing "999-7475", a "Buscar" button, and a "Volver" button.

Generando excepción en la consulta de la información, por lo tanto este escenario tambien se debe tener en cuenta para su respectivo ajuste.

The screenshot shows the same administrative interface as the previous one, but with an error message displayed above the search form: "Error en la Validación de Datos del Formulario". The message lists a single error: "Los Siguientes Errores han sido encontrados en el Formulario: null". The search form below remains the same, with the number "999-7475" entered in the "Número del folio" field.

Teniendo en cuenta lo descrito anteriormente, el alcance del ajuste funcional se limita a las siguientes opciones de las pantallas administrativas del aplicativo SIR;

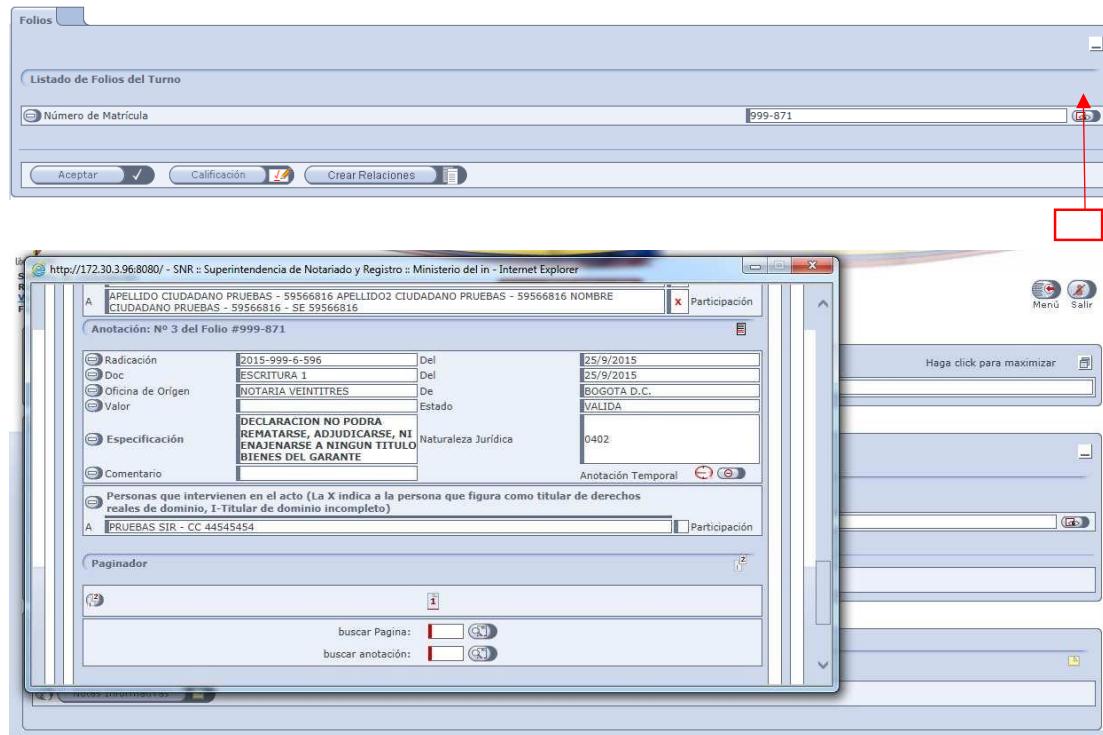
- Ver detalles de folio
- Administración de Traslados de Folios
- Administración de Traslados de Folios Masivo

## 1480.REGISTRADOR.VERIFICAR.INFORMACION.TEMPORAL.CALIFICACION

En el aplicativo SIR cuando un turno de registro se encuentra en la fase de firmar registro el registrador no puede visualizar la información temporal relacionada al folio o testamento.

Teniendo en cuenta la introducción se requiere ajustar la fase de firmar registro para que el registrador pueda visualizar la información temporal realizada al folio y anotaciones, al dar clic en maximizar en la sección folios actualmente se lista el número de folio asociado al turno, cuando se da clic en el botón de vista previa, se abre una ventana emergente donde mostrará la información definitiva y temporal del folio y anotaciones.

Vista Previa



Así mismo los turnos de registro de testamento, al dar clic en maximizar después de la sección folios, mostrará la sección testadores con la información nombre, apellido, tipo y número de identificación del testador, seguida con la sección registro de testamentos que relacionará la información del tomo, número de anotaciones, número de copias, revoca escritura y observación en campos no editables, como se muestra a continuación:

En la fase de firmar registro cuando se requiera devolver el turno a una fase ya sea de testamento con el botón “Devolver Testamento” o registro con el botón “Calificación”, el sistema debe validar que se haya incluido la nota informativa, de lo contrario mostrará el mensaje “No es posible devolver el turno a la fase. Se requiere una nota informativa.”



## DESPLIEGUE 5.1.2

27/05/2016

### 1931.ACTUALIZACION.LOGO.SNR

Se ha identificado que los imprimibles de los Certificados de Tradición y Libertad que se expiden en el aplicativo SIR y en las Oficinas de Registro (por antiguo CTL) están saliendo con el logotipo de la SNR antiguo.

Se requiere realizar la actualización del Logo de la SNR en los imprimibles que genera el aplicativo SIR, teniendo en cuenta las definiciones de las imágenes que se relacionan a continuación:

Logo superior:

Imagen - Logo superior	
Nombre: icono3.gif	
Tamaño 91 x 38 px	
Formato: GIF	
Peso: 3 kb	

Logo Marca de agua:

Imagen - Marca de agua
------------------------

Nombre: waterMark.gif	
Tamaño 480 x 328 px	
Formato: GIF	
Peso: 7 kb	
	

Los logotipos se visualizarán en los imprimibles que son expedidos por los siguientes procesos y opciones del aplicativo SIR:

Proceso / Opción	Imprimible	Tipo de Logo
Certificados Individuales	Certificado de tradición	Superior – Marca de agua
Certificados Masivos	Certificado de tradición	Superior – Marca de agua
Consultas	Solicitud Consulta	Superior
Correcciones	Formulario de Corrección (Temporal y definitivo)	Superior – Marca de agua
	Certificado asociado	
Devoluciones	Recibo de Pago Devoluciones	Superior
Registro	Formulario de Calificación (temporal y definitivo)	Superior – Marca de Agua
	Certificado asociado	
	Reproducción de constancia de inscripción	
	Constancia de testamento	
Reparto Notarial	Caratula de Reparto Notarial	Superior
	Radicador de Minutas	Superior
	Acta de Reparto Notarial	Superior
Restitución Reparto Notarial	Respuesta de Restitución Reparto Notarial	Superior
Antiguo Sistema	Hoja de Ruta	Superior
Notas devolutivas	Notas devolutivas	Superior – Marca de Agua
Notas informativas	Notas informativas	Superior
Reimpresión Formulario de Calificación	Formulario de Calificación	Superior – Marca de Agua
Reimpresión de Consultas	Solicitud Consulta	Superior

Reimpresión de Caratulas Reparto Notarial	Caratula de Reparto Notarial	Superior
Reimpresión de Actas Reparto Notarial	Acta de Reparto Notarial	Superior
Reimpresión Formulario de Correcciones	Formulario de Corrección	Superior – Marca de Agua
Reimpresión constancia de testamento	Constancia de testamento	Superior

Así mismo se efectuará el cambio de los logos anteriormente mencionados en los certificados de tradición que se generen a través del aplicativo de CTL.

### DESPLIEGUE 5.1.1

29/04/2016

#### 1159.111.LIBERACION.FOLIOS.TURNOS.FINALIZADOS

Actualmente el aplicativo SIR, no libera el bloqueo de algunas matrículas asociadas a turnos de Registro y Correcciones, que ya se encuentran avanzados a las fases Desanotado, Entrega y con estado Finalizado.

Cuando un turno de registro y/o correcciones avanza a las fases Desanotado, Entrega y con estado Finalizado, el sistema debe desbloquear los folios que se encuentren asociados al turno, sin tener en cuenta para los turnos de registro la resolución: Inscrito, Inscrito Parcialmente o Devuelto y para los turnos de correcciones si fue aprobada o negada.

Basado en esto se deben tener en cuenta que los folios asociados a turnos de registro y corrección una vez avancen de los siguientes roles y fases no deben presentar bloqueos:

PROCESO	ROL	FASE
Registro	Registrador	Firmar Registro
Correcciones	Responsable Correcciones	Revisar Aprobar

Para la puesta en producción de este desarrollo, ETB debe establecer cuáles folios asociados a los turnos de los procesos de correcciones y registro que se encuentran en los roles y fases, que se relacionan en la siguiente tabla, se encuentren bloqueados:

PROCESO	ROL	FASE
Registro	Entrega por Correspondencia	Desanotado Correspondencia
		Entrega Externa
	Entrega Documentos	Desanotado
		Entrega
Correcciones	Entrega Correcciones	Corrección Desanotada
		Entrega Corrección

Para estos casos se hace necesario aplicar un procedimiento en base de datos que corrija el bloqueo de estos folios.

#### 1242.VER.DETALLES. TURNO.VISUALIZACION.FORMAS.PAGO

En el aplicativo SIR, al consultar un turno de cualquier proceso registral, el sistema debe permitir

visualizar la información de la forma de pago aplicada para el recaudo de los derechos correspondientes.

Se requiere que al realizar la consulta de un turno por Pantallas Administrativas en la opción Ver Detalles de Turnos de la sección Manejo de Turnos, para cualquier proceso y subproceso registral que implique recaudo de dinero tales como: Certificados, Certificados masivos, Consultas, Fotocopias, Registro y Mayor Valor, se visualice la información completa de todas las formas de pago aplicadas correspondientes a los derechos asociados al turno, que se discrimina en la sección Pagos Asociados.

La información que se visualizará en “Ver Detalles del Turnos” para las formas de pago existentes, es la que a continuación se encuentra marcada con “X”, que corresponden a los campos obligatorios al momento de agregar el pago en el aplicativo SIR y que actualmente no se ven en la consulta del turno:

Forma de Pago	Nombre de Banco	Nro. Doc. De Pago	Nro. de Aprobación	Fecha Documento	Valor Documento	Valor Pagado
Efectivo						X
Consignación	X	X		X	X	X
Cheque	X	X		X	X	X
Convenio						X
PSE	X		X	X		X
T. Débito	X	X	X	X		X
T. Crédito	X	X	X	X		X

Dependiendo la forma de pago aplicado para cada turno se visualizará la información de la siguiente manera:

Forma de Pago: Efectivo	Valor Pagado: 4,800
Forma de Pago: CONVENIO	Valor Pagado: 4,800
Forma de Pago: PAGO ELECTRONICO PSE	Banco/Franquicia: BANCO CAJA SOCIAL BCSC - AMERICAN EXPRESS Número Aprobación: 496849570344 Fecha Documento: 08/09/2015 Valor Pagado: 1,600.00
Forma de Pago: TARJETA DEBITO	Banco/Franquicia: BANCAMIA - MASTER CARD Número Documento Pago: 5675768678709809 Número Aprobacion: 75670089780007 Fecha Documento: 4/09/2015 Valor Pagado: 970.00
Forma de Pago: TARJETA DEBITO	Banco/Franquicia: BANCO CAJA SOCIAL BCSC - DEBITO MAESTRO Número Documento Pago: 6557000899087908 Número Aprobacion: 6558000078900 Fecha Documento: 10/09/2015 Valor Pagado: 180.00
Forma de Pago: TARJETA CREDITO	Banco/Franquicia: BANCO FINANDINA S.A. - AMERICAN EXPRESS Número Documento Pago: 978890567464565 Número Aprobacion: 5586790000 Fecha Documento: 4/09/2015 Valor Pagado: 2,300.00
Forma de Pago: TARJETA CREDITO	Banco/Franquicia: BANCO CAJA SOCIAL BCSC - MASTER CARD Número Documento Pago: 4575687089089560 Número Aprobacion: 3534645600078 Fecha Documento: 3/09/2015 Valor Pagado: 890.00
Forma de Pago: CHEQUE	Banco/Franquicia: BANCO CAJA SOCIAL BCSC Número Documento Pago: 893453000282351 Número Aprobacion: 28,000.00 Fecha Documento: 1/09/2015 Valor Pagado: 8,000.00
Forma de Pago: CHEQUE	Banco/Franquicia: CAV GRANAHORRAR Número Documento Pago: 7568007897898903 Número Aprobacion: 12,000.00 Fecha Documento: 2/09/2015 Valor Pagado: 2,000.00
Forma de Pago: CONSIGNACION	Banco/Franquicia: HELM BANK (ANTES BANCO DE CREDITO) Número Documento Pago: 0000097213712312 Número Aprobacion: 678,000.00 Fecha Documento: 1/09/2015 Valor Pagado: 6,780.00
Forma de Pago: CONSIGNACION	Banco/Franquicia: CITY BANK COLOMBIA Número Documento Pago: 0000897540871230 Número Aprobacion: 50,000.00 Fecha Documento: 2/09/2015 Valor Pagado: 5,000.00

Los demás campos que se visualizan actualmente en la sección Pagos Asociados y cuando existe un Mayor Valor, se mantendrán e irán antes de la relación de las formas de pago asociadas al turno que se consulta. (Ver Imagen).

- ✓ Para turnos de Certificados, Consultas, Fotocopias y Registro:

Pagos asociados			
Valor derechos de registro:	25,000	Valor certificados asociados:	0
Valor de impuestos:	0	Valor de multa:	0
Otro impuesto:		Valor otro impuesto:	0
Forma de Pago:	Efectivo	Valor Pagado:	25,000

- ✓ Para turnos con Mayor Valor:

<b>Pagos asociados</b>			
Valor derechos de registro:	609,300	Valor certificados asociados:	0
Valor de impuestos:	0	Valor de multa:	0
Otro impuesto:		Valor otro impuesto:	0
Forma de Pago:	EFFECTIVO	Valor Pagado:	609,300
<b>Pago Mayor Valor</b>			
SE DEBE CANCELAR POR LA INSCRIPCION DE DERECHO DE PREFERENCIA Y LA PROHIBICION DE TRANSFERENCIA EL VALOR DE 16,500. POR C/U			
Valor derechos de registro:	33,000	Valor certificados asociados:	0
Valor de impuestos:	0	Valor pagado:	33,000

Realizando pruebas de desarrollo se evidencia el comportamiento en el aplicativo SIR en la opción ver detalles del turno, cuando se anulan turnos con formas de pago consignación y cheque, el sistema restablece el saldo del valor del documento y relaciona en el campo valor pagado la cantidad de 0.

Teniendo en cuenta el escenario anterior se debe ajustar el formulario incluyendo el campo pago anulado que relaciona el valor del documento aplicado al momento de efectuarse la anulación del turno, observándose como se muestra a continuación:

Forma de Pago:	Valor Pagado:	Número Aprobación:	Fecha Documento:	Valor Pagado:
EFFECTIVO	128,000	1242567665	10/07/2015	100,000
PAGO ELECTRONICO PSE	CAV COLMENA - AMERICAN EXPRESS		No. Aprobación:	Fecha Documi Valor Pagado:
Forma de Pago:	Banco/Franquicia:	No. Documento Pago:	No. Aprobación:	Fecha Documi Valor Pagado:
TARJETA DEBITO	OCCIDENTE - MASTER CARD	43532657	41326353	10/11/2015 200,000
Forma de Pago:	Banco/Franquicia:	No. Documento Pago:	Valor Documento:	Fecha Documi Valor Pagado:
CHEQUE	MEGABANCO	73445135	528,000	10/11/2015 0 100,000
Forma de Pago:	Banco/Franquicia:	No. Documento Pago:	Valor Documento:	Fecha Documi Valor Pagado:
CONSIGNACION	BANCO DE BOGOTA	4365265	528,000	10/11/2015 0 100,000

El cambio del campo “Pago anulado” y la información que se relaciona en este, aplicará para los turnos que se anulen después de desplegar la versión 5.1.1 del aplicativo SIR.

**NOTA:** El ajuste solicitado no aplica para la visualización del detalle de la forma de pago en los turnos de certificados individuales asociados a un turno de certificado masivos, ya que el sistema tiene definida la forma de pago “Pago Heredado” para los certificados individuales.

#### 1245.HABILITAR.TIPO.PREDIO

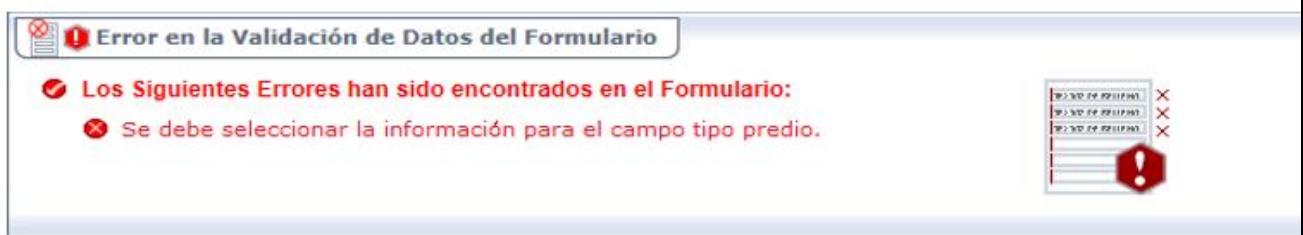
El sistema no permite actualizar o agregar en un folio el tipo de predio cuando se inscribe una anotación de actualización de predio, nomenclatura, área, linderos entre otros.

Desde el proceso de Registro en la fase de calificación cuando se efectúe inscripción de anotaciones que contengan ciertos actos de naturaleza jurídica, se requiere que el sistema habilite el campo Tipo del Predio, permitiendo agregar o actualizar la información en este campo.

En la fase calificación cuando se cree una anotación y se encuentre en estado temporal que contenga ciertas naturalezas jurídicas (ver listado), al dar clic en el botón “Regresar” de la interfaz crear anotaciones, el sistema volverá a la pantalla Folio, donde se deberá habilitar el campo “Tipo del Predio”, permitiendo seleccionar alguna de las opciones que almacena el aplicativo SIR, ya sea Rural, Urbano o Sin Información, como se muestra a continuación:

<input type="radio"/> Tipo del Predio	-SELECCIONE UNA OPCION-
<input checked="" type="radio"/> I - CALLE 20 7-61 BARRIO SAN IGNACIO	RURAL
	SIN INFORMACION
	URBANO

En el caso de no escoger alguna de estas opciones, el sistema validará y mostrará el mensaje “Se debe seleccionar la información para el campo tipo predio”;



Los cambios realizados al tipo del predio de un folio, deben ser almacenados en el esquema de auditoria SIR en la tabla AUD\_SIR\_NE\_FOLIO en los campos ID\_TIPO\_PREDIO e ID\_TIPO\_PREDIO\_NEW.

## 1246.ASIGNACION.REPORTES.165.166.167.ROL.REGISTRADOR

Con el fin de que los registradores tengan acceso a reportes específicos que le permitan realizar seguimiento y tomar acciones para el control en el manejo de turnos de registro, radicados, repartidos, en trámite y pendientes por calificar en cada una de las respectivas Oficinas de Instrumentos Públicos.

Se requiere que los reportes 165, 166, 167 y E02, se asignen al rol registrador y administrador regional obteniéndose únicamente para el círculo desde el cual se genera.

De acuerdo con lo requerido, se asociarán a los roles Registrador y Administrador Regional los siguientes reportes:

Reporte 165: Documentos Radicados Mensualmente

Reporte 166: Documentos Repartidos Mensualmente

Reporte 167: Documentos Pendientes por Calificar por cada Abogado

Reporte E02: Estadísticas Generales

Una vez el usuario que tenga asignado el rol Registrador o Administrador Regional, ingrese por Pantallas Administrativas a la sección Reportes y seleccione la opción Listado de Reportes, visualizará y permitirá acceder a cada uno de estos reportes mencionados anteriormente.

Para los reportes 165, 166 y 167 en el parámetro Círculo Origen y para el reporte E02 en el parámetro Círculo registral, solo se listara la ORIP al cual pertenece el usuario que está realizando la consulta.

La pantalla de los reportes 165, 166 y 167, se visualizará de la siguiente manera:

La pantalla del Reporte E02: Estadísticas Generales, se visualizará de la siguiente manera:

### 1251.PROCESO. CORRECCIONES.ESTADO.TRASLADO

Con el fin de controlar y evitar, que los funcionarios de la ORIPS mediante el proceso de correcciones no puedan cambiar folios al estado trasladado, es necesario ajustar esta funcionalidad. Toda vez que en el sistema exista una opción específica destinada para realizar traslado de folios.

En el proceso de correcciones en la fase de ejecutar corrección simple, el sistema no debe permitir cambiar ningún folio a estado trasladado.

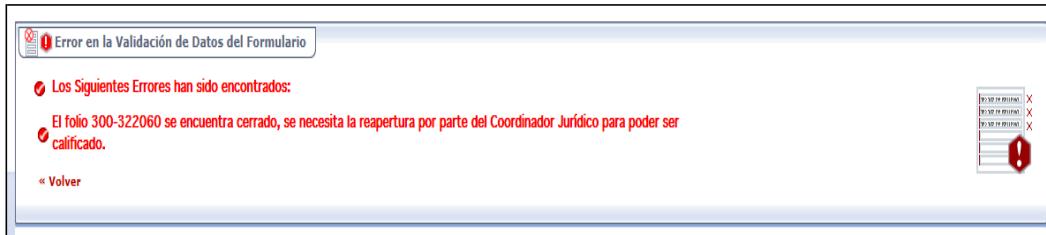
En el marco de este requerimiento se establece que, en el proceso de Correcciones desde la fase Ejecutar Corrección Simple del rol Auxiliar de Correcciones, el sistema no permitirá cambiar el campo Estado Folio, que se encuentra en la sección Datos Folio, al estado “Traslado”; para tal fin solo se visualizarán los estados Activo, Anulado, Cerrado y Custodia, que se desplegarán de la siguiente manera:

### 1253.CALIFICACION.FOLIOS.CERRADOS

Cuando un turno de registro en trámite tiene asociados folios en estado cerrado, en el proceso de calificación no debe permitir crear ningún tipo de anotación, estándar y/o cancelación, ni segregar, englobar, copiar anotaciones a folios en el estado mencionado.

Es necesario que en el proceso de Registro al realizar una calificación de una matrícula el sistema no debe permitir realizar ninguna de las opciones existentes en la interfaz Folio, tales como: Crear (anotación), Crear Cancelación, Copiar Anotación, Segregar y Englobar a folios asociados en estado Cerrado.

Al intentar realizar cualquiera de las acciones anteriormente mencionadas, el sistema mostrará el siguiente mensaje restrictivo:



## DESPLIEGUE 5.1.0

01/04/2016

### 1214.AJUSTE.REPORTES.FINANCIEROS.RECAUDO

En el marco de este requerimiento se establece que, se realizarán ajustes a los reportes financieros y de recaudo existentes en el aplicativo SIR, para que refleje en detalle la cantidad y dineros recaudados por concepto de certificados desde otros puntos de pago (PSE, Baloto, Supercades, CAE, CEP, entre otros).

Se crearán los siguientes usuarios en los reportes contemplados en este requerimiento, de acuerdo al medio de recaudo ya sea PORTAL-SNR o NODO.CENTRAL y conservando el comportamiento de cada reporte, salvo los ajustes aquí mencionados en cada uno de ellos:

DEFINICIÓN DE USUARIOS	
Usuarios PORTAL-SNR	Usuarios NODO.CENTRAL
PSE PORTAL	KIOSCOS
SUPERCADES	COLPATRIA
CAE	
CEP	
OTRAS ORIPS	

Los reportes a modificar son:

3. **Reporte 06:** Diario radicador de certificados del día: Conforme al comportamiento actual de este reporte se realizarán los siguientes ajustes:

- ✓ Se relacionarán en orden consecutivo los certificados radicados en el día: sin importar desde que punto o medio de pago fueron expedidos.
- ✓ En la columna Derechos: se mostrará la tarifa diferencial o tarifa plena del certificado (\$13.900 o \$14.800).
- ✓ En la columna Cajero: se colocará el nombre del usuario que se encuentra almacenado en el campo SRIO\_NOMBRE\_DE\_USUARIO de la tabla SIR\_OP\_USUARIO.
- ✓ En la columna Observaciones: se cambiará el texto no expedido por “No Impreso”
- ✓ El reporte estará compuesto por dos secciones:
  - Todos los turnos que radicó la oficina.
  - Todos los turnos que se radicaron para otras ORIPS SIR.
- ✓ El campo “TOTALES” se cambiará por: “DETALLE DEL RECAUDO”
- ✓ En la sección donde se relaciona el resumen por Tipo de Pago:
  - En el campo “Total Consignación:” se relacionarán los pagos recaudados por este medio y no tendrá en cuenta los pagos heredados de PSE Portal.
  - El campo “Total Otros Pagos:” se eliminará de esta sección y los pagos asociados a este ítem irán relacionados a la forma del medio de pago correspondiente. El tipo de pago se tomará del turno padre, ya que estos otros pagos hacen relación a los pagos heredados de los certificados masivos y certificados asociados.
  - El campo “Total Oficina Vr:” será cambiado por: “Total Recaudo ORIP”, que corresponde a la sumatoria del valor recaudado por los usuarios de la ORIP por concepto de pagos con Efectivo, Consignación y Cheque.
  - El campo “Total Otras ORIPS Vr:” será cambiado por: “Total Para Otras ORIPS SIR”. Sin embargo se determinó conjuntamente con la SNR en reunión del 17 de febrero del presente año, que se eliminará, ya que sus pagos se encuentran relacionados en las formas de pago listadas en la sección “DETALLE DEL RECAUDO”.
  - Se agregarán los totales de los siguientes medios de recaudo, debajo del campo “Total Recaudo ORIP” que relacionarán los valores pagados por sus conceptos tal como se muestra a continuación:
    - Total PSE Portal:
    - Total Kioscos:
    - Total Baloto Colpatria:
    - Total Supercades:
    - Total CAE:
    - Total CEP:
    - Total para otras ORIPS Portal:
    - Total por otras ORIPS Portal:
  - Al finalizar los totales mencionados anteriormente se realizará una sumatoria de estos

medios de recaudo, cuyo campo será identificado con el nombre “Total otros medios de recaudo:”

- Finalmente irá el campo “Total” que corresponde a la sumatoria de los ítems Total recaudo ORIP, Total Convenio, Total Pago Electrónico PSE, Total Tarjeta Crédito, Total Tarjeta Débito y Total otros medios de recaudo.

DETALLE DEL RECAUDO	
TIPO PAGO	TOTAL
Total Efectivo:	47,189
Total Consignación:	39,191
Total Cheque:	20,146
Total Convenio:	44,080
Total Pago Electrónico PSE:	1,090
Total Tarjeta Crédito:	23,909
Total Tarjeta Débito:	10,395
<b>Total Recaudo ORIP:</b>	<b>106,526</b>
Total PSE Portal:	0
Total Kioscos:	0
Total Baloto Colpatria:	0
Total Supercades:	0
Total CAE:	0
Total CEP:	0
Total para otras ORIPS Portal:	0
Total por otras ORIPS Portal:	0
<b>Total Otros medios de Recaudo:</b>	<b>0</b>
<b>Total:</b>	<b>289,600</b>

- ✓ En la sección donde se relaciona el resumen por cantidad de certificados:

- El campo “Total certificados otras ORIPS:” será cambiado por: “Total certificados para otras ORIPS SIR:”
- Se agregarán los usuarios PSE Portal, Kioscos, Colpatria, Supercades, CAE, CEP y Otras ORIPS para expedir certificados, los cuales también estarán relacionados debajo del campo “Total certificados para otras ORIPS SIR:”:
  - Total certificados por otras ORIPS SIR
  - Total certificados expedidos PSE Portal:
  - Total certificados expedidos Kioscos:
  - Total certificados expedidos Baloto Colpatria:
  - Total certificados expedidos Supercades:
  - Total certificados expedidos CAE:
  - Total certificados expedidos CEP:
  - Total certificados para otras ORIPS Portal: (Certificados expedidos por botón de pago para otras oficinas desde una ORIP determinada).
  - Total certificados por otras ORIPS Portal: (Certificados expedidos por botón de pago

desde otras oficinas para una ORIP determinada).

Se aclara en reunión conjuntamente con la SNR el 17 de Febrero del año en curso que para los ítems Total certificados para otras ORIPS Portal y Total certificados por otras ORIPS Portal, el recaudo será efectuado por los usuarios que se crearán para la nueva bancarización por parte de la SNR.

- Debajo del campo “Total certificados por otras ORIPS Portal:” y antes del campo “Total certificados locales:”, se agregarán los siguientes campos:

- Total certificados tarifa Normal
- Total certificados tarifa Media
- Total certificados tarifa Especial
- Total certificados tarifa Exenta

Total certificados expedidos:	31
Total certificados reimpresos:	2
Total certificados anulados:	2
Total certificados para otras ORIPS SIR:	3
Total certificados por otras ORIPS SIR:	1
Total certificados expedidos PSE Portal:	0
Total certificados expedidos Kioscos:	0
Total certificados expedidos Baloto Colpatria:	0
Total certificados expedidos Supercades:	0
Total certificados expedidos CAE:	0
Total certificados expedidos CEP:	0
Total certificados para otras ORIPS Portal:	0
Total certificados por otras ORIPS Portal:	0
Total certificados tarifa Normal:	13
Total certificados tarifa Media:	4
Total certificados tarifa Especial:	0
Total certificados tarifa Exenta:	12
Total certificados locales:	29
Derechos certificados locales:	\$ 260,000
Derechos certificados para otras ORIPS:	\$ 29,600
Total derechos ajustados a la centena:	\$ 289,600

#### 4. Reporte 25: Informe recaudos del día (tiracaja):

- ✓ El sistema debe permitir generar este informe para otros usuarios diferentes a los cajeros y liquidadores de la ORIP, como son los que realizan las transacciones de PSE, Baloto, Supercades, entre otros. Para estos usuarios, se deben aplicar las mismas validaciones y datos que para los usuarios de la ORIP, como es reflejar el valor de recaudo y el total consolidado por tipo de pago, para tal fin se agregarán a la lista de usuarios por Círculo Registral, de la Pantalla Administrativa – Reportes - Listado de Reportes - Reporte 25: Informe recaudos del día (tiracaja): los usuarios de Nodo Central y Portal SNR relacionados en el cuadro Definición de Usuarios.
- ✓ El campo “TOTALES” se cambiará por: “DETALLE DEL RECAUDO”.

- ✓ En la sección TOTALES el campo “Total Oficina Vr:” será cambiado por: “Total Recaudo ORIP”, que corresponde a la sumatoria del valor recaudado del usuario de la ORIP por concepto de pagos con Efectivo, Consignación y Cheque.
- ✓ Para los usuarios de Nodo Central y Portal SNR se relacionarán los mismos ítems, pero traerá por defecto el valor de “0” (cero) para aquellos que no tengan ningún pago asociado.
- ✓ El campo “Total Otras ORIPS Vr:” será cambiado por: “Total Para Otras ORIPS SIR”. El cual se determinó conjuntamente con la SNR en reunión del 17 de febrero del presente año que se eliminará, ya que sus pagos se encuentran relacionados en las formas de pago listadas en la sección “DETALLE DEL RECAUDO”.
- ✓ El campo “Certificados Otras ORIPS” de la sección que relaciona los certificados nacionales, se cambiará por: “Certificados para otras ORIPS”.
- ✓ Realizadas las pruebas de calidad de la ORIP de Pereira del 11 al 16 de Febrero del 2016, se generó este reporte para los usuarios MONICA.ROBLEDO, MARIA.MATURANA y OSCAR.VELASQUEZ, donde se verificaron los turnos radicados del proceso de Devoluciones evidenciándose las siguientes inconsistencias:
  - La columna Forma de Pago relaciona como Efectivo todos los conceptos de la devolución asociados en la radicación del turno de devoluciones, tales como: Turnos, Cheque o Consignación.
  - Cuando un turno tiene asociado la forma de pago nota débito, relaciona en los campos Vr.Docu.Pago, Vr. Aplicado y Diferencia como valor, la cantidad de cero (0), se requiere detallar el valor pagado por concepto de la devolución.
  - La relación de los turnos del proceso de Devoluciones se refleja en el usuario que radica el turno y en el usuario que registra el pago de la devolución. En la sección donde se detallan los turnos radicados para el usuario que realiza la creación del turno del proceso de Devoluciones no se listará ningún turno, con el fin de que en la sección “DETALLES DEL RECAUDO” no se sume ningún valor por este concepto y se afecte el total recaudado; para el usuario que realiza el pago de la devolución se mostrará la relación del turno radicado relacionando la forma de pago, así mismo se visualizará en la sección turnos de devolución.
- ✓ En la sección “Certificados para otras ORIPS” cuando se asocia más de una forma de pago a un certificado nacional, el sistema duplica la cantidad y el valor del certificado, por ejemplo:

<b>Certificados para otras ORIPS</b>		
Círculo	Cantidad	Valor
BUENAVENTURA (372)	1	\$ 14,800
ARMENIA (280)	2	\$ 29,600
Total certificados para otras Orips		\$ 44,400

Solo se generó un certificado para la ORIP de Armenia con el turno 2016-280-1-6 y el pago tiene asociadas las formas de pago Efectivo por \$4.800 y Consignación del Banco Tequendama por \$10.000:

	<p><b>OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS</b>  <b>NIT: 899999007-0 PEREIRA</b>  <b>SOLICITUD CERTIFICADO DE TRADICIÓN</b>  Impreso el 11 de Febrero de 2016 a las 10:57:45 am</p> <p style="text-align: right;"><b>44374916</b></p> <p>No. RADICACIÓN: <b>2016-280-1-6</b></p> <p>TIPO DE CERTIFICADO: NACIONAL  MATRÍCULA: <b>280-2345</b>  CERTIFICADO(S) SE EXPIDE(N) DE INMEDIATO  NOMBRE DEL SOLICITANTE: BULLA MARIA ELENA  CERTIFICADOS: 1</p> <p><b>FORMA DE PAGO:</b></p> <p>CONSIG. BANCO: 29 Nro: 123545 FECHA CONSIG.: 10/02/2016 VALOR: \$ 10.000  EFECTIVO VALOR: \$ 4.800</p> <p>VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: <b>\$ 14.800</b>  EL CERTIFICADO SE EXPIDE DE ACUERDO A LOS DATOS SUMINISTRADOS  Usuario: 53107</p>	
--	---	--

Se corregirá este comportamiento con el fin de que muestre la cantidad y el valor del certificado radicado y recaudado.

**5. Reporte 51:** Resumen diario de ingresos y egresos: de acuerdo al comportamiento actual de este reporte se realizarán los siguientes ajustes:

- ✓ En la sección donde se detallan los Conceptos por usuario, de cada uno de los usuarios de la ORIP, se deberán incluir los usuarios correspondientes a PORTAL-SNR y NODO.CENTRAL relacionados en el cuadro Definición de Usuarios, siempre y cuando se haya recaudado algún valor por sus conceptos y se mantendrán las mismas columnas definidas actualmente en el reporte.
- ✓ La sección RESUMEN GENERAL POR CONCEPTOS se cambiará el título por: “RESUMEN GENERAL POR CONCEPTOS DE RECAUDO DE LA OFICINA”.
- ✓ En la sección del RESUMEN GENERAL POR CONCEPTOS DE RECAUDO DE LA OFICINA, se incluirá el campo “Total Recaudo ORIP”, que corresponde a la sumatoria total de los ítems que relacionan las formas de pago: Efectivo, Consignación y Cheque, ubicándose después del campo “Total General”, y solo aplicará para los usuarios de la ORIP.
- ✓ Debajo de la sección RESUMEN GENERAL POR CONCEPTOS DE LA OFICINA se incluirá el RESUMEN GENERAL POR CONCEPTOS DE OTROS MEDIOS DE RECAUDO que tendrá el mismo diseño de la sección anterior; las columnas que no apliquen valores de acuerdo a los conceptos definidos en el requerimiento traerán por defecto el valor de “0” (cero), así mismo el Ingeniero Guillermo Cabrera aclara en la reunión realizada conjuntamente con la SNR el 17 de febrero del año en curso, que los valores que se reflejarán en el reporte, serán las cantidades que se registran actualmente en el esquema de la base de datos del aplicativo SIR.
- ✓ En la sección CERTIFICADOS OTRAS ORIPS, se cambiará el título por “CERTIFICADOS PARA OTRAS ORIPS SIR”, este ítem hace referencia a los certificados Nacionales.

- ✓ Realizadas las pruebas de calidad de la ORIP de Pereira del 11 al 16 de Febrero del 2016, se generó este reporte, donde se verificaron los turnos radicados del proceso de Devoluciones evidenciándose las siguientes inconsistencias:
  - La columna Total Aplicado duplica el valor de la columna Efectivo en la sección de Conceptos por usuario.
  - El valor por concepto de devoluciones no se está asociando correctamente en las columnas donde se discriminan las formas de pago con el cual se efectuó la devolución.
  - El detalle asociado al Concepto Devoluciones se refleja en el usuario que radica el turno y no en el usuario que registra el pago de la devolución.
- ✓ Se corregirán estas inconsistencias en la sección de Conceptos por usuario, en el concepto que hace referencia a los turnos del proceso de Devoluciones con sus respectivas formas de pago.
- ✓ En la sección “Resumen General por Bancos” se deberán incluir los pagos cuando se realice un mayor valor generado del proceso de correcciones.
- ✓ En la sección “RESUMEN GENERAL POR BANCOS DEVOLUCIONES” se observará la información correspondiente al Banco, Valor Aplicado y Valor de la Devolución por cada banco que haya sido asociado al registrar el pago de los turnos del proceso de Devolución, ya que actualmente no se visualiza esta información.

**6. Reporte E02: Estadísticas Generales:** Conforme al comportamiento actual de este reporte se realizarán los siguientes ajustes:

- ✓ Certificados Inmediatos Nacionales: se cambiará el nombre por “CERTIFICADOS EXPEDIDOS PARA OTRAS ORIPS SIR”.
- ✓ En el listado donde se discrimina el proceso CERTIFICADOS, se relacionarán los siguientes ítems, trayendo por defecto el valor de “0” (cero) de aquellos que no tengan ningún registro asociado por estos conceptos:
  - CERTIFICADOS EXPEDIDOS PSE PORTAL
  - CERTIFICADOS EXPEDIDOS KIOSCOS
  - CERTIFICADOS EXPEDIDOS BALOTO - COLPATRIA
  - CERTIFICADOS EXPEDIDOS SUPERCADE
  - CERTIFICADOS EXPEDIDOS CAE
  - CERTIFICADOS EXPEDIDOS CEP
  - CERTIFICADOS EXPEDIDOS PARA OTRAS ORIPS PORTAL (Certificados expedidos por botón de pago para otras oficinas desde una ORIP determinada).
  - CERTIFICADOS EXPEDIDOS POR OTRAS ORIPS PORTAL: (Certificados expedidos por botón de pago desde otras oficinas para una ORIP determinada).
  - CERTIFICADOS EXPEDIDOS POR OTRAS ORIPS SIR
  - CERTIFICADOS EXPEDIDOS PARA OTRAS ORIPS SIR (Certificados Nacionales en el aplicativo SIR).

- ✓ Al finalizar cada proceso registral que recauda dinero en el aplicativo SIR, se incluirá el campo Subtotal haciendo referencia a la sumatoria de cada uno de los procesos: Certificados, Consultas, Devoluciones de Dinero, Documentos y Fotocopias.
- ✓ En la sección Documentos: se debe agregar el ítem “DOCUMENTOS POR RADICACION ELECTRONICA”, aquí se debe tener en cuenta la cantidad y valor de los turnos generados por el Web Service de radicación electrónica.
- ✓ En la sección de TOTALES se realizarán las siguientes modificaciones:
  - El campo “TOTALES” se cambiará por: “DETALLE DEL RECAUDO”
  - En el campo “Total Consignación.” se relacionarán los pagos recaudados por este medio y no tendrá en cuenta los pagos heredados de PSE Portal, ya que estos recaudos se contemplarán en el ítem Total PSE Portal.
  - El campo “Total Otros Pagos.” se eliminará de esta sección y los pagos asociados a este ítem irán relacionados a la forma del medio de pago correspondiente. El tipo de pago se tomará del turno padre, ya que estos otros pagos hacen relación a los pagos heredados de los masivos y certificados asociados.
  - El campo “Total.” se cambiará por “Total Recaudo ORIP”, que corresponde a la sumatoria del valor recaudado por los usuarios de la ORIP por concepto de pagos con Efectivo, Consignación y Cheque.
  - El campo “Total Oficina Vr.” se cambiará por “Total”, que corresponde a la sumatoria de todos los tipos de pago, relacionados en la columna “TIPO PAGO”.
  - El campo “Total Otras ORIPS Vr.” se quitará de esta sección ya que su valor se encuentra relacionado en el Proceso Certificados mediante el campo “CERTIFICADOS EXPEDIDOS PARA OTRAS ORIPS SIR”.
  - Se agregarán los totales de los siguientes medios de recaudo, debajo del campo “Total Recaudo ORIP”, que relacionará los valores pagados por cada uno de sus conceptos, tal como se muestra a continuación:
    - Total PSE Portal:
    - Total Kioscos:
    - Total Baloto Colpatria:
    - Total Supercades:
    - Total CAE:
    - Total CEP:
    - Total para otras ORIPS Portal:
    - Total por otras ORIPS Portal:
  - Al finalizar los totales mencionados anteriormente se realizará una sumatoria de estos medios de recaudo, cuyo campo será identificado con el nombre “Total otros medios de recaudo.”

- Finalmente irá el campo “Total” que corresponde a la sumatoria de los ítems Total recaudo ORIP, Total Convenio, Total Pago Electrónico PSE, Total Tarjeta Crédito, Total Tarjeta Débito y Total otros medios de recaudo.

DETALLE DEL RECAUDO	
TIPO PAGO	TOTAL
Total Efectivo:	47,189
Total Consignación:	39,191
Total Cheque:	20,146
Total Convenio:	44,080
Total Pago Electrónico PSE:	1,090
Total Tarjeta Crédito:	23,909
Total Tarjeta Débito:	10,395
<b>Total Recaudo ORIP:</b>	<b>106,526</b>
Total PSE Portal:	0
Total Kioscos:	0
Total Baloto Colpatria:	0
Total Supercades:	0
Total CAE:	0
Total CEP:	0
Total para otras ORIPS Portal:	0
Total por otras ORIPS Portal:	0
<b>Total Otros medios de Recaudo:</b>	<b>0</b>
<b>Total:</b>	<b>289,600</b>

- ✓ Los títulos de los procesos CERTIFICADOS, CONSULTAS, DEVOLUCIONES, DOCUMENTOS y FOTOCOPIAS, irán en negrilla con tipo de fuente Arial y tamaño 10.
- ✓ Realizadas las pruebas de calidad de la ORIP de Pereira del 11 al 16 de Febrero del 2016, se generó este reporte evidenciándose las siguientes inconsistencias que serán corregidas con este desarrollo:
  - El campo “CERTIFICADOS ANULADOS”, no está incluyendo los certificados nacionales anulados.
  - El campo “CERTIFICADOS DE CORRECCIONES”, realiza el conteo y el valor de turnos que no corresponden a certificados del proceso de correcciones.
  - El campo “DOCUMENTOS MAYOR VALOR” no trae ningún valor por concepto de los derechos de este subprocesso; de igual manera las formas de pago asociadas al pago del mayor valor, no se están relacionando en la sumatoria de los tipos de pago de la sección “DETALLE DEL RECAUDO”.
  - La sección BANCOS no está mostrando la relación de todos los bancos asociados por concepto de los pagos agregados en los procesos registrales contemplados para este reporte.
- ✓ En la fase de pruebas externas realizadas los días 10 y 11 de Marzo del 2016, los Ingenieros Pablo Vera, Rodrigo Pérez y la funcionaria Digna Isabel Quiroz, determinaron que se deben realizar en el reporte los ajustes que se relacionan a continuación:
  - El nombre del campo “DOCUMENTOS MAYOR VALOR” será cambiado por “DOCUMENTOS MAYOR VALOR PAGADOS” que contemplará solo los pagos

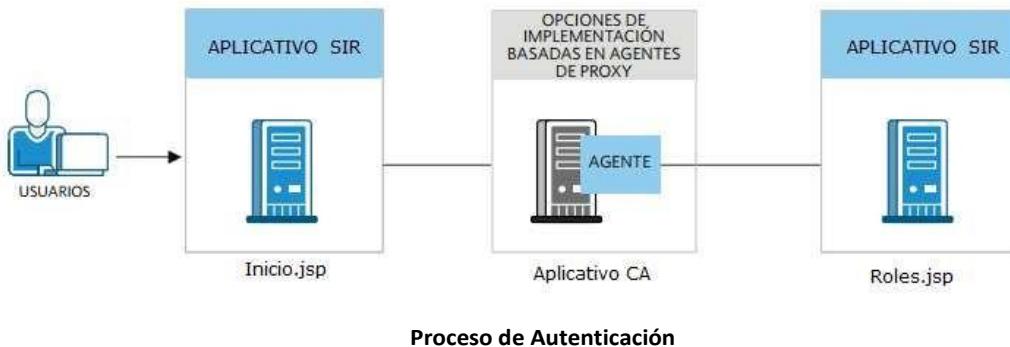
realizados por concepto de mayor valor del proceso de registro, dentro de la fecha consultada para este reporte.

- La sección Devoluciones de dinero para los ítems Devoluciones pagadas, Devoluciones Radicadas, Devoluciones de Dinero por Certificados y Devoluciones de Dinero por Documentos, el reporte mostrará en la columna valor de los derechos, el valor radicado o pagado por este proceso. Estos valores no se incluirán en las secciones DETALLE DEL RECAUDO, ni BANCOS, ya que estos valores no representan recaudo de dinero para las oficinas.

## DESPLIEGUE – 5.0.0 23/11/2015

### **1241. ADAPTACION DEL PROCESO DE AUTENTICACIÓN Y SSO DE CA SITEMINDER**

En el aplicativo SIR se adaptó el proceso de autenticación y SSO de CA SiteMinder, instalando un servidor web (iplanet) como proxy a Jboss donde se encuentra ubicado SIR, en el cual se instaló un agente de CA SiteMinder para interceptar la solicitud de acceso a SIR, solicitando las credenciales y una vez autenticado el usuario se generará vía HTTP Headers algunas variables incluyendo SM\_USER, capturando la información que contiene el nombre del usuario que fue autenticado, con esta información la aplicación debe obtener la variable header y procesar su lógica de autorización y negocio para dicho usuario.



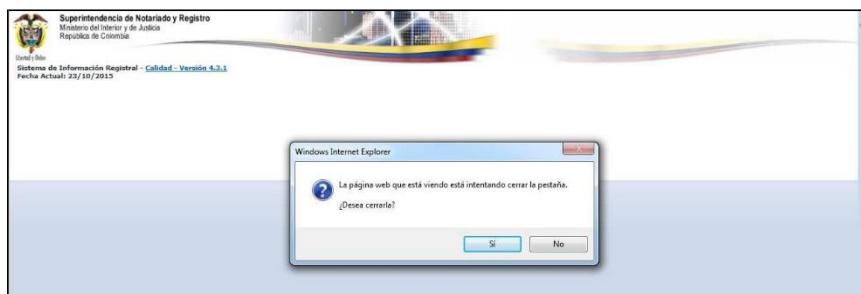
Teniendo en cuenta lo anterior se cambió la pantalla de acceso del aplicativo SIR, por la pantalla de autenticación de CA SiteMinder como se muestra a continuación;

Pantalla de Login Actual



Pantalla de Login CA SiteMinder

- Para la página de inicio se definirá en el documento de análisis y diseño del desarrollo, la URL del aplicativo SIR.
- Se eliminará el cuadro de dialogo de confirmación que se despliega después de haber ingresado la URL de acceso al aplicativo SIR, ya que se requiere para el nuevo diseño de login no exista una segunda opción de autenticación de acceso al sistema. A continuación se relaciona la imagen como se muestra actualmente de acuerdo al escenario descrito.



- Después de obtener la autenticación, el sistema mostrará la pantalla de ROLES, Ver imagen.



- Cuando el usuario haga logout del aplicativo SIR, invocará la URL de la autenticación de CA para salirse del aplicativo SIR.
- En el aplicativo SIR en la sección de pantallas administrativas, se eliminará la opción cambio de claves de usuarios ya que se reestablecerá por la parte de administración de CA.
- Este requerimiento tendrá como prerequisito definir la infraestructura que interactuará con el cambio de la autenticación y login a través de CA SiteMinder, para desplegar el desarrollo en ambiente de producción.
- El ajuste contemplado en este requerimiento no presentará conflictos con versiones de Internet Explorer.
- Se debe mantener la validación para usuarios de única sesión en el aplicativo SIR, ya que por pc debe haber activa una sesión.

### **DESPLIEGUE – 4.3.1**

**16/10/2015**

#### **1540\_111\_AJUSTE.IMPRIMIBLE.CTL.FOLIO**

Actualmente cuando se genera el certificado de tradición de matrículas del aplicativo FOLIO a través del sistema CTL, el imprimible no está listando las matrículas segregadas, la causa se debe a que el imprimible que se modificó para el aplicativo SIR es el mismo que maneja el sistema CTL para el aplicativo FOLIO sin tener conocimiento de esta condición por parte de ETB, incidiendo en el comportamiento de la construcción de estos certificados, los cuales no se contemplaron en las pruebas de integración que efectúa la SNR para el despliegue 4.3.0 del aplicativo SIR.

Teniendo en cuenta la problemática definida en la introducción se requiere realizar el ajuste para que los certificados de tradición que se generan en CTL de las matrículas del sistema FOLIO, se relacione la información de los folios segregados, sin afectar el diseño que se hizo para el requerimiento No 1155 Modificar imprimible de calificación y corrección, desplegado con la versión 4.3.0 del aplicativo SIR.

### **DESPLIEGUE – 4.3.0**

**02/10/2015**

#### **904\_111\_Actualizacion Fichas Catastrales**

El proceso de actualización masiva de números prediales, ya se ha venido aplicando en forma masiva en todas las Oficinas de Registro, incluyendo los sistemas de información FOLIO y SIR, para lo cual se aplican procedimientos (funciones, paquetes, entre otros) a nivel de Base de datos.

Teniendo en cuenta que el IGAC y los catastros descentralizados vienen entregando grandes volúmenes de números prediales, y en especial el IGAC, ha entregado 4.907.023 nuevos números prediales de 30 dígitos, de los cuales están pendientes por actualizar 2.850.727 correspondientes a Oficinas de Registro que operan con SIR; se requiere que se modifiquen los procedimientos ya definidos para que se realice la actualización masiva.

Durante la ejecución del procedimiento de actualización masiva se debe tener en cuenta las siguientes validaciones y restricciones;

1. Las validaciones de los turnos en trámite o con prioridad de turno no serán tenidas en cuenta.

2. Las matriculas relacionadas en el listado entregado por el Proyecto Interrelaciòn, deberán ser actualizadas en su totalidad, sin tener en cuenta su estado, excepto las que hayan sido objeto de traslado (referente a estas últimas, ETB debe generar un reporte el cual se debe enviar a la SNR).

3. Por cada matricula actualizada deberá incluirse una salvedad con las siguientes características:

- a. Id\_Usuario: ICARE, este usuario deberá ser creado por base de datos.
- b. Turno: ICARE –vigencia, la vigencia corresponde al año en el cual se realiza la actualización.
- c. Texto de la salvedad: “SE INCLUYE NUEVO NUMERO PREDIAL DE 30 DIGITOS SUMINISTRADOS POR EL I.G.A.C. (SNC), RES No. 8589 DE 27-11-2008 PROFERIDA POR LA S.N.R (CONVENIO IGAC-SNR DE 23-09-2008)”.

4. En las pantallas de consultas e imprimible del folio del aplicativo SIR, se debe observar la información del código catastral actualizado, permitiendo visualizar los 30 dígitos o más que compone este código, a continuación se relacionan los procesos e imprimibles que contemplan la visualización del dato;

<b>CONSULTAS DE FOLIO</b>	
<b>Proceso de Certificados</b> -Fase Ampliación Tradición -Fase Oficio de Pertenencia -Buscador Antiguo Sistema – Fase Hoja de Ruta -Encargado Antiguo Sistema – Revisión Final	<b>Proceso de Correcciones</b> -Fase Revisión y Análisis -Fase Ejecutar Corrección simple -Fase Revisar y aprobar -Fase Mesa Control Correcciones -Fase Ejecutar Actuaciones Administrativas -Encargado Antiguo Sistema – Fase Revisión Final
<b>Proceso Devolución</b> -Usuario Administrativo – Fase Análisis.	
<b>Proceso de Registro</b> -Fase Calificación (Información Folio – Inscripción) - Fase Digitador Masivo - Registrador – Fase Firma Registro -Coordinador Jurídico – Fase Aprobar Corrección Calificación	<b>Imprimibles</b> -Certificados de Tradición -Certificados Asociados (Corrección – Registro) -Formulario Calificación -Formulario Corrección -Vista preliminar del certificado en el proceso de consultas -Impresión simple de folio
<b>Pantallas Administrativas</b> -Ver detalle del turno -Ver detalles del folio	Con base a la actualización de fichas catastrales, en el proceso de consultas al realizar la búsqueda por el criterio de cedula catastral, el sistema debe reportar como resultado las matriculas que contengan esa información.  - Despues de ejecutarse el procedimiento de actualización masiva de las fichas catastrales, ETB deberá reportar;  a.Listado de matrículas con actualización satisfactoria. b.Listado de matrículas encontradas como inconsistentes y/o erradas informando el motivo por el cual no se realizó la actualización.
<b>1155_111_Modificar imprimible calificación corrección</b>	
En el aplicativo SIR para los procesos de registro y/o correcciones en los formularios de calificación, corrección y los	

imprimibles de los certificados de tradición, la anotación segregadora del folio matriz, en las matriculas segregadas debe reflejar el nombre del inmueble.

Las matriculas se deben listar en frente de la anotación segregadora, como se observa a continuación;

#### Formulario de Calificación y Corrección

ANOTACIÓN: Nro: 11	Fecha 11/3/2015	Radicación 2015-300-6-8871
DOC: ESCRITURA 370	DEL: 6/3/2015	NOTARIA SEXTA DE BUCARAMANGA..... VALOR ACTO: \$ 0.
ESPECIFICACION: MODO DE ADQUISICIÓN : 0145 FUSIÓN		
PERSONAS QUE INTERVIENEN EN EL ACTO (X-Titular de derecho real del dominio, I-Titular de dominio incompleto)		
A: MENDOZA BERNABE	CC# 123456	
FOLIOS DERIVADOS DE LA ANOTACIÓN		
300-388254 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 300-388256 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 300-388258 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 300-388260 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 300-388262 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS		
300-388255 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 300-388257 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 300-388259 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 300-388261 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS		

#### Validación del formulario de calificación y corrección

- Tipo de ordenamiento: orden ascendente, alternado secuencial y en columnas.
- Cantidad de columnas: 2 columnas, de acuerdo a la cantidad de filas por columnas se distribuyen los registros uniformemente.
- Longitud del campo inmueble: 40 caracteres incluyendo los espacios, en caso de sobrepasar esta longitud la información se truncara.
- Longitud del campo matricula: 15 caracteres, incluyendo el código de la ORIP y los espacios.
- Matriculas que se calificaron: En el formulario de calificación solo se mostrara el folio matriz y las matriculas que se calificaron con el turno, es decir que en el caso de existir una segregación o un englobe en esta sección no se mostrarán las matriculas segregadas o englobadas, como se muestra a continuación;

	<b>FORMULARIO DE CALIFICACION CONSTANCIA DE INSCRIPCION</b>
Página: 1	Impreso el 4 de Mayo de 2015 a las 04:53:43 pm
Con el turno 2015-300-6-8871 se calificaron las siguientes matrículas: 300-128829	

- Matriculas que se corrigieron: En el formulario de corrección, solo se mostrara el folio matriz y las matriculas que se corrigieron con el turno, es decir que en el caso de existir una segregación o un englobe en esta sección no se mostrarán las matriculas segregadas o englobadas, como se muestra a continuación;

	<b>FORMULARIO DE CORRECCION</b>
Página: 1	Impreso el 10 de Junio de 2015 a las 05:18:39 pm
Con el turno 2015-342-3-77 se corrigieron las siguientes matrículas: 342-314 342-12578	

## Certificados de tradición

 Página: 4	<b>OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE BUCARAMANGA CERTIFICADO DE TRADICIÓN MATRÍCULA INMOBILIARIA</b> <b>Nro Matrícula: 300-128829</b>  Impreso el 5 de Mayo de 2015 a las 04:01:09 pm <b>"ESTE CERTIFICADO REFLEJA LA SITUACIÓN JURÍDICA DEL INMUEBLE HASTA LA FECHA Y HORA DE SU EXPEDICIÓN"</b> No tiene validez sin la firma del registrador en la última página  <b>NRO TOTAL DE ANOTACIONES: "11"</b>  <b>CON BASE EN LA PRESENTE SE ABRIERON LAS SIGUIENTES MATRÍCULAS</b> 10->300-388249 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 10->300-388250 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 10->300-388251 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 10->300-388252 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 10->300-388253 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388254 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388255 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388256 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388257 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388258 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388259 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388260 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388261 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388262 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS  <b>SALVE DADES: (Información Anterior o Corregida)</b> Anotación Nro: 0 No. corrección: 1 Radicación: ICARE-2015 Fecha: 15/4/2015 SE incluye nuevo número pédrial de 30 dígitos suministrados por el I.G.A.C. (SNC), RES Nro. 8589 DE 27-11-2008 PROFERIDA POR LA SINR (CONVENIO IGAC-SNR DE 23-09-2008)  ===== <b>FIN DE ESTE DOCUMENTO</b>
--	--

### Validación Certificados de tradición

- Tipo de ordenamiento: orden ascendente, alternado secuencial y en columnas.
- Cantidad de columnas: 2 columnas, de acuerdo a la cantidad de filas por columnas se distribuyen los registros uniformemente.
- Longitud del campo inmueble: 40 caracteres incluyendo los espacios, en caso de sobrepasar esta longitud la información se truncara.
- Longitud del campo matricula: 15 caracteres, actualmente el certificado incluye la anotación del folio matriz, el código de la ORIP, caracteres especiales y los espacios, es decir 10->300-388249, en el caso de que el campo matricula excede la longitud definida, se tomara espacios del campo inmueble hasta completar la información del campo matricula.

### Impresión simple de folio

<b>NRO TOTAL DE ANOTACIONES: "11"</b>  <b>CON BASE EN LA PRESENTE SE ABRIERON LAS SIGUIENTES MATRÍCULAS</b> 10->300-388249 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 10->300-388250 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 10->300-388251 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 10->300-388252 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 10->300-388253 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388254 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388255 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388256 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388257 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388258 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388259 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388260 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388261 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS 11->300-388262 LOTE SALDOS NUMERO DOS(2) LAS PRIMAVERAS
--

- Tipo de ordenamiento: orden ascendente, alternado secuencial y en columnas.
- Cantidad de columnas: 2 columnas, de acuerdo a la cantidad de filas por columnas se distribuyen los registros uniformemente.
- Longitud del campo inmueble: 40 caracteres incluyendo los espacios, en caso de sobrepasar esta longitud la información se truncara.

-Longitud del campo matricula: 15 caracteres, actualmente el certificado incluye la anotación del folio matriz, el código de la ORIP, caracteres especiales y los espacios, es decir 10->300-388249, en el caso de que el campo matricula excede la longitud definida, se tomara espacios del campo inmueble hasta completar la información del campo matricula.

#### **Alcance funcional del requerimiento**

La información del inmueble de las matriculas segregadas y englobadas, se define en los procesos de los siguientes imprimibles que genera el aplicativo SIR;

- Certificados tradición: Inmediatos, ampliación de tradición, pertenencia, antiguo sistema y Reimpresión de certificados.
- Corrección: Certificados asociados, formulario de corrección temporal y definitivo.

Actualmente el formulario de corrección, en el folio matriz no contempla los folios derivados de la anotación, por lo tanto se debe realizar la adaptación del diseño a este formato basado en el formulario de calificación.

- Registro: Certificados asociados, formulario de calificación temporal y definitivo.
- Pantallas Administrativas: Impresión simple de folio. Reimpresión formulario de calificación y reimpresión formulario de corrección.
- Certificados de tradición generados por el aplicativo de Certificados de tradición en línea (CTL).

#### **1157\_111\_No incluir linderos propiedad horizontal**

De conformidad a la Ley 1579 de 2012, se debe dar cumplimiento para agregar linderos teniendo en cuenta los casos de reglamento de propiedad horizontal, con base a los artículos que se relacionan a continuación;

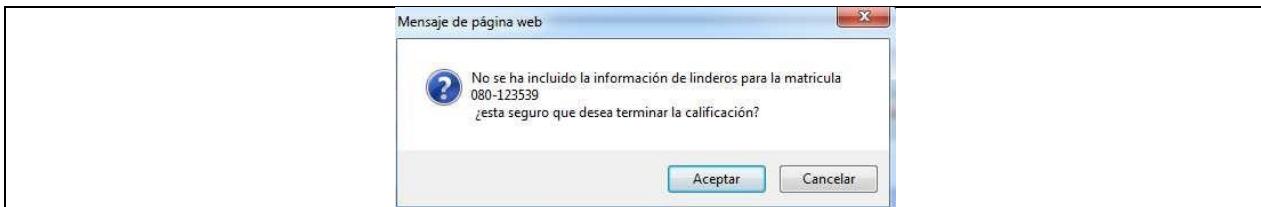
Artículo 8. Parágrafo 1°. Sólo se podrá omitir la transcripción de los linderos en el folio de matrícula inmobiliaria, para las unidades privadas derivadas de la inscripción del régimen de propiedad horizontal.

Artículo 66. En adelante en todos los folios de matrícula deberán consignarse los datos relativos a la descripción, cabida y linderos del predio de que se trate, los cuales se transcribirán en su totalidad a excepción de los inmuebles derivados del régimen de propiedad horizontal donde bastará la cita de la escritura pública que los contenga.

En los procesos de registro y corrección para los folios creados, segregados y englobados en estado temporal en los cuales se han incluido anotaciones el sistema debe validar la naturaleza jurídica con la que realiza la apertura del nuevo folio, si es obligatorio u opcional para el ingreso de cabida y linderos.

Teniendo en cuenta lo anterior se realizaran las validaciones en los siguientes casos;

- Las matriculas nuevas en estado temporal, si la primera anotación derivada del folio, la naturaleza jurídica contiene código opcional, al avanzar el turno el sistema mostrara el mensaje “No se ha incluido la información de linderos para la matrícula XXX-XXXX ¿Está seguro que desea terminar la calificación?”, si selecciona “Cancelar” el turno no debe avanzar a la siguiente fase o si selecciona “Aceptar” el turno debe avanzar a la siguiente fase, como se muestra en la siguiente imagen;



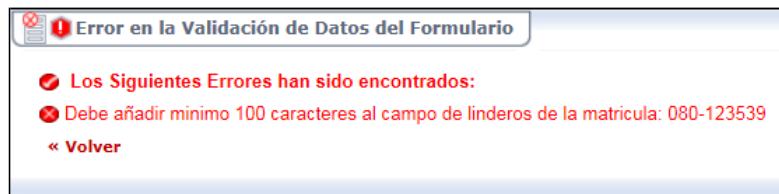
Este mensaje de confirmación debe mostrarse al avanzar los turnos de registro de la fase de digitador masivo a calificación y de la fase de calificación a mesa de control, así mismo en el proceso de correcciones al avanzar de la fase ejecutar corrección simple a la fase de revisar y aprobar, permitiendo continuar con el proceso así tengan o no cabida y linderos estas matriculas temporales las cuales tienen anotaciones que cumplan las condiciones anteriormente mencionadas.

El siguiente listado son los códigos de naturaleza jurídica en los cuales es opcional incluir los linderos, pero que el sistema debe permitir ingresarlos si los calificadores o registradores lo consideran necesario.

Código	Nombre
0102	Acta entrega de inmuebles a entidad
0107	Adjudicación de la cosa hipotecada
0108	Adjudicación en remate
0117	Adquisición por absorción de acciones y patrimonio de entidades financieras
0120	Caducidad administrativa
0127	Consolidación de dominio pleno
0139	Escisión
0143	Fusión
0144	Permuta
0154	Restitución en fideicomiso civil
0155	Restitución en fiducia mercantil
0164	Transferencia de dominio a título de beneficio en fiducia mercantil
0168	Transferencia de dominio a título de leasing habitacional de vivienda familiar
0169	Transferencia de dominio a título de leasing habitacional de vivienda no familiar
0170	Titulación por saneamiento contable
0181	Adjudicación en proceso de liquidación judicial art. 58 ley 1116 de 2006
0301	Adjudicación sucesión derecho de cuota
0302	Adjudicación sucesión nuda propiedad
0303	Adjudicación sucesión subsuelo con antecedente registral
0307	Compraventa derechos de cuota
0308	Compraventa nuda propiedad
0309	Compraventa subsuelo con antecedente registral
0310	Compraventa usufructo
0316	Constitución régimen de condominio
0317	Constitución reglamento de propiedad horizontal
0322	Donación nuda propiedad
0330	Reforma régimen de condominio
0331	Reforma reglamento de propiedad horizontal
0332	Remate derecho de cuota

0346	Adición régimen de condominio
0347	Adición régimen de propiedad horizontal
0348	Adjudicación liquidación de la sociedad conyugal derecho de cuota
0349	Adjudicación liquidación de la sociedad conyugal nuda propiedad
0357	Declaración de reserva, alinderacion y creación de área del sistema de parques nacionales y naturales
0359	Declaratoria de bien inmueble de interés cultural numeral 1.2. Art. 7 de la ley 1185 de 2008
0360	Declaratoria de bien existencia del plan especial de manejo y protección numeral 1.2. Art. 7 de la ley 1185 de 2008
0834	Extinción parcial régimen de condominio
0835	Extinción parcial régimen de propiedad horizontal
0836	Extinción total régimen de condominio
0837	Extinción total régimen de propiedad horizontal
0907	Cambio de razón social

- Las matriculas nuevas en estado temporal, si la primera anotación derivada del folio, la naturaleza jurídica contiene código obligatorio, al avanzar el turno el aplicativo debe obligar a incluir la información de linderos de conformidad con el artículo 8º de la ley 1579 de 2012, el cual consta mínimo de cien (100) caracteres, a continuación del texto pre-establecido por el sistema, de lo contrario debe mostrar un mensaje restrictivo indicando que se ha omitido diligenciar el campo con la información de linderos, como se muestra en el siguiente imagen;



Este mensaje de validación debe mostrarse al avanzar los turnos de registro de la fase de digitador masivo a calificación y de la fase de calificación a mesa de control, así mismo en el proceso de correcciones al avanzar de la fase ejecutar corrección simple a la fase de revisar y aprobar.

El siguiente listado son los códigos de naturaleza jurídica en los cuales es obligatorio incluir los linderos.

Código	Nombre
0101	Accesión del suelo
0103	Adjudicación baldíos
0104	Adjudicación baldíos en propiedad colectiva a comunidades negras
0105	Adjudicación baldíos resguardos indígenas
0106	Adjudicación de bienes vacantes
0109	Adjudicación en sucesión
0110	Adjudicación liquidación de la comunidad
0111	Adjudicación liquidación sociedad comercial
0112	Adjudicación liquidación sociedad conyugal
0113	Adjudicación liquidación sociedad patrimonial de hecho
0114	Adjudicación por expropiación
0115	Adjudicación sucesión partición adicional

	0116	Adjudicación unidad agrícola familiar
	0118	Aporte a sociedad
	0119	Aporte de subsidio en especie
	0121	Cesión a título gratuito de bienes fiscales
	0122	Cesión de bienes obligatoria
	0124	Cesión obligatoria de zonas con destino a uso publico
	0125	Compra venta
	0126	Compraventa parcial
	0128	Constitución de fiducia mercantil
	0129	Dación en pago
	0130	Dación en pago obligatoria
	0131	Declaración judicial de pertenencia
	0133	Declaratoria de propiedad pública sobre zonas de cesión obligatoria gratuita
	0138	Donación
	0140	Expropiación por vía administrativa
	0141	Expropiación por vía judicial
	0145	Prescripción agraria
	0162	Título de minas con antecedente registral
	0163	Transacción
	0165	Transferencia de dominio por solución o pago efectivo
	0166	Adjudicación terreno ejido
	0167	Adjudicación liquidación sociedad de hecho
	0172	Transferencia de dominio de bienes fiscales
	0173	Saneamiento titulación ley no. 1182 de 2008
	0175	Cesión de baldío urbano de la nación al municipio. Instrucción administrativa no. 18 de 2009 (art. 123 de la ley 388 de 1997)
	0176	Formalización del título de propiedad artículo 139 de la ley 1151 de 2007
	0313	Constitución de fideicomiso civil
	0321	Deslinde y amojonamiento
	0903	Actualización de linderos
	0913	Declaración parte restante
	0915	Des englobe
	0916	Deslinde de terreno de propiedad de la nación
	0917	Determinación área y linderos predio del municipio
	0918	División material
	0919	Englobe
	0920	Loteo
	0924	Reloteo

- Las matriculas nuevas en estado temporal, si la primera anotación derivada del folio, la naturaleza jurídica no pertenece a un código opcional u obligatorio, al avanzar el turno a la siguiente fase el sistema no deberá mostrar mensajes ni restricciones continuando con el trámite del turno.
- las matriculas temporales que han sido objeto de creación, englobe o segregación y que contengan naturaleza jurídica con código 0125 Compra venta, el sistema debe obligar a incluir linderos, cuando el turno de registro se avance de la fase de digitador masivo y de la fase de calificación.

### **1158\_111\_Inactivar tarifa especial certificados**

En el proceso de radicación registro se debe inactivar la tarifa especial, de acuerdo a la resolución N° 0640 del 23 de Enero de 2015, esta tarifa no se aplicará para ningún proceso registral.

Para inactivar la tarifa especial de \$1.200 en el proceso de Registro, se debe tener en cuenta los siguientes aspectos:

- En la fase solicitud del rol Liquidador Registro:

\* Inactivar el tipo de tarifa especial, cuando exista una solicitud de un certificado asociado a un turno de registro.

\* Inactivar el tipo de tarifa especial, en la opción de Preliquidar en un turno de registro con certificado asociado.

- En la fase calificación del rol Calificador:

\* En la opción Información Turno – Preliquidar, se debe inactivar el tipo de tarifa especial.

\*Se debe guardar y conservar el historial de los turnos, donde se ha aplicado este tipo de tarifa anteriormente.

### **1203\_111\_Ajuste versión categoria reparto notarial**

De acuerdo a la problemática presentada en la Oficina de Cartagena con la Notaría Segunda, se ajusta que en el aplicativo SIR, se valide la última versión de la categoría de la oficina origen y así al realizar el reparto notarial el sistema solo repartirá las minutas a la última versión de la notaría en su correspondiente categoría.

Este ajuste se realizara para las todas las oficinas origen de tipo notaria, en el Versionamiento de la categoría, guardando el histórico de las versiones anteriores.

**DESPLIEGUE – 4.2.1  
05/08/2015**

### **1209\_111\_Ajuste Imprimible Certificado Botón de Pago - SIR**

Con el objetivo de cumplir una de las actividades de la oficina de TI de la SNR, donde se busca permitir que cualquier ciudadano pueda solicitar y adquirir un certificado de tradición mediante los aplicativos SIR y/o Botón de Pago; en este momento la presentación de dichos certificados es diferentes en las oficinas de Soacha y Soledad, por tal motivo se requiere que estos dos certificados sean idénticos en cuanto a la estructura y contenido en cada uno de ellos.

Se requiere que los certificados de tradición expedidos mediante el aplicativo SIR y Botón de pago sean idénticos en cuanto a la estructura y el contenido de la información de cada inmueble.

Actualmente cuando se solicita un Certificado de Libertad por el aplicativo SIR de las oficinas de Soacha y Soledad sale con la información del traslado, tal como se muestra a continuación:

Impreso el 25 de Mayo de 2015 a las 01:49:24 pm  
**"ESTE CERTIFICADO REFLEJA LA SITUACION JURIDICA DEL INMUEBLE  
HASTA LA FECHA Y HORA DE SU EXPEDICION"**

No tiene validez sin la firma del registrador en la última página

CIRCULO REGISTRAL: 051 SOACHA DEPTO: CUNDINAMARCA MUNICIPIO: SOACHA VEREDA: SOACHA  
FECHA APERTURA: 24/9/2014 RADICACION: 2014-80694 CON: ESCRITURA DE 9/9/2014

ESTADO DEL FOLIO: **ACTIVO**

COD CATASTRAL:  
COD CATASTRAL ANT: SIN INFORMACION

**INFORMACION TRASLADO DE MATRICULA**

**FUNDAMENTOS LEGALES**

Decreto(s):

Resolución(es) de Traslado Circulo Origen: Número: 278 Fecha 14/05/2015

Resolución(es) de Circulo Destino: Número: 1 Fecha 17/05/2015

Círculo Registral Origen: 50S BOGOTA ZONA SUR Matrícula Origen: 50S-40672622

Pero cuando el procedimiento es realizado a través del aplicativo de Botón de Pago la presentación que actualmente tiene el certificado en las oficinas de Soacha y Soledad es la siguiente:

Impreso el 20 de Mayo de 2015 a las 10:52:33 am  
**"ESTE CERTIFICADO REFLEJA LA SITUACION JURIDICA DEL INMUEBLE  
HASTA LA FECHA Y HORA DE SU EXPEDICION"**

No tiene validez sin la firma del registrador en la última página

CIRCULO REGISTRAL: 051 SOACHA DEPTO: CUNDINAMARCA MUNICIPIO: SOACHA VEREDA: SOACHA  
FECHA APERTURA: 19/12/1995 RADICACION: 1995-90516 CON: ESCRITURA DE 11/12/1995

ESTADO DEL FOLIO: **ACTIVO**

COD CATASTRAL: 2575401020000041809019000003  
COD CATASTRAL ANT: 25754010204180337901

**DESCRIPCION: CABIDA Y LINDEROS:**

CONTENIDOS EN ESCRITURA NRO 6100 DE FECHA 11-12-95 EN NOTARIA 5 DE SANTAFE DE BOGOTA APARTAMENTO 102 INTERIOR 34 CON AREA DE 38.34 MTS2 CON COEFICIENTE DE .38 (SEGUN DECRETO 1711 DE JULIO 6/84).

**COMPLEMENTACION:**

HOLGUINES S.A. ADQUIRIO EN MAYOR EXTENSION POR COMPRA A POLITECNICO GRANCOLOMBIANO, POR LAESCRITURA #3327 DEL 30-11-1993 NOTARIA 44 DE BOGOTA, REGISTRADA ALFL. 050-0831375.-ESTE HUBO POR COMPRA A INVERSIONES Y

Teniendo en cuenta que se requiere que ambos documentos tengan la misma estructura y contenido como el que actualmente se genera en el aplicativo SIR, es necesario:

1. El imprimible del certificado de tradición emitido a través de Botón de Pago, debe incluir la información que se encuentra señalada en esta última imagen, tales como:
  - ✓ Información correspondiente al traslado de la matrícula:
    - Decreto(s)
    - Resolución(es) de Traslado Círculo Origen
    - Resolución(es) de Círculo Destino
    - Círculo Registral Origen
    - Matrícula Origen.
2. La información correspondiente a los campos Código Catastral y Código Catastral Ant, deben ajustarse al margen derecho del imprimible respetando el margen establecido para el certificado y no como está ocurriendo actualmente.
3. En la parte final del imprimible del certificado debe aparecer la información referente al código de identificación NIS y la URL para la verificación de la autenticidad del certificado de tradición y libertad, tal como evidencia en el imprimible generado en el aplicativo SIR:

**FIN DE ESTE DOCUMENTO**

El interesado debe comunicar cualquier falla o error en el registro de los documentos

USUARIO: 64964 impreso por: 64964

TURNO: 2015-008-1-6514 FECHA:17/6/2015

NIS: KMM/LPz30JrvjVCEnnZmrijDPyz899CauW1U8hv7UKg=

Verificar en: [http://172.30.1.131:8190/W\\$-SIRClient/](http://172.30.1.131:8190/W$-SIRClient/)

**EXPEDIDO EN: APARTADO**

Ya que en el certificado de Botón de Pago, solo se nombran los campos pero no se visualiza esta información, tal como se evidencia en la siguiente imagen:

**FIN DE ESTE DOCUMENTO**

El interesado debe comunicar cualquier falla o error en el registro de los documentos

USUARIO: 57372 impreso por: 57372

TURNO: 2015-041-1-339 FECHA:19/5/2015

NIS:

Verificar en:

**EXPEDIDO EN: PORTAL**

4. Las validaciones que se tienen actualmente en el aplicativo SIR para la generación de certificados contempla que los folios a asociar no se encuentren en los siguientes estados y/o bloqueos:

4.1 Folios en estado:

- Bloqueado
- Anulado
- Custodia
- Trasladado

Actualmente en Botón de Pago es posible generar certificados en estado Bloqueado, para corregir esta inconsistencia se tendrá en cuenta la validación del estado bloqueado (ID\_ESTADO = B) que se almacena en la tabla SIR\_NE\_FOLIO.

4.2 Folio bloqueado con turno en trámite

Al momento de realizar la validación de turnos en trámite, se debe tener en cuenta el requerimiento de fase entregado, donde se incluyeron las nuevas fases mencionadas en la siguiente tabla:

PROCESO	ROL	FASE
Registro	Entrega por Correspondencia	Desanotado Correspondencia
	Entrega Externa	Entrega Externa
	Entrega Documentos	Desanotado
		Entrega

	Correcciones	Entrega Correcciones	Corrección Desanotada
Entrega Corrección			

## DESPLIEGUE – 4.2.0

15/07/2015

### **1195\_111\_Impresión\_Certificados\_Exentos\_PDF**

La Superintendencia de Notariado y Registro requiere que el Sistema de Información registral SIR, genere los Certificados Tipo Exentos en formato PDF, de tal forma que se permita el almacenamiento de los mismos en medios externos.

Dando alcance a la circular 249 del 13 de abril de 2015, sobre la prohibición de impresión en la expedición de certificados de tradición y libertad para entidades exentas, mediante el aplicativo SIR y la circular 251 del 15 de abril de 2015 donde se comunica el plazo en el que entrara a regir (13 de mayo de 2015) dicha norma, se solicita desarrollar un componente en SIR que permita presentar los certificados de tipo exentos en PDF para ser entregados en medios magnéticos a las entidades públicas que los soliciten.

En el marco de este requerimiento se establece que, a través del rol Usuario Operativo se generarán los certificados masivos exentos, tal como se viene manejando, es decir, conservando funcionalidades existentes en el Sistema de Información Registral SIR y las reglas del negocio, como la generación de turno, el avance a la fase de entrega hasta la finalización del mismo, salvo la validación que realiza el aplicativo de no permitir la expedición de certificados con matrículas asociadas de mayor extensión que para este desarrollo si permitirá realizarse a través del rol Usuario Operativo.

Solo se modificaría la impresión del certificado, que no sería en medio físico a través del administrador de impresión, por el tipo de impresión Certificados Masivos, sino se generará mediante un archivo PDF, el cual se almacenará en cada uno de los computadores donde se realice la solicitud, en la ruta seleccionada por el usuario.

El nombre del archivo PDF generado, será igual al número de la matrícula asociada a la solicitud, con la respectiva extensión .pdf. Si por alguna razón, en el caso que fallará la generación del archivo pdf, se radicaría una nueva solicitud y por ende, un nuevo turno.

Es de aclarar que el imprimible del reporte de certificados masivos que resume la solicitud del turno masivo, se comportará como viene funcionando, es decir, se generará solo este imprimible en físico.

En cuanto a la opción de imprimir Nota Informativa, esta se deberá eliminar de la pantalla inicial de la solicitud.

Para los certificados individuales con tarifa exenta, que se generan a través del rol Cajero Certificados, el comportamiento continuará siendo el mismo que maneja actualmente el aplicativo SIR.

En cuanto al número o cantidad de matrículas que se pueden cargar y generar por solicitud será de 230, esto con el fin de que no halla afectación, ni ocasiona impacto en el normal funcionamiento, en cuanto a la prestación del servicio en las ORIPS.

### **1202\_111\_Notas\_Devolutivas\_Cambio\_Texto**

Teniendo en cuenta que mediante el Decreto 2723 del 29 de Diciembre de 2014, se modifica la estructura de la Superintendencia de Notariado y Registro, es necesario ajustar el texto

predeterminado del Formato de Notas Devolutivas.

En el Formato de Notas Devolutivas establecido en el Aplicativo SIR, se requiere modificar el texto predefinido en el Reporte de Nota Devolutiva que actualmente cita como se muestra a continuación:

UNA VEZ SUBSANADA(S) LA(S) CAUSAL(ES) QUE MOTIVO LA NEGATIVA DE INSCRIPCION, FAVOR RADICAR NUEVAMENTE EN ESTA OFICINA, EL DOCUMENTO PARA SU CORRESPONDIENTE TRAMITE, ADJUNTANDO LA PRESENTE NOTA DEVOLUTIVA.

CUANDO LA CAUSAL O LAS CAUSALES QUE RECHAZA (N) LA INSCRIPCION DEL DOCUMENTO NO SEA (N) SUBSANABLE (S), SE CONFIGURE EL PAGO DE LO NO DEBIDO, SE PRODUZCAN PAGOS EN EXCESO O SE DESISTA DEL TRAMITE, EL TERMINO PARA SOLICITAR LA DEVOLUCION DE DINERO PARA LOS DERECHOS DE REGISTRO ES DE CUATRO (4) MESES CALENDARIO, CONTADOS A PARTIR DE LA FECHA DE EJECUTORIA DEL ACTO ADMINISTRATIVO QUE NIEGA EL REGISTRO O SEDESANOTE EL DOCUMENTO INSCRITO.

PARA SOLICITAR LA DEVOLUCION DE DINERO POR CONCEPTO DE IMPUESTO DE REGISTRO, DEBE DIRIGIRSE A LA GOBERNACION DEL DEPARTAMENTO O DE RISARALDA, EN LOS TERMINOS DEFINIDOS POR EL ARTICULO 15 DEL DECRETO 650 DE 1996.

LOS ACTOS O NEGOCIOS JURIDICOS A QUE SE REFIERE EL ARTICULO 4 DE LA LEY 1579 DE 2012, DEBERAN PRESENTARSE PARA SU INSCRIPCION, DENTRO DE LOS DOS (2) MESES CALENDARIO, SIGUIENTES A LA FECHA DE SU OTORGAMIENTO PARA ACTOS NOTARIALES O LA FECHA DE EJECUTORIA PARA PROVIDENCIAS JUDICIALES O ADMINISTRATIVAS, VENCIDOS LOS CUales, SE COBRARAN INTERESES MORATORIOS POR IMPUESTO DE REGISTRO, PREVISTOS EN LA LEY 223 DE 1995 Y SU DECRETO REGLAMENTARIO **695** DE 1996 ARTICULO 14.

EXCEPTUESE DE LO ANTERIOR, LOS CASOS RELACIONADOS CON EL NEGOCIO JURIDICO DE HIPOTECA Y EL ACTO DE CONSTITUCION DE PATRIMONIO DE FAMILIA DE QUE TRATA EL ARTICULO 28 DE LA LEY 1579 DE 2012, LOS CUales SE DEBEN REGISTRAR DENTRO DE LOS NOVENTA (90) DIAS HABILES SIGUIENTES A SU AUTORIZACION, VENCIDO EL TERMINO REGISTRAL ANTES SEÑALADO DEBERAN CONSTITUIRSE DE CONFORMIDAD CON EL PRECITADO ARTICULO 10.

CONTRA EL PRESENTE ACTO ADMINISTRATIVO PROCEDA EL RECURSO DE REPOSICION ANTE EL REGISTRADOR DE INSTRUMENTOS PUBLICOS Y EN SUBSIDIO EL DE APELACION ANTE **LA SUBDIRECCION DE REGISTRO** DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO DENTRO DE LOS DIEZ(10) DIAS HABILES SIGUIENTES A SU NOTIFICACION, EN VIRTUD DE LO PREVISTO POR EL NUMERAL DOS (2) DEL ARTICULO **20** DEL DECRETO **2163** DEL 17 DE JUNIO DE 2014, PREBADO EL CUMPLIMENTO DE LOS REQUISITOS ESTABLECIDOS EN LOS ARTICULOS 76 Y 77, LEY 1437 DE 2011 (CODIGO DE PROCEDIMIENTO ADMINISTRATIVO Y DE LO CONTENCIOSO ADMINISTRATIVO).

**REGISTRO Y REGISTRO**  
LA GUARDIA DE LA FE PUBLICA

FUNCIONARIO CALIFICADOR  
4073

REGISTRADOR DE INSTRUMENTOS PUBLICOS

-----  
**FIN DE ESTE ACTO ADMINISTRATIVO**

Teniendo en cuenta lo anterior solo se modificará el texto resaltado y por la nueva información indicada por la SNR, este cambio se relaciona y resalta en la siguiente imagen:

UNA VEZ SUBSANADA(S) LA(S) CAUSAL(ES) QUE MOTIVO LA NEGATIVA DE INSCRIPCION, FAVOR RADICAR NUEVAMENTE EN ESTA OFICINA, EL DOCUMENTO PARA SU CORRESPONDIENTE TRAMITE, ADJUNTANDO LA PRESENTE NOTA DEVOLUTIVA.

CUANDO LA CAUSAL O LAS CAUSALES QUE RECHAZA (N) LA INSCRIPCION DEL DOCUMENTO NO SEA (N) SUBSANABLE (S), SE CONFIGURE EL PAGO DE LO NO DEBIDO, SE PRODUZCAN PAGOS EN EXCESO O SE DESISTA DEL TRAMITE, EL TERMINO PARA SOLICITAR LA DEVOLUCION DE DINERO PARA LOS DERECHOS DE REGISTRO ES DE CUATRO (4) MESES CALENDARIO, CONTADOS A PARTIR DE LA FECHA DE EJECUTORIA DEL ACTO ADMINISTRATIVO QUE NIEGA EL REGISTRO O SEDESANOTE EL DOCUMENTO INSCRITO.

PARA SOLICITAR LA DEVOLUCION DE DINERO POR CONCEPTO DE IMPUESTO DE REGISTRO, DEBE DIRIGIRSE A LA GOBERNACION DEL DEPARTAMENTO O DE RISARALDA, EN LOS TERMINOS DEFINIDOS POR EL ARTICULO 15 DEL DECRETO 650 DE 1996.

LOS ACTOS O NEGOCIOS JURIDICOS A QUE SE REFIERE EL ARTICULO 4 DE LA LEY 1579 DE 2012, DEBERAN PRESENTARSE PARA SU INSCRIPCION, DENTRO DE LOS DOS (2) MESES CALENDARIO, SIGUIENTES A LA FECHA DE SU OTORGAMIENTO PARA ACTOS NOTARIALES O LA FECHA DE EJECUTORIA PARA PROVIDENCIAS JUDICIALES O ADMINISTRATIVAS, VENCIDOS LOS CUales, SE COBRARAN INTERESES MORATORIOS POR IMPUESTO DE REGISTRO, PREVISTOS EN LA LEY 223 DE 1995 Y SU DECRETO REGLAMENTARIO **695** DE 1996 ARTICULO 14.

EXCEPTUESE DE LO ANTERIOR, LOS CASOS RELACIONADOS CON EL NEGOCIO JURIDICO DE HIPOTECA Y EL ACTO DE CONSTITUCION DE PATRIMONIO DE FAMILIA DE QUE TRATA EL ARTICULO 28 DE LA LEY 1579 DE 2012, LOS CUales SE DEBEN REGISTRAR DENTRO DE LOS NOVENTA (90) DIAS HABILES SIGUIENTES A SU AUTORIZACION, VENCIDO EL TERMINO REGISTRAL ANTES SEÑALADO DEBERAN CONSTITUIRSE DE CONFORMIDAD CON EL PRECITADO ARTICULO 10.

CONTRA EL PRESENTE ACTO ADMINISTRATIVO, PROCEDA EL RECURSO DE REPOSICION ANTE EL REGISTRADOR DE INSTRUMENTOS PUBLICOS Y EN SUBSIDIO, EL DE APELACION ANTE **LA SUBDIRECCION DE APOYO JURIDICO REGISTRAL** DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO DENTRO DE LOS DIEZ(10) DIAS HABILES SIGUIENTES A SU NOTIFICACION, EN VIRTUD DE LO PREVISTO POR EL NUMERAL DOS (2) DEL **ARTICULO 21** DEL DECRETO **2723** DEL 29 DE DICIEMBRE DE 2014, PREBADO EL CUMPLIMENTO DE LOS REQUISITOS ESTABLECIDOS EN LOS ARTICULOS 76 Y 77, LEY 1437 DE 2011 (CODIGO DE PROCEDIMIENTO ADMINISTRATIVO Y DE LO CONTENCIOSO ADMINISTRATIVO).

**REGISTRO Y REGISTRO**  
LA GUARDIA DE LA FE PUBLICA

FUNCIONARIO CALIFICADOR  
4073

REGISTRADOR DE INSTRUMENTOS PUBLICOS

-----  
**FIN DE ESTE ACTO ADMINISTRATIVO**

Es de aclarar que se conservará el historial de las notas devolutivas que se originaron anteriormente en turnos de registro con resolución devuelto o inscrito parcialmente.

#### DESPLIEGUE – 4.1.0

20/03/2014

#### 003\_589\_Error Usuario de Radicación de Turno de Certificado\_Sesion

Debido a la problemática que se está presentando actualmente en el aplicativo SIR que es la mezcla de información entre sesiones. Se realizo el ajuste implementando un filtro que permite direccionar si hay alguien logueado a la página de selección de roles del usuario que este activo en sesión en este momento. Esta solución se desarrolla para asegurar que la información entre sesiones no se mezcle.

Hay que tener en cuenta que dependiendo de la configuración del navegador se permitirá varias

sesiones en diferentes ventanas sin mezclar la información entre las sesiones de las diferentes ventanas.

Cuando hay sesiones de diferentes usuarios logueados y el primer usuario que inicio sesión, realiza alguna acción, el sistema mostrara el mensaje “Sesión cerrada por logueo con otro usuario”, o sino cargara el menú de roles del último usuario que se logueo.

#### **004\_589\_Funcionario\_Fase\_Entregado**

En la consulta de turnos de Correcciones y Registro, en el Bloque Historial de Fases del Turno, en la fase Entregado el aplicativo mostrara el nombre del Funcionario que ejecuta la acción de “Entregado”.

**1.** En el proceso de Registro, en la fase de Certificados Asociados al momento de avanzar un turno individual el sistema crea una relación la cual debe quedar en el historial del turno.

En la fase de “DESANOTADO” se crea un nuevo formulario en el cual se visualizan las relaciones y los turnos asociados a la relación; el sistema permitirá seleccionar una, varias o todas las relaciones, tal como se muestra a continuación;

RELACION	TURNO
<input type="checkbox"/> 2013-300-45	2013-300-6-78
	2013-300-6-79
	2013-300-6-80
<input type="checkbox"/> 2013-300-46	2013-300-6-77

**ACEPTAR**      **CANCELAR**

Cuando el usuario confirme el recibido en la fase DESANOTADO, el sistema asignará el nombre de usuario y la fecha en la cual se realizó la acción. Al avanzar los turnos de esta fase, se visualizaran en la “ENTREGA”

Para la fase de “ENTREGA” el sistema desplegará el formulario que actualmente se despliega en la fase “DESANOTADO”

**2.** Para el proceso de correcciones, en la fase de mesa de control de correcciones, al momento de avanzar un turno individual el sistema crea una relación la cual debe quedar en el historial del turno.

En la fase de “CORRECCION DESANOTADA” se crea un nuevo formulario en el cual se visualizan las relaciones y los turnos asociados a la relación; el sistema permitirá seleccionar una, varias o todas las relaciones, tal como se muestra a continuación;

RELACION	TURNO
<input type="checkbox"/> 2013-300-45	2013-300-3-78
	2013-300-3-79
	2013-300-3-80
<input type="checkbox"/> 2013-300-46	2013-300-3-77

**ACEPTAR**      **CANCELAR**

Cuando el usuario confirme el recibido en CORRECCION DESANOTADA, el sistema asignará el nombre de usuario y la fecha en la cual se realizó la acción. Al avanzar los turnos de esta fase se

visualizaran en la fase “ENTREGA” cuando estos avancen de la fase “CORRECCION DESANOTADA”.

Para la fase de “ENTREGADO” el sistema desplegará el formulario que actualmente se despliega en la fase “CORRECCION DESANOTADA”

También se hace el ajuste para que la nueva fase de los procesos de registro y correcciones cuando un turno se va avanzar desde la fase entregado para finalizar el turno se llame “FINALIZADO”.

Cuando es una corrección Interna, el comportamiento del historial de las fases del turno es diferente ya que los turnos llegan hasta la fase de revisar-aprobar y desde ahí se avanza hasta la fase de entregado.

Historial de fases del turno					
FASE	ATENDIDO POR	FECHA	RELACION FASE	RELACION SIGUIENTE	
SOLICITUD	MIGUELA OSORIO	29/10/2013 03:32:29 PM			
REVISIÓN Y ANÁLISIS	MIGUELA OSORIO	29/10/2013 03:32:29 PM			
EJECUTAR CORRECCION SIMPLE	MARIAN VALENCIA	29/10/2013 03:35:41 PM			
REVISAR APROBAR	MIGUELA OSORIO	29/10/2013 03:54:18 PM			
ENTREGADO		29/10/2013 03:56:21 PM			

Por lo tanto el escenario de corrección interna no aplicaría para este ajuste.

Para el escenario de actuaciones administrativas, como el turno no pasa por la fase de mesa de control correcciones, por lo tanto el aplicativo no genera relación, avanzando el turno de la fase ejecutar actuaciones administrativas a la fase entrega corrección y luego continua el trámite finalizando el turno.

### 3. Pantallas administrativas – administración de relaciones

#### Registro

- En el proceso Registro se incluyó la fase Desanotado, cuando se seleccione esta fase el aplicativo debe desplegar en el campo relación el siguiente ítem;
  - Relación de turnos desanotados a entrega

Cuando se escoge el ítem descrito anteriormente el aplicativo mostrará los campos número relación y el botón consultar. En el campo número relación se debe ingresar el id de la relación y al dar clic sobre el botón consultar, el sistema desplegará los turnos que contienen esa relación permitiendo escoger uno, varios o todos los turnos de esta relación.

Si se requiere avanzar los turnos por relación a la fase Entregado se debe dar clic en el botón aceptar de lo contrario se debe dar clic en el botón volver para regresar a las pantallas administrativas.

- En el proceso Registro se incluyó la fase Desanotado por correspondencia, cuando se seleccione esta fase el aplicativo debe desplegar en el campo relación los siguientes ítems;
  - Relación de turnos desanotados correspondencia a entrega externa

Cuando se escoge alguno de los ítems descritos anteriormente el aplicativo mostrará los campos número relación y el botón consultar. En el campo número relación se debe ingresar el id de la relación y al dar clic sobre el botón consultar, el sistema desplegará los turnos que contienen esa relación permitiendo escoger uno, varios o todos los turnos de esta relación.

Si se requiere avanzar los turnos por relación a la fase Entrega externa se debe dar clic en el botón aceptar de lo contrario se debe dar clic en el botón volver para regresar a las pantallas administrativas.

### **Pantallas administrativas – Reporte Rel 01**

El reporte rel 01 se debe ajustar para los siguientes procesos;

- Registro se debe incluir las fases desanotado y desanotado por correspondencia.
- Correcciones se debe incluir la fase corrección desanotada.

Para los turnos anteriores a este desarrollo que están en la fase “DESANOTADO” el aplicativo les creara un número de relación para que puedan ser avanzados a la siguiente fase y ser finalizados sin presentar inconvenientes de acuerdo a la nuevo ajuste en el aplicativo SIR.

#### **022\_589\_Nota\_Informativa\_Mayor\_Valor**

Para dar cumplimiento a la Normatividad vigente, se requiere en el Aplicativo SIR, ajustar la nota informativa de la Notificación al Ciudadano, para el pago de Mayor Valor, excluyendo de su contenido el texto “Resolución número 0035 de enero 9 de 2009 y en su lugar incluir “Resolución No. 0089 de Enero 8 de 2014”.

También se debe excluir de la Nota informativa el texto “Resolución 069 de 2011 de la Superintendencia de Notariado y Registro”.

En conclusión en la Nota informativa debe figurar dentro del paréntesis :” (Decreto No. 2280 de 2008 ,Resolución No.0089 de Enero 8 de 2014, de la Superintendencia de Notariado y Registro)”.

#### **027\_589\_Acto\_liquidación\_copias**

De conformidad con la Resolución 0126 del 9 de enero de 2013, que regula los derechos registrales no figura el Acto de Liquidación “ 05- COPIAS” y la tarifa a aplicable para este acto.

Al radicar un documento con copias adicionales, la finalidad de estas copias adjuntas consiste en reproducir en ellas la constancia de inscripción del documento objeto de registro.

La resolución 126 de 9 de enero de 2013 en concordancia con el Decreto 2280 de 2008 fija las tarifas por el concepto del ejercicio de la función registral y en estos no se contempla el acto “05-Copias” que figura en la lista de actos de liquidación Registro del Aplicativo SIR . En esta normatividad se contempla solo la Constancia de inscripción que de acuerdo a la ley “El registrador debe Reproducir sobre la copia autentica o autenticada del documento inscrito” para la cual se fija una tarifa de \$ 9.600 y se encuentra en la lista actos con el código “012-R reproducción constancia de inscripción”.

Con base en las anteriores consideraciones y teniendo en cuenta que no existe fundamento legal para mantener Activo el Acto de Liquidación 05-Copias, se requiere que éste sea inactivado y se mantenga el historial del uso dado a este acto.

#### **028\_589\_Formato\_de\_certificado\_expedido\_para\_matrículas\_trasladas**

En cumplimiento de los diferentes Decretos y actuaciones administrativas que se profieran para llevar a cabo traslados de folios de matrículas inmobiliarias de un circulo registral a otro, se requiere ajustar el imprimible de los certificados de tradición que hayan sido objeto de traslado, con el fin de que los diferentes usuarios registrales tengan pleno conocimiento de la actuación realizada, de sus motivos y de la asignación del nuevo número de matrícula inmobiliaria.

Esta información deberá estar disponible para incluirla en cada uno de los certificados de matrícula inmobiliaria en cualquier caso de traslado de folios.

En virtud del Decreto 911 del 08 de mayo de 2013. El cual Resuelve: Artículo 1°. Modifícase los círculos registrales de Dabeiba, Frontino, Turbo (Antioquia), Artículo 2°. Créase el Círculo Registral de Apartadó, Departamento de Antioquia con una Oficina Seccional de Registro de Instrumentos Públicos”, para las matrículas inmobiliarias que han sido objeto de traslado es necesario ajustar el formato del imprimible del Certificado de Tradición en el cual se informe al ciudadano la normatividad mediante la cual se fundamenta el cambio de matrícula y Círculo Registral

En las inmobiliarias que han sido objeto de traslado, se requiere que en el imprimible del Certificado de Tradición expedido por cualquier medio, se visualice la siguiente información :

Como título ““INFORMACION TRASLADO DE MATRICULA”

**FUNDAMENTO DEL TRASLADO : DECRETO 911 DEL 08 DE MAYO DE 2013**

(Este Decreto solo Aplica para las matrículas trasladadas del Círculo de Dabeiba Municipio de Carepa, Chigorodó ---del Círculo de Turbo Municipio Apartado” y del Círculo de Frontino Municipio de Mutata ., para las matrículas trasladadas del Círculo de Frontino (Círculo Origen) a Dabeiba se debe agregar la Resolución 02 del 11-06-2013 )

Círculo Registral Origen:	<input type="text"/>	Número Matrícula Origen:	<input type="text"/>
Resolución de Traslado Círculo Origen:	<input type="text"/>	Fecha:	<input type="text"/>
Resolución de Círculo Destino:	<input type="text"/>	Fecha:	<input type="text"/>

Para este tipo de matrículas en la Oficina Destino, esta información se debe visualizar en la consulta de matrículas de cualquier proceso y pantallas Administrativas en el cual se consulte.

En la Oficina de Origen, en la consulta de matrícula por pantallas administrativas se visualizará la información mencionada.

Actualmente se visualiza en la consulta de matrícula como esta imagen, pero debe ajustarse de acuerdo al presente requerimiento.

En la impresión simple de Folio también debe visualizarse la información solicitada,

## DESPLIEGUE – 4.0.0

11/10/2013

### 005\_589\_Requerimiento\_Reporte\_G02

Para que las Orips y el Nivel Central SNR, puedan obtener información consolidada de los turnos repartidos a cada uno de los calificadores y la gestión realizada en el proceso de calificación, se ajustó el Reporte G02 - INFORME ESTADÍSTICO DE TURNOS POR CALIFICADOR.

El reporte G02 contiene las siguientes especificaciones:

Criterio para generar el reporte:

- Rango de fechas: fecha inicial, fecha final
- Circulo registral (por defecto debe traer el circulo donde se está generando el reporte).
- Calificador: lista de selección con los nombres de los calificadores e incluir ítem TODOS (el reporte se puede generar para uno, varios o todos los calificadores).

El imprimible del reporte

- El título debe ser: INFORME ESTADÍSTICO DE TURNOS POR CALIFICADOR.
- Nombre de la Oficina de Registro que genera el reporte y numero del circulo
- Rango de fechas
- Fecha y hora de la impresión.

Las columnas del reporte:

- CALIFICADOR (Nombre y apellidos)
- TURNOS REPARTIDOS: son los turnos asignados por reparto y recibidos por reasignación, reanotar y restitución, cuando un turno ha sido asignado por reparto a un calificador y se reasigna a otro usuario calificador el sistema lo descuenta del usuario calificador que inicialmente se asignó por reparto (en el rango de fechas).
- TURNOS CALIFICADOS (en el rango de fechas) Se tomará la información referente a la

resolución del turno y que los turnos hayan superado la fase firma del Registrador, el total de turnos calificados por cada calificador, es la sumatoria de los turnos Devueltos al Público, Inscritos e Inscritos Parcialmente, la sumatoria de estas tres columnas es el resultado de la columna mencionada.

- **TURNOS INSCRITOS:** Es el total de turnos calificados, seleccionando cuales fueron inscritos.
- **TURNOS DEVUELTO AL PUBLICO SIN REGISTRAR:** Es el total de turnos calificados seleccionando cuales fueron devueltos al público sin registrar.
- **TURNOS INSCRITOS PARCIALMENTE:** Es el total de turnos calificados, seleccionando cuales fueron inscritos parcialmente.
- **TURNOS PENDIENTES POR CALIFICAR:** Se totalizan aquí los turnos en el rango de fecha seleccionado, teniendo en cuenta los turnos que se encuentran en la fase calificación, también se incluyen los turnos que se encuentren en cualquiera de las fases del proceso de Mayor Valor, corrección de encabezado, confrontación correctiva. No se contaran los turnos que en la fase de calificación son redireccionados a Antiguo Sistema y Correcciones. Con el fin de evitar duplicidad de información, ya que estos se contaran en el ítem “TURNOS REDIRECCIONADOS A.S Y CORRECCIONES”.
- **TURNOS REDIRECCIONADOS A ANTIGUO SISTEMA Y CORRECCIONES:** Son los turnos que se encuentran en cualquiera de las fases del subproceso de Antiguo Sistema y Correcciones.
- **TURNOS EN TRAMITE:** Refleja el total de los turnos que se encuentran en la fase Mesa de control y Firma.

#### RESUMEN:

En caso de seleccionar más de un CALIFICADOR , el reporte debe tener al final un resumen así:

- **TOTAL TURNOS REPARTIDOS:** Es la sumatoria de todos los turnos repartidos y reasignados a los calificadores en el rango de fecha seleccionado.
- **TOTAL TURNOS CALIFICADOS** Es la sumatoria de todos los turnos calificados por todos los calificadores en el rango de fecha seleccionado, mostrando que el total de turnos calificados es la sumatoria de turnos inscritos, Devueltos al Público e Inscritos Parcialmente
- **TOTAL TURNOS INSCRITOS**
- **TOTAL TURNOS DEVUELTO AL PUBLICO:** Es la sumatoria de todos los turnos devueltos al público sin registrar por todos los calificadores en el rango de fecha seleccionado.
- **TOTAL TURNOS INSCRITOS PARCIALMENTE:** Es la sumatoria de todos los turnos inscritos parcialmente por todos los calificadores en el rango de fecha seleccionado.
- **TOTAL TURNOS EN TRAMITE:** Es la suma todos los turnos que se encuentran en la fase Mesa de Control y Firma del Registrador.
- **TOTAL TURNOS REDIRECCIONADOS A ANTIGUO SISTEMA Y CORRECCIONES**

**- TOTAL TURNOS PENDIENTES POR CALIFICAR.**

Se realiza la modificación del nombre del reporte en el listado de reportes y en el formulario donde se ingresan los parámetros, por el texto “INFORME ESTADISTICO DE TURNOS POR CALIFICADOR”

**008\_589 - Formas\_de\_Pago**

La Superintendencia de Notariado y Registro esta implementado con varias entidades bancarias, el pago de derechos de registro (documentos, certificados, mayores valores, consultas, fotocopias) con medios electrónicos, de manera que los usuarios tenga la facilidad de realizar sus pagos por diferentes medios.

Se realizan las modificaciones en el aplicativo SIR, para que se permita realizar el pago de derechos de registro por diferentes formas de pago como:

- EFECTIVO
- CONSIGNACION
- CHEQUE
- CONVENIO
- PAGO ELECTRONICO PSE
- TARJETA CREDITO
- TARJETA DEBITO
- TIMBRE.

Así mismo se hacen los ajustes en los siguientes Reportes que detallan los Recaudos de Derechos de Registro.

- Reporte 05: Diario mayores valores y devoluciones del dia
- Reporte 06: Diario radicador de certificados del día
- Reporte 08: Diario radicador de consultas
- Reporte 25: Informe recaudos del dia (tiracaja)
- Reporte 51: Resumen diario de ingresos y egresos
- Reporte 74: Informe de ingresos por certificados
- Reporte 99: Ingresos diarios por documentos
- Reporte 156: Diario Anulados del día
- Reporte E02: Estadísticas Generales

**020\_589\_Código\_Notaria\_NC**

Se modifica la estructura de la tabla SIR\_NE\_OFICINA\_ORIGEN, adicionando el campo ID\_UNICO de tipo alfanumérico y de longitud de 20 caracteres. Así, mismo se estandariza y actualiza la información, de acuerdo al archivo en formato Excel enviado, con los datos de las notarías existentes a nivel nacional, así mismo se efectúa el respectivo versionamiento sobre este objeto, manteniendo el histórico y la integridad referencial, de la información inmobiliaria en la base de datos de SIR.

También se aplica el versionamiento al código único de identificación, que corresponde al CIRCULO REGISTRAL de las oficinas de registro.

**021\_589\_Rol\_Registrador\_Inhabilitar\_Pantallas\_Administrativas**

Los Roles asignados a los Registradores de Instrumentos Públicos o a los funcionarios de las ORIPS y/o SNR, se efectúa paralelamente a la asignación de pantallas administrativas inherentes a cada uno de esos roles, que al igual tienen funciones operativas de responsabilidad administrativa y jurídica en el sistema.

Se realiza el ajuste para que una vez se inactive algún rol de los asignados al registrador o a algún funcionario, igualmente se inhabiliten las pantallas administrativas relacionadas con el rol principal.

#### **024\_589\_Reanotación\_Turnos\_Registro**

Cuando se trate de turnos de Registro: Radicado un turno de registro en cualquier ORIP que desarrolle actividades registrales con el Aplicativo SIR, una vez que estos hayan superado la fase de firma del Registrador, con tipo de calificación inscrito e inscrito parcialmente, el sistema no debe permitir reanotar dichos turnos, cuando el usuario intente reanotar un turno de registro que no cumpla con esta condición debe mostrar mensaje “Este turno no puede ser reanotado, supero la fase Firma de Registro”

Los turnos de Registro con tipo de calificación Devuelto al Público que superen la Fase de Firma del Registrador y hasta la fase de Entrega, pueden ser reanotados, obligando a incluir la nota informativa con un texto mínimo de 30 caracteres, para justificar la reanotación.

Los documentos reanotados deben ser asignados nuevamente al usuario calificador al cual le había sido repartido inicialmente si este se encuentra activo, en el caso que el usuario calificador se encuentre inactivo el sistema debe permitir seleccionar de la lista de calificadores un usuario diferente a este para asignar el turno.

Al Reanotarse un turno de Registro con Resolución Devuelto al Público y del cual se ha generado Nota Devolutiva y este pasa a la fase de calificación, el aplicativo debe guardar el Histórico de la Nota Devolutiva y permitir la calificación del turno regstral o la generación de una nueva Nota Devolutiva, reflejando y guardando la respectiva trazabilidad en el sistema sobre las acciones.

El sistema debe validar que las matrículas asociadas al turno objeto de reanotación, no hayan tenido trámite con turnos posteriores de inscripciones o Devoluciones, no debe permitir la reanotación.

El sistema validara en el caso de que no se hayan expedido certificados de tradición sobre dichas matrículas, el sistema verificará:

- a. Si se han radicado turnos del proceso (1) certificados inmediatos posteriores, no se permitirá la reanotación del turno de registro.
- b. Para turnos del proceso (6) con acto “Embargo” resolución “Devuelto al público” y con turno de certificado asociado expedido, el sistema no permitirá su reanotación.

Los turnos reanotados deben tomar las validaciones que se le efectúen pero conservando el historial anterior, guardando los usuarios intervenientes, fecha y hora en cada una de las acciones realizadas, las cuales se reflejarán en las fases del historial del turno.

Como la finalidad de la reanotación es la de enmendar errores del Tipo de calificación del documento y no la corrección de otros errores presentados en la misma, el sistema al desplegar la opción de reanotar debe tener un mensaje al respecto., así ”ESTA OPCION ES PARA ENMENDAR ERRORES DEL TIPO DE CALIFICACION DEL DOCUMENTO (Devuelto al Público)

En el caso de los testamentos se debe permitir su reanotación llevando el turno a la fase de Testamentos, para asignarlo al usuario que inicialmente lo negó, de lo contrario podrá seleccionar de la lista de usuarios de testamentos.

Para ningún turno de documento o de testamento, Devuelto al Público, Inscrito o Inscrito parcialmente, debe permitirse su reanotación si ya se encuentra Entregado o Finalizado.

La opción de REANOTAR debe ser inherente únicamente al ROL REGISTRADOR Y COORDINADOR JURIDICO.

**025\_589\_Anulación\_Habilitacion\_Historial\_Turnos**

Con el objetivo que el Aplicativo SIR refleje la información veraz respecto de los usuarios del sistema que realizan acciones dentro del mismo se realiza el ajusta para que cuando se anula un turno de cualquiera de los procesos registrales, el aplicativo debe mostrar en el historial de Fases del Turno, la fase, usuario, fecha y hora correcta, y así mismo guardar el historial de la información del funcionario, fecha y hora en que se llevó a cabo la anulación del turno.

Igualmente si se habilita o se restituye un turno, el sistema debe reflejar la información correcta de acuerdo a los parámetros antes señalado.

**DESPLIEGUE – 3.1.0**

26/07/2013

**056\_453\_Modificación\_de\_Naturaleza\_Juridica**

Mediante memorando SNR2012IE013178 de septiembre 21 de 2012, el Superintendente Delegado para el Registro, doctor Enrique José Nates Guerra solicita incluir en el Aplicativo SIR, el Código de Naturaleza jurídica 0476 “Embargo en Proceso de Reorganización contenido en la Ley 1116” así mismo solicita la revisión de los códigos en el Aplicativo.

Con el fin de atender la solicitud, en reunión de la parte funcional y técnica con la Doctora María Clemencia Rangel, Coordinadora Grupo de Gestión Jurídica, Inspección y Vigilancia de las Orips, se preciso el objeto de las resoluciones 6992 de 2009, 3906 y 4788 de 2010, para realizar los ajustes de los Actos de Naturaleza Jurídica en el Aplicativo SIR

Se solicita actualizar la tabla SIR\_NE\_NATURALEZA\_JURIDICA, para dar aplicabilidad a las resoluciones 6992 de 2009, 3906 y 4788 de 2010 así:

CODIGO	DESCRIPCIÓN
0357	DECLARACION DE RESERVA, ALINDERACION Y CREACION DE AREA DEL SISTEMA DE PARQUES NACIONALES Y NATURALES
0358	SERVIDUMBRE LEGAL DE HIDROCARBUROS
0476	EMBARGO PROCESO DE REORGANIZACION LEY 1116 DE 2006
0477	DEMANDA PROCESO ESPECIAL DE SANEAMIENTO DE LA TITULACION DE LA PROPIEDAD INMUEBLE ART. 8 DE LA LEY 1182 DE 2008.

Se requiere crear un versionamiento para guardar los cambios realizados sobre la tabla en mención.

**060\_453\_Ajuste\_Campo\_de\_Linderos\_en\_SIR**

Para dar cumplimiento a la normatividad vigente, se solicita realizar los siguientes ajustes en el Aplicativo SIR :

- En la creación de Folios de Matricula inmobiliaria, mediante la funcionalidad de segregar y englobar en los procesos de Registro y Correcciones, se debe ajustar el texto del campo de linderos, el cual es conformado por el sistema con los datos del documento objeto de registro, el nombre del predio, área, coeficiente y la normatividad de los linderos , así :

- De este texto se debe suprimir “Artículo 11 del Decreto 1711 de 1984” y agregar “Articulo 8 Parágrafo 1º. de la Ley 1579 de 2012.

**EJEMPLO DEL TEXTO QUE EL SISTEMA DEBE CONFORMAR:**

“APTO 201 CON AREA DE 100 MTRS2 COEFICIENTE DE PROPIEDAD 50% CUYOS LINDEROS Y DEMAS ESPECIFICACIONES OBRAN EN ESCRITURA 01 DE FECHA 08/10/2001 NOTARIA PRIMERA DE BOGOTA D. C, ART. 8 PARÁGRAFO 1º.DE LA LEY 1579 DE 2012”

- El texto del campo de linderos, también debe conformarlo el sistema en las funcionalidades :
  - Crear Folio, del proceso de calificación.

Cuando se trate de creación de matrículas, generadas por inscripciones que contengan los siguientes actos de códigos de Naturaleza Jurídica, es obligatorio que a continuación del texto que conforma el sistema en el campo linderos, se realice la transcripción de los linderos de conformidad con el artículo

8° de la Ley 1579 de 2012.

0101	ACCESION DEL SUELO
0103	ADJUDICACION BALDIOS
0104	ADJUDICACION BALDIOS EN PROPIEDAD COLECTIVA A COMUNIDADES NEGRAS
0105	ADJUDICACION BALDIOS RESGUARDOS INDIGENAS
0106	ADJUDICACION DE BIENES VACANTES
0109	ADJUDICACION EN SUCESION
0110	ADJUDICACION LIQUIDACION DE LA COMUNIDAD
0111	ADJUDICACION LIQUIDACION SOCIEDAD COMERCIAL
0112	ADJUDICACION LIQUIDACION SOCIEDAD CONYUGAL
0113	ADJUDICACION LIQUIDACION SOCIEDAD PATRIMONIAL DE HECHO
0114	ADJUDICACION POR EXPROPIACION
0115	ADJUDICACION SUCESION PARTICION ADICIONAL
0116	ADJUDICACION UNIDAD AGRICOLA FAMILIAR
0118	APORTE A SOCIEDAD
0119	APORTE DE SUBSIDIO EN ESPECIE
0121	CESION A TITULO GRATUITO DE BIENES FISCALES
0122	CESION DE BIENES OBLIGATORIA
0124	CESION OBLIGATORIA DE ZONAS CON DESTINO A USO PUBLICO
0126	COMPRAVENTA PARCIAL
0128	CONSTITUCION DE FIDUCIA MERCANTIL
0129	DACION EN PAGO
0130	DACION EN PAGO OBLIGATORIA
0131	DECLARACION JUDICIAL DE PERTENENCIA
0133	DECLARATORIA DE PROPIEDAD PUBLICA SOBRE ZONAS DE CESION OBLIGATORIA GRATUITA
0138	DONACION
0140	EXPROPIACION POR VIA ADMNISTRATIVA
0141	EXPROPIACION POR VIA JUDICIAL
0145	PRESCRIPCION AGRARIA
0162	TITULO DE MINAS CON ANTECEDENTE REGISTRAL
0163	TRANSACCION
0165	TRANSFERENCIA DE DOMINIO POR SOLUCION O PAGO EFECTIVO
0166	ADJUDICACION TERRENO EJIDO
0167	ADJUDICACION LIQUIDACION SOCIEDAD DE HECHO
0172	TRANSFERENCIA DE DOMINIO DE BIENES FISCALES
0173	SANEAMIENTO TITULACION LEY No. 1182 DE 2008
0175	CESION DE BALDIO URBANO DE LA NACION AL MUNICIPIO. INSTRUCCIÓN ADMINISTRATIVA No. 18 DE 2009 (ART. 123 DE LA LEY 388 DE 1997)
0176	FORMALIZACION DEL TITULO DE PROPIEDAD ARTICULO 139 DE LA LEY 1151 DE 2007
0313	CONSTITUCION DE FIDEICOMISO CIVIL
0321	DESLINDE Y AMOJONAMIENTO
0903	ACTUALIZACION DE LINDEROS
0909	CONSTITUCION DE URBANIZACION
0913	DECLARACION PARTE RESTANTE
0915	DESENGLOBE
0916	DESLINDE DE TERRENO DE PROPIEDAD DE LA NACION
0917	DETERMINACION AREA Y LINDEROS PREDIO DEL MUNICIPIO
0918	DIVISION MATERIAL
0919	ENGLOBE
0920	LOTEO
0924	RELOTEO

- El sistema debe obligar a incluir la información de los linderos, en las matriculas nuevas, mínimo cien (100) caracteres, a continuación del texto pre-establecido por el sistema y de lo contrario debe mostrar un mensaje restrictivo indicando que se ha omitido diligenciar el campo de la información de los linderos.

**Esta validación debe ejecutarse al avanzar:**

Turnos de Registro:

- a) Del proceso de Digitador a Calificación
- b) En la Creación de Folios en la fase de Calificación botón “CREAR FOLIOS”  
En la primera pantalla de esta funcionalidad al dar clic en el botón “Siguiente”  
El sistema debe validar que se ha ingresado la información de los linderos, de lo contrario mostrar el mensaje restrictivo.
- c) De la fase Calificación a Mesa de Control.

Subproceso Antiguo Sistema

- d) De la fase Hoja de Ruta a Revisión final, para los turnos de registro, certificados de pertenencia, antiguo sistema, correcciones, registro con base a los cuales se ha digitado hoja de ruta para creación de matrícula, no se generara el texto que conforma el sistema mencionado inicialmente a este requerimiento, pero si debe validar el sistema que se ha incluido información mayor a 200 caracteres.

Turnos de Corrección

- a) Del proceso Digitador a la fase Ejecutar corrección
- b) De la fase Ejecutar Corrección a la fase Revisar y Aprobar

Creación Directa de Folios - Pantallas Administrativas

En la primera pantalla de la Creación Directa de Folios al dar clic en el botón “Siguiente” el sistema debe validar que se ha ingresado la información de los linderos , de lo contrario debe mostrar mensaje restrictivo “no ha ingresado la información de los linderos”

- **Respecto a la excepción,** “Solo se podrá omitir la transcripción de los linderos en el folio de matrícula inmobiliaria, para las unidades privadas derivadas de la inscripción del régimen de propiedad horizontal”

Cuando se trate de creación de matrículas, generadas por inscripciones que contienen los siguientes actos de códigos de Naturaleza Jurídica, es **opcional** agregar la información de los linderos a continuación del texto que conforma el sistema en el campo linderos:

“Actos en los cuales no es obligatorio transcribir los linderos pero que el sistema permita consignarlos si los calificadores o registradores lo consideran necesario. “

0102	ACTA ENTREGA DE INMUEBLES A ENTIDAD
0107	ADJUDICACION DE LA COSA HIPOTECADA
0108	ADJUDICACION EN REMATE
0117	ADQUISICION POR ABSORCION DE ACCIONES Y PATRIMONIO DE ENTIDADES FINANCIERAS
0120	CADUCIDAD ADMINISTRATIVA
0125	COMPRAVENTA
0127	CONSOLIDACION DE DOMINIO PLENO
0139	ESCISION
0143	FUSION
0144	PERMUTA
0154	RESTITUCION EN FIDEICOMISO CIVIL
0155	RESTITUCION EN FIDUCIA MERCANTIL
0164	TRANSFERENCIA DE DOMINIO A TITULO DE BENEFICIO EN FIDUCIA MERCANTIL

0168	TRANSFERENCIA DE DOMINIO A TITULO DE LEASING HABITACIONAL DE VIVIENDA FAMILIAR
0169	TRANSFERENCIA DE DOMINIO A TITULO DE LEASING HABITACIONAL DE VIVIENDA NO FAMILIAR
0170	TITULACION POR SANEAMIENTO CONTABLE
0181	ADJUDICACIÓN EN PROCESO DE LIQUIDACIÓN JUDICIAL ART. 58 LEY 1116 DE 2006
0301	ADJUDICACION SUCESION DERECHO DE CUOTA
0302	ADJUDICACION SUCESION NUDA PROPIEDAD
0303	ADJUDICACION SUCESION SUBSUELO CON ANTECEDENTE REGISTRAL
0307	COMPRAVENTA DERECHOS DE CUOTA
0308	COMPRAVENTA NUDA PROPIEDAD
0309	COMPRAVENTA SUBSUELO CON ANTECEDENTE REGISTRAL
0310	COMPRAVENTA USUFRUCTO
0316	CONSTITUCION REGIMEN DE CONDOMINIO
0317	CONSTITUCION REGLAMENTO DE PROPIEDAD HORIZONTAL
0322	DONACION NUDA PROPIEDAD
0330	REFORMA REGIMEN DE CONDOMINIO
0331	REFORMA REGLAMENTO DE PROPIEDAD HORIZONTAL
0332	REMAT DERECHO DE CUOTA
0346	ADICION REGIMEN DE CONDOMINIO
0347	ADICION REGIMEN DE PROPIEDAD HORIZONTAL
0348	ADJUDICACION LIQUIDACION DE LA SOCIEDAD CONYUGAL DERECHO DE CUOTA
0349	ADJUDICACION LIQUIDACION DE LA SOCIEDAD CONYUGAL NUDA PROPIEDAD
0357	DECLARACION DE RESERVA, ALINDERACION Y CREACION DE AREA DEL SISTEMA DE PARQUES NACIONALES Y NATURALES
0359	DECLARATORIA DE BIEN INMUEBLE DE INTERES CULTURAL NUMERAL 1.2. ART. 7 DE LA LEY 1185 DE 2008
0360	DECLARATORIA DE BIEN EXISTENCIA DEL PLAN ESPECIAL DE MANEJO Y PROTECCION NUMERAL 1.2. ART. 7 DE LA LEY 1185 DE 2008
0834	EXTINCION PARCIAL REGIMEN DE CONDOMINIO
0835	EXTINCION PARCIAL REGIMEN DE PROPIEDAD HORIZONTAL
0836	EXTINCION TOTAL REGIMEN DE CONDOMINIO
0837	EXTINCION TOTAL REGIMEN DE PROPIEDAD HORIZONTAL
0907	CAMBIO DE RAZON SOCIAL

Al avanzar turnos de registro de la fase Calificación a Mesa de Control, con matrículas creadas con base a este tipo turnos, , el sistema debe mostrar mensaje de confirmación “ no ha incluido la información de linderos para las matriculas nuevas desea avanzar el turno “sí” o no” -- al seleccionar “no” el turno no debe avanzar a la siguiente fase , al seleccionar “sí” el turno debe avanzar a la fase mesa de control.

Así mismo debe hacer validación y confirmación al avanzar turnos de corrección de la fase Ejecutar corrección a la fase Revisar y Aprobar con base a los cuales se ha creado matriculas con los códigos de naturaleza jurídica que es opcional la transcripción de linderos.

### DESPLIEGUE – 3.0.0

01/06/2013

#### 023\_453\_Traslado\_Masivo\_Folios

Se realiza el ajuste para que el sistema permita realizar el traslado masivo de Folios de Matrícula Inmobiliaria, que geográficamente estén ubicados en un municipio determinado de un Círculo Registral y deberán ser trasladados a otro Círculo Registral, independiente del aplicativo con el que operen las ORIP's involucradas, es decir, SIR o Folio Magnético.

#### **Descripción:**

Se crea el ROL Traslado Masivo de Folios, el cual será asignado al Administrador Nacional y se ejecutará solamente con autorización escrita del jefe de la Oficina de Informática y Director de Desarrollo Registral, de la SNR. Este ROL permitirá trasladar una cantidad indeterminada de Folios, de una Oficina Origen SIR a una Oficina Destino SIR o Folio, se deben aplicar las condiciones y

validaciones existentes en la opción de pantalla “Administración Traslado de Folios” y que esta funcionalidad además permita lo siguiente:

- Al seleccionar el Rol Traslado Masivo de Folios el sistema debe mostrar una lista retráctil de los Municipios correspondientes al Círculo Registral Origen y permita seleccionar el municipio de donde se requiere trasladar los folios. El sistema debe desplegar los siguientes campos:

Círculo Registral Origen: Lista de valores, con los círculos registrales de las ORIPs que operan con SIR.

Departamento: Lista de valores del o departamentos asociados al círculo registral origen seleccionado.

Municipio: Lista de valores de los municipios, asociados al círculo registral origen seleccionado y departamento seleccionado.

Se incluye un botón que permite realizar la búsqueda del archivo plano depurado, con la información de los folios a trasladar del municipio seleccionado. Se aplica la misma funcionalidad de la pantalla administrativa de “Administración Firmas del Registrador”. Al momento de seleccionar el archivo depurado de los folios a trasladar, debe indicarse en un campo no editable el nombre del archivo seleccionado.

Una vez completada la información antes descrita, existe un botón que permite cargar o desplegar en la misma pantalla solo los números de matriculas contenidos en el archivo depurado.

Al final de la lista de matriculas a trasladar, se incluyen los campos correspondientes para seleccionar el círculo Registral Destino, el Departamento Destino y el Municipio Destino y realizar el respectivo cambio de círculo y/o Municipio, incluidas sus veredas, siempre y cuando el traslado se realice entre ORIPs que operan con SIR. Si el municipio y/o veredas no existen para el círculo registral destino, con previa autorización el sistema creara estos registros en las tablas correspondientes, y sus respectivas zonas registrales.

En esta pantalla también se incluye lo siguiente:

- Un texto “Resolución” y un campo de texto para agregar el número de Resolución que ordena el traslado.
- Un texto “Fecha” y un campo que permite agregar la fecha de la resolución .

Una vez diligenciados los campos antes descritos, debe existir un botón que permita efectuar el respectivo traslado masivo de folios del circulo registral origen al círculo registral destino. Al presionar este botón, el sistema despliega un mensaje en el que se incluya círculo, departamento y municipio destino y que pregunte al usuario si está seguro o no de realizar este traslado. En caso negativo, debe mantenerse en esta pantalla para que verifique nuevamente la información a trasladar.

Una vez finalizado el proceso de traslado masivo, el archivo con el cual se efectuó el cargue de las matriculas, será guardado en el sistema (directorio definido en el servidor de aplicaciones para guardar los archivos de traslados validos) en el formato establecido y con la siguiente estructura: id del círculo, id del municipio, fecha de generación y al final incluir el texto ORIP.

Así mismo, al finalizar el proceso de traslado masivo de folios, el sistema debe desplegar para cada matricula origen su correspondiente matricula destino. Así mismo, debe indicar que matriculas no fueron posibles de trasladar y la razón o motivo de esta situación.

- Se implementa una nueva opción o pantalla administrativa denominada “Generación archivo – Traslado Masivo de Folios”, que permite la generación de un archivo plano, para que el funcionario responsable del traslado de folios masivamente, verifique, valide, seleccione y depure sobre este archivo, si los folios a trasladar cumplen con los requerimientos para tal fin.

Previamente, debe parametrizarse el directorio en el cual se van a guardar los archivos generados para realizar el proceso de traslado masivo, en el servidor de aplicaciones del SIR.

Podría aplicarse el mismo procedimiento que aplica para las firmas de los registradores de las ORIPs, que actualmente operan con SIR.

Esta pantalla administrativa, debe mostrar una lista retráctil de los Municipios correspondientes al Círculo Registral Origen y permita seleccionar el municipio de donde se requiere trasladar los folios.

El archivo plano que se generara, de acuerdo a los parámetros ingresados, debe contener las siguientes columnas:

1. Número de matrícula inmobiliaria,
2. Número de anotaciones existentes,
3. Id y nombre de municipio
4. Id y nombre de vereda
5. Estado del folio (activo, cerrado, inconsistente, entre otros)
6. Turnos en trámites

El archivo resultante, se nombrará con la siguiente estructura: id del círculo, id del municipio y fecha de generación y definirse el formato en el que se grabara esta información. Ejemplo: Se trasladaran los folios del circulo registral de Dabeiba (007) del municipio de Chigorodó (172) el día 22 de mayo del año en curso, el archivo debe ser nombrado así: 00717222052012.

Este archivo permite ser exportado a una hoja de cálculo (excel u otra), para facilitar la labor de revisión, validación y depuración por parte del funcionario responsable de los trasladados.

Una vez realizada la labor de verificación por parte del funcionario, el archivo depurado sera cargado para iniciar el proceso de traslado masivo de folios, manteniendo las mismas columnas con las cuales se genero inicialmente el archivo de matrículas a trasladar; por lo tanto, esta información debe ser guardada con la siguiente estructura: id del círculo, id del municipio, fecha de generación y al final incluir el texto ORIP y definir el formato en el que se grabara esta información.

Ejemplo: Se revisaron los folios a trasladar del circulo registral de Dabeiba (007) del municipio de Chigorodó (172) el día 22 de mayo del año en curso, el archivo debe ser nombrado así: 00717222052012-ORIP.

- Una vez seleccionadas una, varias o todas las matrículas requeridas para el traslado, el sistema debe permitir seleccionar el Círculo Registral Destino y el Municipio Destino y realizar el respectivo cambio de Círculo y/o Municipio, incluidas sus veredas, siempre y cuando el traslado se realice entre ORIP's que operan con SIR. Si el municipio y/o veredas no existen para el círculo registral destino, el sistema deber crear estos registro en las tablas correspondientes, y sus respectivas zonas registrales.
- Se permite seleccionar si el traslado se realiza a una Orip que opera con Folio o con SIR.

- En la pantalla de traslado masivo de folios, el aplicativo debe incluir lo siguiente:

- Un texto “Resolución” y un campo de texto para agregar el número de Resolución que ordena el traslado.

- Un texto “Fecha” y un campo que permite agregar la fecha de la resolución.

- Si el traslado es a una ORIP que opera con FOLIO, el sistema debe dejar un log histórico de la información trasladada, para lo cual se requiere que toda la información de uno o varios folios trasladados sean guardados en un formato XML. La información que actualmente existe en la tabla SIR\_NE\_TRASLADO, con relación al formato XML, debe modificarse en los siguientes campos:

- Incluir código del departamento
- Incluir código del municipio
- Incluir código de la vereda
- Incluir la información detallada del documento registrado (SIR\_NE\_DOCUMENTO), para la data correspondiente al encabezado de la matrícula trasladada.
- Incluir información de lo contenido en el campo relacionado con la complementación.
- Incluir la información detallada del documento registrado (SIR\_NE\_DOCUMENTO), para la data correspondiente a las anotaciones de la matrícula trasladada.
- Incluir la información detallada de los ciudadanos que intervienen en cada anotación de la matrícula inmobiliaria trasladada (SIR\_NE\_CIUDADANO).

Se implementa un WebService editable y consumible, que permita tomar la información de los folios trasladados a ORIP's que operan con FOLIO MAGNÉTICO, con el fin de realizar el cargue de estas matrículas masivamente; esto basado en los XMLs generados.

Así mismo debe desplegar todas las matrículas que están pendientes de confirmar el traslado a la ORIP destino, incluyendo las siguientes columnas:

- Consecutivo.
- Número de Matrícula que corresponde al consecutivo de matrículas de la ORIP Destino.
- Número de Anotaciones.
- Código (id) y nombre del Departamento.
- Código (id) y nombre del Municipio
- Código (id) y nombre de la Vereda.

Debe ordenar y agrupar las matrículas por estos ítems departamento, municipio y vereda.

El sistema permite consultar cada una de las matrículas. Al consultar los detalles del folio traslado en la Oficina Origen y la nueva en la Orip Destino, el sistema debe mostrar la información del traslado (como lo refleja en la opción de Administración de Traslado de Folios, resolución, fecha, círculo del cual fue trasladado, etc.) y en el campo “ESTADO DEL FOLIO” el sistema debe mostrar el mensaje “FOLIO TRASLADADO AL CÍRCULO XXX” y la fecha y hora en que fue trasladada.

El sistema debe permitir marcar una, varias o todas las matrículas que requieren la confirmación. Al final de la lista de matrículas, debe existir un campo para indicar la resolución y fecha de la misma, por la cual se incluyen las nuevas matrículas al círculo registral destino. El sistema debe validar que exista como mínimo una matrícula marcada. Debe permitir de manera masiva modificar de ser necesario, el departamento, municipio o vereda.

El botón de CONFIRMAR debe ir al final de la lista de matrículas y no frente de cada una de ellas, y desplegar un mensaje que pregunte al usuario si está seguro de aprobar este traslado.

El aplicativo SIR debe permitir generar Reportes Estadísticos de Traslado Masivo de Folios de Matrícula Inmobiliaria, así:

1) Reporte – Informe Estadístico de TRASLADOS MASIVO DE FOLIOS A OTROS CÍRCULOS – este reporte es para la Orip Origen – Tiene en cuenta las generalidades del encabezado de los reportes.

El sistema recibe como parámetros un rango de fechas y despliega la siguiente información: Fecha de traslado, id y nombre del círculo Registral Destino, id y nombre del Municipio, id y nombre del usuario que ejecutó el traslado y cantidad de matrículas trasladadas a ese círculo y Municipio, número de la resolución que ordena el traslado con su respectiva fecha. Esta información se agrupa por los ítems antes mencionados.

El reporte muestra un gran total de las matrículas trasladadas.

2) Reporte: INFORME Estadístico TRASLADO MASIVO DE FOLIOS A ESTE CÍRCULO: este reporte es para la Orip Destino. Tener en cuenta las generalidades del encabezado de los reportes.

El sistema recibe como parámetros un rango de fechas y despliega la siguiente información: Fecha de confirmación del traslado, id y Nombre del Círculo Registral Origen, id y Nombre del Municipio, id y nombre del usuario que confirma el traslado y cantidad de matrículas trasladadas a ese círculo y municipio, número de la resolución de confirmación con su respectiva fecha. Esta información debe agruparse por los ítems antes mencionados. El Reporte muestra un gran total de las matrículas trasladadas.

El Reporte “-163- Matrículas Trasladadas a este Círculo”, además de reflejar las matrículas trasladadas individualmente, refleja la información de las matrículas trasladadas por el proceso de traslado Masivo.

Se deben aplicar las condiciones y validaciones existentes en la opción de pantalla “Administración traslado de Folios”

#### **061\_453\_Requerimiento\_Auditoria.doc**

Teniendo en cuenta que actualmente el Sistema de Información Registral SIR, debe ajustar el control de auditoria, se requiere implementar un esquema de base de datos denominado AUDITORIA\_SIR el cual permitirá ejercer control y seguimiento sobre las tablas del negocio, dichos objetos a los cuales se les debe realizar seguimiento se describen a continuación:

SIR\_NE\_ANOTACION  
SIR\_NE\_ANOTACION\_CIUDADANO  
SIR\_NE\_AUDITORIA  
SIR\_NE\_SALVEDAD\_AN  
SIR\_NE\_DIRECCION  
SIR\_NE\_FOLIO\_DERIVADO  
SIR\_NE\_COMPLEMENTACION  
SIR\_NE\_DOCUMENTO  
SIR\_NE\_FOLIO

SIR_NE_REL_CANCELACION
SIR_NE_CIUDADANO
SIR_OP_MINUTA
SIR_OP_TURNO_HISTORIA
SIR_REL_BLOQUEO_FOLIO
SIR_NE_SALVEDAD_FO
SIR_NE_NATURALEZA_JURIDICA
SIR_OP_USUARIOS

Además se debe incluir en el esquema AUDITORIA\_SIR el esquema de auriga (roles, estaciones y usuario), tener en cuenta la activación, desactivación, modificación y/o eliminación de triggers en las bases de datos.

Es necesario que dentro del esquema de Auditoria se incorpore el histórico de usuarios y la Oficina ( traza sobre activación o desactivación de usuarios, histórico de traslado de usuarios, asignación o retiro de roles de los usuarios creados en SIR).

Específicamente por cada una de las tablas mencionadas anteriormente, se deben guardar registros cuando se ejecute una actualización o borrado.

Para cada campo de las tablas mencionadas en el ítem anterior, se deben crear dos nuevos campos: uno con el valor actual y otro con el valor nuevo (como estaba y como quedó). Adicional a esto se debe contar con información complementaria como:

CAMPO	DESCRIPCION
USERNAME	Nombre de usuario para conectarse al aplicativo (esquema base de datos)
OSUSER	Usuario Sistema Operativo
TERMINAL	Nombre del computador donde se conecta el usuario
PROGRAM etc)	Mediante qué programa/aplicación ingresó (sqlplus, toad, Enterprise manager,
LOGON_TIME	fecha y hora en la cual se conecta al aplicativo
INSTANCE_ID	Identificación de la instancia la base de datos
IP_TERMINAL	Dirección ip de la cual se está conectando

Se crearán Triggers por cada objeto donde se controlen las operaciones de delete y update. Así como también un paquete el cual se encargará de registrar la operación realizada en cada tabla auditada, esto con el fin de utilizar un solo punto de entrada para la información a auditar.

Además se requiere que exista un ROL AUDITOR, el cual debe estar asociado a un usuario que le permita conectarse al esquema AUDITORIA\_SIR, el cual contará como mínimo con un MENU correspondiente a :

### **CONSULTA**

Consulta de radicaciones por rango de fecha (diario radicador)  
 Consulta de reasignación de turno  
 Consulta de reimpresos

### **REPORTES:**

Reporte de Folios creados en una fecha inicial a una fecha final

Reporte de cambios de matrículas con un mismo turno de certificados  
Reporte de turnos enviados a confrontación correctiva  
Reporte sobre correcciones de encabezado (datos del documento, No escritura, No de turno antes y después)  
Reporte de usuarios que tienen rol de impresión de certificados de tradición ya sea inmediatos, antiguo sistema, correcciones etc.  
Reporte de reimpresiones por rango de fecha donde indique el usuario que lo realizó fecha y hora.  
Reporte de usuarios con sus respectivos roles y estaciones en la Oficina de Registro, donde indique el estado del usuario, fecha de creación de los roles asignados a cada uno de ellos así como las fechas en las que fueron creadas las estaciones asociadas a éste.  
Reporte histórico de habilitación y deshabilitación de usuarios roles y estaciones.

## ADMINISTRACION

Permisos  
Cambio de clave

### DESPLIEGUE – 2.8.0

11/05/2013

#### 001\_453\_Caracteristicas Impresión certificados

Características Impresión certificados.

SIR-183

No se ha implementado un mecanismo de seguridad para los certificados de tradición que expiden las ORIPS. Ejemplo, código de barras, sello de agua, trama de seguridad adecuada, etc. (DR E - 1004)

#### SEGUIMIENTO ADA:

Especificar el mecanismo que se requiere para la implementación de mayor seguridad en los certificados de tradición.

#### OBSERVACIONES SNR:

En efecto se debe implementar el mecanismo solicitado, tener en cuenta el desarrollo que se está efectuando en la impresión de certificados por medio del BOTON DE PAGO, en el cual se imprime un código (alfanumérico), que después puede ser consultado en la página web de la SNR, que siendo consultado indica el número de matrícula inmobiliaria, el turno de expedición, fecha de expedición y numero de anotaciones que tiene el folio de matrícula.

#### Análisis/Solución:

Este requerimiento busca generar un mecanismo de seguridad que pueda ser impreso en el cuerpo del certificado de Tradición y Libertad. Para este requerimiento se ha diseñado una llave cifrada que se imprimirá en la misma línea que ocupa el PIN de los certificados expedidos por el sistema de pago en línea. Esta llave cifrada estará impresa en el cuerpo del certificado, y las entidades (o las ORIPs) que quieran verificar la autenticidad del documento lo podrán hacer usando un Servicio Web publicado sobre la plataforma SIR.

#### Solución:

Se implementó una llave compuesta y cifrada que se imprime de forma directa sobre el Certificado antes de la impresión de la firma del registrador. La llave imprime un conjunto de caracteres cifrados con el algoritmo Base64, y en su interior guarda datos de: Turno y solicitud. Para verificar la autenticidad de la llave impresa se tiene un servicio web <http://192.168.202.25:8190/WS-SIRClient/>, el cual puede ser usado internamente, o en internet, de acuerdo con las reglas que defina SNR. La solución puede ser programada para despliegue en la versión 3.0.0 de aplicación.

También se implementara un sistema de encriptamiento que generara un código que se colocara al

final de los certificados de tradición que expiden las ORIPS.

La veracidad del código podrá ser consultado a través de una página web que se conectara a un WebService que le enviara información referente a el número de matrícula inmobiliaria, el turno de expedición, fecha de expedición y numero de anotaciones que tiene el folio de matrícula.

Este requerimiento fue realizado en tres fases (dos primeras realizadas por el Ing Julio Alcázar y la tercera por el Ing. Ellery Robles):

- 1) Se realiza una encriptación de información y se muestra en el certificado.
- 2) Se construye un Webservice que posee un método que recibe la llave encriptada y devuelve 4 características: la matrícula del certificado, el ultimo Id de anotación de la matrícula, el turno que creó el certificado y la fecha de creación del certificado.

### **Creación Proyecto Web WS-SIR**

Este proyecto es encargado de publicar un WebService que posee un método que recibe una clave y verifica si esta es válida y pertenece a un certificado de tradición y libertad expedida en las ORIPs.

La especificación del WebServices recibe un String y entrega un List que contiene un objeto con 4 características: la matrícula del certificado, el ultimo Id de anotación de la matrícula, el turno que creó el certificado y la fecha de creación del certificado.

Tercera fase del requerimiento.

- **Creacion De Imagenes:**
  - WS-SIRClient.web.css.redmong.images (carpeta de imagenes).
  - WS-SIRClient.web.images.fondo\_snr.jpg.
- **Creacion de Css.**
  - WS-SIRClient.web.css.redmond.skin.css.
- **Se crearon clases nuevas en WS-SIRClient:**
  - WS-SIRClient.web.index.xhtml.
  - WS-SIRClient.Bean.verificar.java.

Se modificaron los siguientes archivos:

#### **007\_453\_Perdida de Folio - Turno de Registro (Informativo)**

Turno: 2010-260-6-10278

Motivo Inconsistencia: Con el turno 2010-260-6-10278 se creó en Calificación el folio 260-264005 el día 23/04/2010 04:33:52 PM. El turno se encuentra en la fase DESANOTADO, Resolución INSCRITO pero no tiene el folio asociado, se consulta el Historial de fases del turno y se observa que en ningún momento se deshizo cambios en el mismo. Al consultar el turno 2010-260-6-9468, el folio 260-142267 se segregó el día 26/04/2010 03:02:18 PM y el sistema creó el folio 260-264005 como segregado. Por favor verificar que sucedió con esta calificación. Se adjuntan formularios de Calificación.

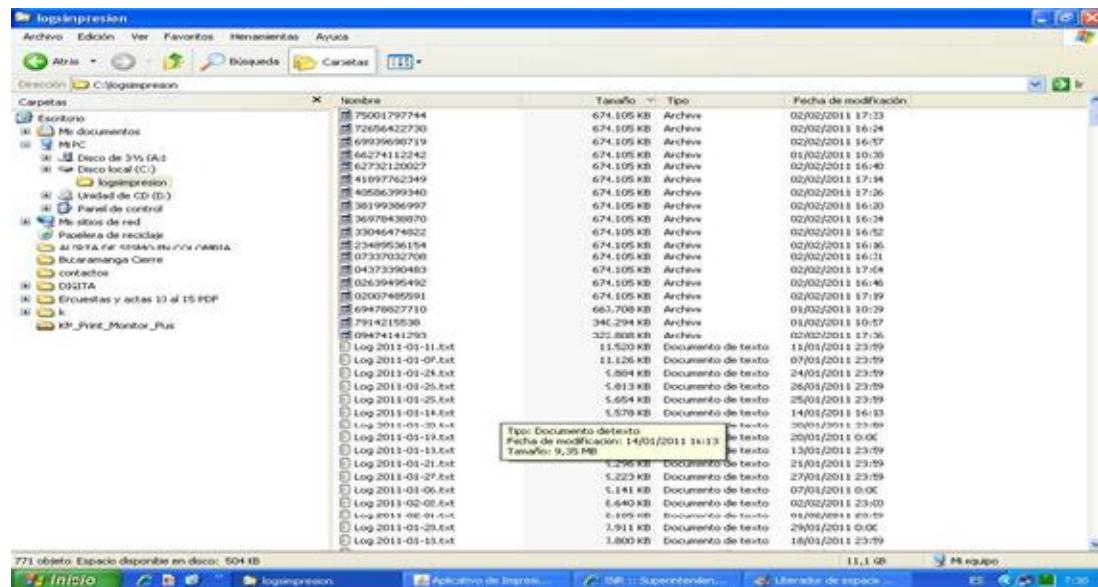
Grupo Cúcuta

#### **008\_453\_Logs de impresion**

Motivo Inconsistencia : Funcionario informa que observando la carpeta de "logsimpresion" del equipo donde se configura el "Aplicativo de Impresión" se observan archivos que ocupan gran parte del espacio del disco duro.

Favor verificar inconveniente ya que estos archivos de gran tamaño consumen bastante espacio en el disco duro y se generan diariamente, se adjunta imagen de carpeta de archivos.

Grupo Bucaramanga



### 009\_453\_Rol Usuario Operativo Certificados - Botón Enviar Caso

Rol Usuario Operativo Certificados. Actualmente existe el botón Enviar Caso



El cual enviaba anteriormente el turno de certificado de mayor extensión al rol Reimpresión Especial, rol que fue eliminado por solicitud de los requerimientos 012 y 246 para el despliegue 2.7.0. El botón “Enviar Caso” no tiene ninguna funcionalidad en el sistema y actualmente la reimpresión de certificados se realiza con el rol Reimpresión Certificados. Analizar la eliminación de este botón para que así mismo se realice la actualización del manual.

### 013\_453\_Error al devolver el turno\_MA

GLPI 30478

Turno:2011-088-6-1609

Motivo Inconsistencia:

Se comunica la usuaria Janeth Amezquita e informa que el turno 2011-088-6-1609 se encuentra en la fase de APROBAR CORRECCION CALIFICACION en la estación de X-JANET.AMEZQUITA ya al devolverlo a la fase de calificación el sistema indica que la operación se realiza satisfactoriamente, pero el turno se queda en la misma fase.

El turno 2011-088-6-1609 se encuentra en la fase de APROBAR CORRECCION CALIFICACION en la estación de X-JANET.AMEZQUITA y al devolverlo a la fase de calificación el sistema indica que

la operación fue se realizada satisfactoriamente tal como lo muestra la imagen adjunta pero el turno se queda en la misma fase, se verifica y el rol de calificación se encuentra activo con la usuaria JANTH AMEZQUITA, se requiere verificar el inconveniente para poder devolver el turno a calificación y seguir con el trámite correspondiente.

#### **016\_453\_Error al generar relaciones\_MA**

GLPI 30865

Motivo Inconsistencia: El usuario Gabriel Moros se comunica indica que al tratar de generar la relación 2011-260-4703 la página se está generando en blanco.

El usuario Gabriel Moros informa que al tratar de generar la relación 2011-260-4703 por la opción de reportes en pantallas administrativas la página queda en blanco sin generar información, se requiere verificar el inconveniente y corregir para poder generar la relación.

#### **025\_453\_Migración\_Folios\_Estado\_Traslado (Informativo)**

**Justificación:**

Poder realizar la migración de folios con estado trasladado.

**Descripción:**

Se solicita la adición del estado TRASLADADO incluido en la tabla MATRÍCULAS, en los ETL de migración, de tal forma que al realizar la migración de una nueva oficina los folios que estén con este estado sean migrados sin inconveniente. El estado traslado se identificará con "T".

**Validaciones Que Realizar:**

Las requeridas para que todos los folios a migrar con estado TRASLADADO se puedan realizar sin inconveniente. La validación que se realiza al momento de realizar una migración corresponde a la cantidad de folios que reporta la SNR en los archivos enviados para la migración.

#### **033\_453\_Ordenamiento y Visualizacion\_Anotaciones**

**Justificación:**

Re especificar el Requerimiento 068 para dar alcance a otras funcionalidades interrelacionadas con el ordenamiento de anotaciones al corregir número de anotación.

**Descripción:**

Para la pantalla administrativa VER DETALLES DE FOLIO, ver detalles del turno (al seleccionar el folio) y las demás consultas de folio existentes en los diferentes procesos registrales del aplicativo SIR es decir

- certificados (fase-Hoja de Ruta)
- corrección (fase revisión y análisis, fase actuaciones administrativas, fase ejecutar correcciones administrativas, fase Mesa control correcciones)
- Registro (fase calificación, fase Digitación Calificaciones, fase-Hoja de Ruta)

Se solicita que la información se visualice por el campo NTCN\_ORDEN Y ordenada por el campo ID\_ANOTACIÓN.

**Re-especificación del presente Requerimiento:**

Se solicita se unifiquen las pantallas de consulta de folio de los procesos registrales del aplicativo SIR

es decir certificados, consultas, corrección, Registro y Pantallas Administrativas ver detalles de folio, ver detalles del turno (al seleccionar el folio) para que se despliegue la información con el siguiente criterio:

Las anotaciones se deben ordenar internamente por el campo ID\_ANOTACION y visualizar el número de la anotación por el campo NTCN\_ORDEN.

La información de anotaciones, relacionada a salvedades, segregaciones, cancelaciones y gravámenes, el sistema debe asociarla y visualizarla por el campo NTCN\_ORDEN y no por el ID\_ANOTACION.

Para el caso de salvedades

- a. **Salvedades de la anotación.** El Sistema debe ordenar ascendente por la anotación y por el campo ID\_SALVEDAD. Al visualizar las salvedades para los procesos registrales del aplicativo SIR es decir:
  - a. Certificados (fase-Hoja de Ruta)
  - b. Corrección (fase revisión y análisis, fase actuaciones administrativas, fase ejecutar correcciones administrativas, fase Mesa control correcciones)
  - c. Registro
  - d. Pantallas Administrativas ver detalles de folio y ver detalles del turno (al seleccionar el folio)

Donde se consulten, se debe cambiar el título de la Columna Anotación Id o ID Anotación por el texto “Anotación” y que las columnas a desplegar sean las mismas que se despliegan en la pantalla de “Ver detalles folio”.

Las columnas que conforman las salvedades de la anotación son; Anotación, Radicación, turno corrección, usuario, fecha y salvedad.

- b. **Salvedades Del Folio.** El Sistema debe ordenar ascendente por el campo ID\_SALVEDAD. Esto aplica al visualizar las salvedades para los procesos registrales del aplicativo SIR es decir certificados, consultas, corrección, Registro y Pantallas Administrativas ver detalles de folio y ver detalles del turno (al seleccionar el folio) donde se consulten las salvedades del folio.

Las columnas que conforman las salvedades del folio son; Radicación, turno corrección, usuario, fecha y salvedad.

Para las segregaciones se debe ordenar por la anotación y por número de matrícula segregada en orden ascendente. (definir pantallas administrativas y procesos)

En el imprimible de Folio Simple las anotaciones se deben ordenar por el campo NTCN\_ORDEN y visualizar el número de la anotación por NTCN\_ORDEN, es decir, mantener el mismo orden establecido en el imprimible del Certificado de Tradición:

- Pantalla administrativa – Impresión “Impresión Simple de Folio”
- Proceso Consultas – fase consulta simple, compleja o exenta
- Pantalla administrativa (Consulta por Índice de Propietario).
- Proceso Correcciones fase Mesa control correcciones.
- Proceso Registro fase Certificados Asociados.

Al agregar una salvedad a una anotación, esta se debe mantener asociada a la anotación que le dio origen, de ninguna manera, estas salvedades pueden heredarse o copiarse a otra anotación, que por

algún motivo requiera incluirse una nueva salvedad, excepto cuando se editan los ciudadanos que afectan varias anotaciones relacionados a este folio u otros folios que también lo tengan relacionados. Las correcciones realizadas se reflejaran correctamente cuando estas correcciones se encuentren en estado definitivo.

**CORREO SNR**

Ingenieros

Buenas tardes

De acuerdo a la reunión de mesa de trabajo del dia de hoy ,con los ingenieros de Desarrollo y Calidad de ETB, las Ingenierías Dora Pachon e Imelda Muñoz y Yamile Moreno, para aclaración al requerimiento ordenamiento de anotaciones se acordó lo siguiente:

- No tener en cuenta las observaciones del correo enviado el 04 de septiembre de 2012, relacionado con el ordenamiento general
  - Con respecto al documento de requerimiento se aclara este ITEM
- Las anotaciones se deben ordenar internamente por el campo ID\_ANOTACION y visualizar el numero de la anotación por el campo NTCN\_ORDEN .

Debe quedar, ordenar internamente las anotaciones por NTCN\_ORDEN y visualizar las anotaciones por NTCN\_ORDEN

Cordialmente,

Yamile Moreno Melo

Direccion de Desarrollo Registral

**OBSERVACIONES**

Teniendo en cuenta este correo aplica el item "Debe quedar, ordenar internamente las anotaciones por NTCN\_ORDEN y visualizar las anotaciones por NTCN\_ORDEN

**036\_453\_Correccion\_Actuaciones\_Administrativas**

**Justificación:**

Para turnos de corrección avanzados a la fase ejecutar Actuaciones Administrativas y que en este proceso se realicen modificaciones a folios asociados al turno de corrección y se avance el turno a entrega de correcciones se debe generar formulario de correcciones.

**Descripción:**

Se requiere realizar ajustes a las siguientes funcionalidades:

- 1) Se requiere que en la fase de ejecutar Actuaciones Administrativas los turnos avanzados a esta fase y que en este proceso se realicen modificaciones a folios asociados al turno de corrección, al terminar el proceso de esta fase y dar clic en el botón aceptar, el sistema debe generar el formulario definitivo de correcciones, en el cual se reflejen las modificaciones de las matrículas asociadas al turno de corrección y avanzarlo a la fase entrega de correcciones.

**037\_453\_Proceso\_correcciones\_permitir\_varias\_correcciones**

**Justificación:**

Al realizar correcciones de ordenamiento en un folio el aplicativo no está guardando las correcciones realizadas.

**Descripción:**

Al realizar la corrección del orden de anotaciones en un folio en estado temporal, el aplicativo debe permitir la corrección de otros campos de esa anotación siempre y cuando se cuenten con los permisos para realizar dicha corrección, sin alterar el nuevo orden asignado. Como por ejemplo cuando se crea

inicialmente una anotación y luego se le cambia el orden con el numero de una anotación existente y ligada al folio que se le está realizando la corrección, luego se realiza una corrección de ciudadano en una de las anotaciones que se le hizo anteriormente una corrección, el aplicativo no está guardando las correcciones iniciales. Los permisos a asignar a un turno para la corrección son:

- Anotaciones - Ciudadanos
- Anotaciones – Especificacion
- Anotaciones – Fecha
- Anotaciones - Numero Radicacion
- Anotaciones – ValorActo
- Anotaciones – Documento
- Anotacion: Estado
- Anotaciones – Numero
- Anotaciones - Anotaciones a cancelar

Para la revisión de este requerimiento tiene como prerequisito el requerimiento “033\_453\_Ordenamiento y Visualizacion\_Anotaciones”.

#### **057\_453\_Reporte\_Articulo\_70\_Acto\_Administrativo (Informativo)**

##### **JUSTIFICACION:**

Para dar cumplimiento al artículo 70 del Código de Procedimiento Administrativo –Ley 1437 de 2011

**Artículo 70. Notificación de los actos de inscripción o registro.** Los actos de inscripción realizados por las entidades encargadas de llevar los registros públicos se entenderán notificados el día en que se efectúe la correspondiente anotación. Si el acto de inscripción hubiere sido solicitado por entidad o persona distinta de quien aparezca como titular del derecho, la inscripción deberá comunicarse a dicho titular por cualquier medio idóneo, dentro de los cinco (5) días siguientes a la correspondiente anotación

\*\*\*

Es necesario efectuar los ajustes a los sistemas de información misional, que operan en la Entidad, con el fin de dar cumplimiento a lo estipulado en la Ley 1437 de 2011.

Según comunicado SNR2012IE10461 de fecha 14 de septiembre de 2012, suscrito por el doctor Enrique José Nates y mediante correo electrónico enviado por la Ing. Leyla Soraya Guzmán, se impartieron las siguientes directrices frente al tema:

Generar un reporte por Orip en formato pdf , el cual debe contener como mínimo la siguiente información:

- Matricula inmobiliaria
- Descripción del acto (código y Descripción )
- Fecha de Inscripción del documento:
- Nombre completo de intervenientes (DE-A)–cuando exista más de un interveniente en una anotación en cualquiera de los roles (DE o A) se deben listar seguidos y horizontalmente, separados por el signo punto y coma (;)

Se tendrá en cuenta la fecha en la cual fue desanotado el turno, validando el rango de fechas de generación del reporte.

En el reporte se debe ver reflejado todos los actos que a continuación se relacionan y que además hayan sufrido alguna modificación durante la semana:

##### **CODIGO DESCRIPCION**

0137	Destinación definitiva
0142	Extinción del Derecho de dominio privado
0212	Valorización
0214	Liquidación del efecto de Plusvalía

0305	Afectación por causade una obra pública
0345	Afectación por causa de categorías ambientales
0356	Declaratoria de Zona Franca
0357	Declaración de reserva, alinderación y creación de área del sistema de parques nacionales y naturales
0359	Declaratria de bien inmueble de interés cultural numeral 1.2 Art 7 de la Ley 1185 de 2008
0360	Declaratoria de bien existencia del plan especial de manejo y protección numeral 1.2 Art 7 de la ley 1185 de 2008
0400	Medida cautelar
0401	Comiso especial
0402	Declaración no podra rematarse, adjudicarse ni enajenarsea ningun titulo bienes del garante
0403	Declaratoria de interes social
0404	Declaratoria de utilidad publica
0405	Decomiso de bienes
0406	Demanda en acción de simulación
0407	Demanda en acción Revocatoria
0408	Demanda en acuerdo de reestructuración
0409	Demanda en proceso de declaratoria de unión marital de hecho.
0410	Demanda en proceso de deslinde y amojonamiento
0411	Demandada en proceso de divorcio
0412	Demandada en proceso de pertenencia
0413	Demandada en proceso de separación de cuerpos
0414	Demandada en proceso de servidumbres
0415	Demandada en proceso divorcio
0416	Demandada en proceso separación de bienes
0417	Demandada ordinaria de enriquecimiento sin justa causa
0418	Demandada ordinaria por lesión enorme
0419	Demandada por expropiación
0420	Demandada por Ineficiencia del acuerdo de reestructuración
0421	Derecho de preferencia
0422	Emabargo concordatorio
0423	Embargo de alimentos
0424	Embargo de bienes y haberes de propiedad el intervenido
0425	Embargo de la sucesión
0426	Embargo derechos y acciones por garantía hipotecaria
0427	Embargo ejecutivo con acción personal
0428	Embargo ejecutivo con acción mixta
0429	Embargo ejecutivo con acción real
0430	Embargo ejecutivo derechos de cuota
0431	Embargo en acción de simulación
0432	Embargo en acción revocatoria
0433	Embargo en acuerdo de reestructuración
0434	Embargo en liquidación obligatoria
0435	Embargo en proceso de divorcio
0436	Embargo en proceso de fiscalia
0437	Embargo en proceso de separación de bienes
0438	Embargo especial Art 66 codigo de procedimiento penal (Ley 600/2000)

0439	Embargo laboral
0440	Embargo penal
0441	Embargo por impuestos municipales
0442	Embargo por impuestos nacionales
0443	Embargo por ineeficacia del acuerdo de reestructuración
0444	Embargo por jurisdicción coactiva
0445	Embargo por valorización
0446	Iniciación del proceso de enajenación forzosa
0447	Iniciación diligencias administrativas de clarificación de la propiedad
0448	Iniciación diligencias administrativas deslinde de tierras de propiedad de la nación
0449	Iniciación diligencias administrativas por indebida ocupación de baldíos
0450	iniciación procedimiento de expropiación por vía administrativa
0451	Iniciación procedimiento de extinción del derecho de dominio
0452	Iniciación proceso administrativo de adquisición
0453	Inscripción del proceso arbitral
0454	Oferta de compra en bien rural
0455	Oferta de compra en bien urbano
0456	Prevención no procede la realización de nuevos embargos sobre bienes de la intervenida
0457	Prevención registradores abstenerse cancelar gravámenes salvo autorización agente especial
0458	Prohibición administrativa
0459	Prohibición cancelación gravámenes constituidos a favor de intervenida sin autorización liiquidador
0460	Prohibición enajenar sin autorización
0461	Prohibición gravar, ceder, limitar o arrendar sin autorización
0462	Prohibición inscripción actos afecten dominio de bienes intervenida salvo lo realizado por agente especial
0463	prohibición judicial
0464	Prohibición registro actos afecten dominio bienes propiedad de intervenida salvo autorización liquidador
0465	Suspensión Exigibilidad de gravámenes y garantías reales y fiduciarias
0466	Toma de posesión inmediata de bienes, haberes y negocios de entidad vigilada
0467	Demandra de acción de petición de herencia
0468	Demandra en proceso ordinario
0469	Demandra en proceso reivindicatorio
0470	Abstenerse inscribir enajenaciones por declaratoria inminencia de riesgo o desplazamiento forzado
0471	Embargo en proceso de jurisdicción voluntaria
0472	Embargo en proceso ordinario
0473	Demandra en proceso liquidatorio
0474	Prohibición de enajenar derechos inscritos en predio declarado abandonado por el titular
0475	Embargo proceso abreviado
0476	Embargo proceso de reorganización Ley 1116 de 2006
0477	Demandra de proceso especial de saneamiento de la titulación de la propiedad inmueble art 8 de la ley 1182 de 2008
0478	Embargo proceso Art 95 de la Ley 1306 de 2009
0479	Suspensión provisional a la libre disposición de dominio en proceso de justicia y paz - ley 975 de 2005
0480	Inscripción de denuncia de desplazamiento y usurpación de bienes
0481	Inicio de proceso de liquidación judicial Art 47 Ley 1116 de 2006

0482	Protección jurídica del predio Art 13 No. 2 Decreto 4829 de 2001
0506	Destinación provisional
0507	Bienes recibidos por el Fondo para reparación de víctimas
0603	Afectación a vivienda familiar sobre mejoras en predio ajeno par. Art 5 Ley 258/96
0604	Compraventa de cosa ajena
0609	Declaración mejoras en predio ajeno par. Art 5 Ley 258/96
0619	Declaratoria de posesión regular Art 1. Ley 1183 de 2008

Es necesario aclarar que si el acto de inscripción fue solicitado por entidad o persona distinta de quien aparezca como titular del derecho, la inscripción deberá reflejarse en el reporte.

Dicho reporte será generado todos los viernes en horas nocturnas de forma automática y será publicado el mismo día de manera automática en la página WEB de la Entidad. Cada semana el reporte generado remplazará el de la semana inmediatamente anterior y será almacenado en un repositorio definido por la Oficina de Informática por un término mínimo de dos (2) años.

#### OBSERVACIONES

En el encabezado del reporte se debe visualizar NOMBRE COMPLETO DE LA ORIP

Un texto que indique “En virtud de la Ley 1437 de Enero 18 de 2011 (Artículo 70 del Código de Procedimiento Administrativo)”

Fecha de generación del reporte y fecha del período reportado

Número que identifique dicho reporte.

Se debe tener un total de actos por Oficina de Registro discriminado por códigos.

El Aplicativo debe permitir la consulta del reporte por fecha.

#### **058\_453\_Usuarios\_de\_Consulta\_Rol\_Impresion\_Reportes\_SIR**

#### **JUSTIFICACION:**

El Superintendente Delegado para el Registro, doctor Enrique José Nates Guerra mediante comunicado SNR 2012IE015041 de octubre 26 de 2012, solicita crear Usuarios de Consulta para funcionarios de esta dependencia que puedan generar todos los reportes del sistema SIR.

Igualmente Mediante oficio SNR2012IE14580 del 19 de octubre de 2012 el Director de Registro, solicita asignar Opción del Aplicativo SIR que permita generar reportes a la funcionaria Leipzig Duran Murillo

#### **DESCRIPCIÓN DEL REQUERIMIENTO:**

Se solicita creación de un nuevo Rol en el Aplicativo SIR, el cual debe permitir la generación de todos los reportes de los Círculos registrales de las Orips que operan en SIR.

Este Rol solamente debe ser asignado por el Administrador Nacional, por solicitud escrita de la SNR.

Una vez puesto en producción este requerimiento, se deben crear los siguientes usuarios a quienes se les debe asignar el mencionado Rol.

MARIA CLEMENCIA RANGEL FRANCO C.C. 28.423.586

ERNESTO ROA ROSAS C.C. 79.270.042

De otra parte se solicita asignar el Rol mencionado al usuario del Sistema SIR, Leipzig Duran.

#### **064\_453\_Duplicidad\_Nombre\_Circulo**

#### **JUSTIFICACION:**

En el proceso de migración de la Orip Bolívar –Antioquia, se detectó la duplicidad del nombre del círculo registral, por existir otra Orip con el mismo nombre pero de diferente Departamento, el cual

corresponde a Bolívar – Cauca, lo cual está generando confusión para seleccionar e identificar el Círculo correcto en el proceso de radicación de Certificados Nacionales y en la parametrización para generar reportes.

**Descripción del Requerimiento:**

En los parámetros para generar los reportes en el Aplicativo SIR, únicamente se debe visualizar y traer predeterminado el círculo desde donde se solicita el reporte, es decir solo desplegará el nombre del círculo Registral.

Igualmente en el proceso de radicación de certificados nacionales se debe crear un check box donde despliegue los departamentos y de acuerdo al departamento seleccionado, el sistema debe desplegar los círculos que pertenecen a ese departamento.

De acuerdo a el Acta de Reunión efectuada el pasado 22 de enero de 2013, se citaron los siguientes puntos como respuesta de la SNR ante este requerimiento:

RESPUESTA SNR:

- Queda a la espera del correo para depurar los usuarios que tienen asignados los roles.
- Se aprueba la modificación al listado de círculos donde se concatene el nombre del círculo con el nombre del departamento.
- Este ajuste se debe tener en cuenta para el nombre del circulo en todos los reportes, pantalla de certificados nacionales y todos los demás que impliquen la identificación del círculo.

**DESPLIEGUE – 2.7.9.1**

**01/03/2013**

**062\_453\_Modificación\_Identificación\_Circulo\_Tarifa**

Teniendo en cuenta que al realizar la migración de Bolívar – Antioquia, ETB reportó que al tratar de configurar la tarifa MINDER, se presentó un error de llave única por cuanto en la tabla SIR\_OP\_TARIFA, los campos ID\_TIPO, ID\_VERSION y ID\_CODIGO, donde el ID\_CODIGO corresponde al nombre del círculo, por lo cual se solicita realizar el siguiente cambio, el cual fue planteado por ustedes de acuerdo con la solicitud realizada por SNR y enviado por correo electrónico el día martes 08 de enero del año en curso, el cual se anexa en 4 folios así :

Buenos días, se verifica cambio a realizar en la aplicación, se debe actualizar una clase:

gov.sir.hermod.dao.impl.jdogenie.JDOLiquidacionesDAO.java

metodo  
getLiquidacionActo

Línea

TarifaEnhanced tarifaMinder = tarifasDAO.getTarifa(CTipoTarifa.REGISTRO\_MINDER,  
acto.getLiquidacion().getSolicitud().getCirculo().getNombre(), pm);

Por

TarifaEnhanced tarifaMinder = tarifasDAO.getTarifa(CTipoTarifa.REGISTRO\_MINDER,  
acto.getLiquidacion().getSolicitud().getCirculo().getNombre()+'\_'+acto.getLiquidacion().getSolicitud()  
().getCirculo().getIdCirculo(), pm);

Esta configuración se realiza teniendo en cuenta que el campo ID\_CODIGO de la tabla  
SIR\_OP\_TARIFA se le agregue “\_IDCIRCULO” EJM: “\_999”

**DESPLIEGUE – 2.7.9**

**16/02/2013**

**005\_453\_Error al Avanzar el Turno**

**REQUERIMIENTO GLPI No. 28137**

Turno:2011-200-6-9193

**Motivo Inconsistencia:**

Se comunica la usuaria Josefina Sánchez e informa que al tratar de avanzar el turno 2011-200-6-9193 para corrección de encabezado el sistema informa que este turno depende del turno de correcciones 2011-200-3-886 y no avanza.

Al tratar de avanzar el turno 2011-200-6-9193 para corrección de encabezado el sistema informa que este turno depende del turno de correcciones 2011-200-3-886 y no avanza, se requiere verificar y corregir el inconveniente ya que el turno de correcciones se encuentra anulado para poder seguir con el trámite del turno de registro.

Cuando un turno de registro crea un turno de corrección interna desde la fase calificación y este turno de corrección es anulado, el sistema al tomar el turno de registro notifica que hay una corrección pendiente y es el turno de correcciones que se anuló.

Grupo: Neiva

**014\_453\_Bloqueo de Folios\_MA**

GLPI 30712

**Motivo Inconsistencia:**

La usuaria Josefina Sánchez informa que al ingresar por la opción Desbloqueo Folio Restitución y colocar el número de folio y dar clic en aceptar la pantalla se queda en blanco tal como se muestra en la imagen adjunta.

Al ingresar por la opción Desbloqueo Folio Restitución y colocar el número de folio y dar clic en aceptar la pantalla se queda en blanco tal como se muestra en la imagen adjunta, se realizaron pruebas en producción y ocurre lo mismo, pero al consultar el folio si se genera el desbloqueo pero el sistema no refresca la pantalla y la deja en blanco tal como se describió anteriormente, se requiere verificar y corregir el inconveniente para poder desbloquear los folios correctamente.

**020\_453\_Redondeo en Reportes**

**Justificación:**

A partir del 16 de enero de 2012, fecha en la cual se Implementó en el Aplicativo SIR el Redondeo y actualización en las tarifas para la vigencia 2012, se presenta inconsistencia en la información de todos los Reportes, mostrando redondeo de tarifas para todas las fechas anteriores a su implementación, lo correcto es que se debe reflejar la información almacenada en la base de datos.

**Descripción:**

- Todos los reportes deben reflejar la información almacenada en la base de datos.
- Todos los reportes que se generen de fechas anteriores al 16 de enero de 2012, deben reflejar la información almacenada en la Base de Datos.
- En los reportes no se debe realizar redondeo para los valores de las cuantías de los actos que se reflejan actualmente en los reportes, entre otros como se observa el reporte 99.

**028\_453\_Ver\_detalles\_del\_turno\_correcciones**

**Justificación:**

Que al consultar un turno de corrección por pantallas administrativas, se pueda visualizar el motivo de la solicitud de corrección.

**Descripción:**

Se requiere realizar ajustes a las siguientes funcionalidades:

- 2) En la consulta de turnos de correcciones por pantallas administrativas “Ver detalles del turno”, se requiere que el aplicativo muestre la información del motivo de la solicitud, es decir, la incluida en el campo “Descripción” en el proceso de radicación de la solicitud de corrección. Este campo se visualizara en la sección Datos básicos de turno, después del ítem “fase relación actual”, este campo se desplegará solo para los turnos de correcciones.

Datos básicos de Turno	
Número del Turno	2012-999-3-157
Fecha de Radicación	19/06/2012
Tipo de Certificado	
Tipo de Tarifa	
Número de Recibo	
Valor Pagado	0
Valor por Pagar	0
Valor Devuelto	0
Valor por Devolver	0
Estado Actual	REVISION Y ANALISIS
Estación Asignada	X-USUARIO.PRUEBAS
Resolución	cuadro
Relación Actual	NO ESTA RELACIONADO
Fase Relacion Actual	NO ESTA RELACIONADO
Motivo de la Solicitud - Descripción	

- 3) Se requiere que en la consulta de turnos de correcciones por pantallas administrativas “Ver detalles del turno”, el historial de las actuaciones administrativas se refleje en orden de ingreso de la información, así mismo se debe incluir una columna para que despliegue la descripción de cada uno de los estados.

### **031\_453\_Imprimible\_Tipo\_Certificado**

#### **Justificación:**

Identificar el tipo de Certificado en el imprimible del Recibo de Solicitud de Certificados INMEDIATOS, NACIONALES, ANTIGUO SISTEMA, PERTENENCIA, AMPLIACION DE TRADICION.

#### **Descripción:**

Se requiere que en el imprimible de las solicitudes de Certificado de Tipo INMEDIATO, NACIONALES, ANTIGUO SISTEMA, PERTENENCIA Y AMPLIACION DE TRADICION, muestre un texto a la izquierda, siguiente a la línea del número de radicación, así: “TIPO DE CERTIFICADO: xxxxxxxxxxxxxxxx” se mostrara el nombre del tipo a que corresponda la solicitud.

### **039\_453\_Validaciones\_Folios\_Estado\_Bloqueado**

#### **Justificación:**

Validar que los folios en estado bloqueado (B) se restrinjan en el aplicativo SIR.

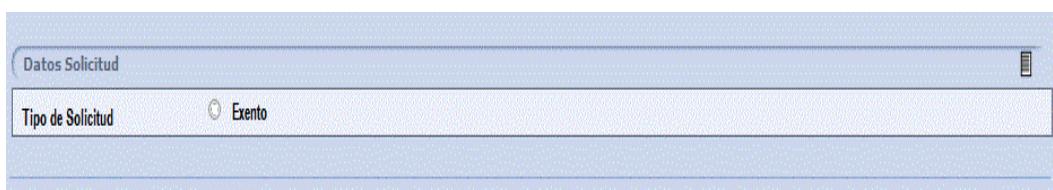
#### **Descripción:**

Validar que los folios en estado bloqueado (B) se restrinjan en el aplicativo SIR para los procesos de:  
 Certificados  
 Consultas  
 Correcciones  
 Registro  
 Certificados Masivos

Y pantallas administrativas:

Impresión Simple de Folios  
 Ver detalles de Folio  
 Consulta por Índice de Propietario

El estado de estos folios cambiara cuando se requiera realizar un traslado del folio, esta solicitud la realizar la SNR en el momento que lo requiera (Se ejecutara script en el cual se cambiara el estado

"B" a "S")	
<b>041_453_Requerimientos_Salvedad</b>	
<b>JUSTIFICACION:</b>	
Teniendo en cuenta que al realizar la grabación de la información de los folios de matricula inmobiliaria de las Orips que se encuentran en proceso de Sistematización, se detecto una salvedad, cuya longitud de la descripción supera los 255 caracteres que actualmente tiene definido este campo. Por lo anterior se debe ampliar el campo de Descripción del objeto SIR_NE_SAVEDAD_AN, el cual debe permitir la inserción de información, con tamaño mayor a 255 caracteres.	
<b>DESCRIPCIÓN:</b>	
Se solicita alterar las tablas:	
<ol style="list-style-type: none"> <li>1. SIR_NE_SALVEDAD_AN, de tal forma que el campo SLAN_DESCRIPCION quede con un tamaño de 1024 bytes.</li> <li>2. SIR_NE_SALVEDAD_AN_TMP, de tal forma que el campo SATM_DESCRIPCION quede con un tamaño de 1024 bytes.</li> </ol>	
<b>042_453_Texto_Recibo_Certificados_Nacionales</b>	
<b>JUSTIFICACION:</b> Mediante el caso 35468 radicado en la Herramienta de GLPI, la Orip de Bucaramanga reporta a la Mesa de ayuda que el imprimible del recibo de pago de la solicitud de certificado Nacional, muestra un texto que dice "NO SE EXPIDE DE INMEDIATO EL CERTIFICADO", lo cual no es correcto.	
Por lo expuesto se debe corregir este texto por que la expedición de estos certificados es inmediato.	
<b>DESCRIPCIÓN:</b>	
Es necesario validar la información entre el recibo de solicitud de certificado de tradición y la base de datos, ya que al Imprimir un Certificado Nacional refleja erradamente el texto "NO SE EXPIDE DE INMEDIATO EL CERTIFICADO" este mensaje debe corregirse por el texto "CERTIFICADO (S) SE EXPIDE (N) DE INMEDIATO"	
<b>044_453_Liquidación_Fotocopias_Exentas</b>	
<b>JUSTIFICACION:</b>	
Mediante oficio 373 del 28 de junio de 2012 la Registradora seccional de Instrumentos Pùblicos de San Marcos Sucre, informa que en el Proceso de Liquidación de Fotocopias, el Aplicativo SIR, no tiene la opción de tarifa exenta, para atender las solicitudes de fotocopias de documentos que reposan en los archivos de la Orip, las cuales son requeridas por entidades Estatales.	
En el Rol Usuario Operativo Fotocopias, fase de Liquidación de Fotocopias en la pantalla de liquidación se debe agregar una caja de chequeo que permita seleccionar "tipo de solicitud Exenta" esto para cuando se trate de liquidar solicitudes de fotocopias que no se aplica tarifa porque son entidades estatales.	
<b>DESCRIPCION:</b>	
Al seleccionar Tipo de Solicitud Exenta el aplicativo debe liquidar valor \$0, para todos los ítems de la lista de selección "Medio" ( Normal, Mecánico, Óptico, Scaner) y para cualquier tipo de documento de los requeridos en la solicitud.	
	
<b>045_453_Correccion_o_Formulario_Inscripcion_Testamentos</b>	
<b>JUSTIFICACION:</b>	
La Orip de Sansón informa telefónicamente que no encuentra dentro del proceso de Correcciones una opción que permita corregir la información ingresada en la calificación del registro de Testamentos,	

reflejada en el formulario de este proceso.

Por lo anterior se requiere ajustar el Aplicativo, el cual debe permitir corregir la información ingresada en la calificación de documentos con actos de Testamentos y que se encuentre en estado definitivo.

#### **DESCRIPCION:**

#### **FASE SOLICITUD**

- En esta fase se debe crear una sección “Turno testamentos”, donde se agregue un turno de registro con las siguientes características.
  - Acto testamento.
  - Resolución Inscrita.
  - Estado Desanotado.
  - Subtipo de solicitud TESTAMENTO.

#### **Validaciones:**

En esta fase solo se agregara un turno de registro de testamentos y una vez se encuentre asociado, no se debe permitir agregar folios.

El turno a asociar debe ser del mismo circulo.

#### **FASE REVISION Y ANALISIS**

- En esta fase se debe crear una sección “Turno testamentos” donde se visualice el turno de registro asociado a la corrección y al frente muestre un botón que al dar clic en el despliegue el formulario de Testamentos.
- En esta fase se debe crear el permiso de “TESTAMENTOS”, la cual debe ser default para cuando el turno de correcciones tenga asociado un turno de Registro de Testamentos.

#### **Validaciones:**

No debe obligar a que el turno de correcciones tenga folios asociados.

#### **FASE EJECUTAR CORRECCION SIMPLE**

- En esta fase el usuario seleccionara el turno de registro con acto de TESTAMENTO que asoció en el rol cajero correcciones, al seleccionarlo se desplegará el formulario (Ver Rol Testamentos – Formulario Registro de testamentos) en el cual se habilitaran los campos de la información del tipo de documento, número, fecha, oficina de origen, Testador, Tomo, Número de anotación, fecha, Revoca escritura, observaciones. El sistema debe permitir en esta pantalla corregir o agregar información en los campos mencionados.

#### **Validaciones:**

La salvedad debe ser obligatoria.

Se implementará auditorías para guardar el histórico de los cambios realizados al testamento.

## FASE REVISAR Y APROBAR

- En esta fase se debe crear una sección “Turno testamentos” donde se visualice el turno de registro asociado a la corrección y al frente muestre un botón que al dar clic en el despliegue el formulario de Testamentos.
- En el formulario de aprobar corrección en el botón “CONSULTAR”, se debe reflejar los cambios realizados al testamento (actualizar consulta para que muestre las correcciones del testamento).
- El sistema debe permitir devolver ejecutar corrección o deshacer cambios y devolver automáticamente a la fase revisión y análisis.
- Al aprobar los cambios el sistema debe generar el imprimible del formulario de correcciones , en el cual se debe reflejar el número del turno de registro del cual se corrigió el formulario de la constancia de inscripción de Testamentos, el tipo de documento ,numero fecha, oficina de Origen, debe reflejarse la información de las salvedades , con las condiciones que aplican actualmente para los formularios de correcciones, a cambio del número de anotaciones debe indicarse el nombre del campo corregido.

### Validaciones:

El formulario de corrección de testamentos será un nuevo imprimible y no debe afectar el formulario de corrección actual.

Al avanzar el turno de correcciones a Mesa de Control Correcciones, quedara en definitivo la información del formulario de Testamentos.

## MESA DE CONTROL CORRECCIONES

- En esta fase se debe crear una sección “Turno testamentos” donde se seleccione el turno de registro de testamentos asociado y el botón consultar seleccionado.
- Permitir avanzar el turno de registro de testamentos asociado a la fase de Desanotado –entrega de correcciones y Entregado como los demás turnos de correcciones.

**Validaciones:**

En esta fase no debe permitir certificados para este tipo de turnos por no llevar asociada matrícula inmobiliaria (N/A “Consultar Folios válidos para imprimir”).

**VER DETALLES DE TURNO**

- En esta pantalla administrativa se debe crear:
  - Una sección “Turno testamentos” donde se visualice el turno de registro de testamentos asociado.
  - En la parte inferior de la pantalla se pueda consultar el formulario de testamento, de la misma forma se pueda visualizar en la consulta del turno de registro, pero que el sistema no permita imprimirlo, únicamente permitirá consultarla.

**046\_453\_Aguachica\_No\_Recauda\_Impuesto\_Registro****JUSTIFICACIÓN:**

Mediante el caso 35110 radicado en la Herramienta GLPI, la Orip de Aguachica reporta a la Mesa de Ayuda, que el aplicativo liquida Multa de Impuesto en el proceso de radicación Registro, lo cual no es correcto, porque la Orip no recauda impuesto.

Por lo expuesto se deben realizar los ajustes pertinentes en el Aplicativo SIR para solucionar esta incidencia

**DESCRIPCIÓN:**

Se requiere que para las ORIPS en las cuales se ha inhabilitado el cobro o liquidación del Impuesto de Registro, en el Proceso de Radicación Registro, el Aplicativo SIR no debe liquidar multa de impuesto para registro, para documentos de nuevas entradas o documentos con fechas de otorgamiento que superan los términos establecidos para el pago correspondiente.

De la misma forma debe aplicarse para las ORIPS que soliciten inhabilitar esta funcionalidad.

**DESPLIEGUE – 2.7.8****13/11/2012****055\_453\_Notas\_Devolutivas\_Causales\_Texto\_e\_Histórico**

En virtud de la ley 1579 del primero de octubre de 2012 y Código de Procedimiento Administrativo – Ley 1437 de 2011, es necesario modificar el texto de las causales de Devolución y formato de Nota Devolutiva en el Aplicativo SIR.

Lo anterior de acuerdo a la revisión funcional realizada por la Superintendencia Delegada de Registro y Dirección de Registro, según soportes de correos y oficios.

Para dar cumplimiento a la normatividad vigente, se solicita realizar los siguientes ajustes en el Aplicativo SIR:

- 1) Las Causales de Notas Devolutivas se deben ajustar de acuerdo al archivo adjunto, el cual contiene las modificaciones respectivas y estas son las causales que deben quedar vigentes en el Aplicativo SIR.
- 2) En el Formato de Notas Devolutivas también se deben realizar ajustes en los textos conforme el archivo adjunto para el mencionado formato

Es necesario, crear un histórico de las notas devolutivas existentes antes de la fecha de actualización en el aplicativo, de tal manera que se conserve el historial y se puedan consultar e imprimir notas devolutivas con vigencias anteriores, relacionadas con los respectivos turnos, haciendo referencia a la normatividad registral con la que originalmente fue expedida.

De: Yamile Moreno Melo [mailto:yamile.moreno@supernotariado.gov.co]

Enviado el: martes, 30 de octubre de 2012 11:06 a.m.

Para: Fernando Padilla Velez; dany.agon@globalbis.co

CC: Ivonne Amparo Bernal Lopez; Angelica Maria Herreño; Carlos Augusto Prieto Leon; Dora Esperanza

Pachon Rodriguez

Asunto: RV: modificación nota devolutiva

Importancia: Alta

Ingenieros buenos días

Por favor tener en cuenta otras observaciones para dar alcance al requerimiento notas devolutivas “ estas son para el Formato de Notas Devolutivas”

De acuerdo a otra revisión de la Superintendencia Delegada, el penúltimo párrafo de la nota devolutiva quedará así:

“EXCEPTÚESE DE LO ANTERIOR, LOS CASOS RELACIONADOS CON EL NEGOCIO JURÍDICO DE HIPOTECA Y EL ACTO DE CONSTITUCIÓN DE PATRIMONIO DE FAMILIA DE QUE TRATA EL ARTÍCULO 28 DE LA LEY 1579 DE 2012, LOS CUALES SE DEBEN REGISTRAR DENTRO DE LOS NOVENTA (90) DIAS HÁBILES.....”

El formato de notas Devolutivas debe mantener con los logos y fondo que actualmente tiene .

Cordialmente,

Yamile Moreno Melo

Direccion de Desarrollo Registral

Adjunto documentos anexos al requerimiento:



## OFICINA REGISTRO INSTRUMENTOS PUBLICOS BOGOTA ZONA CENTRO

### NOTA DEVOLUTIVA

Página 1

Impresa el 23 de Octubre de 2012 a las 01:52:30 p.m

El documento ESCRITURA Nro. 4513 del 09-11-2011 de NOTARIA CINCUENTA BOGOTA D.C. fue presentado para su inscripción como solicitud de registro de documentos con Radicación: 2011-72022 vinculado a la Matrícula Inmobiliaria: 50C-3677

Conforme con el principio de legalidad previsto en el literal d) del artículo 3 y en el artículo 22 de la Ley 1579 de 2012 (Estatuto de Registro de Instrumentos Públicos) se inadmite y por lo tanto se devuelve sin registrar el presente documento por las siguientes razones y fundamentos de derecho:

- LOS DERECHOS QUE SE PRETENDEN ENAJENAR, SOBRE EL PREDIO, CUENTAN CON UNA MEDIDA DE PROTECCIÓN SOLICITADA POR EL DESPLAZADO TITULAR DE DICHOS DERECHOS Y ESTA DEBE SER CANCELADA PREVIAMENTE (LEY 387 DE 2007 Y ARTÍCULOS 61 Y 62 DE LA LEY 1579 DE 2012)
- EL DOCUMENTO QUE SE PRETENDE ACLARAR NO HA SIDO REGISTRADO (ARTÍCULO 29 LEY 1579 DE 2012).
- 

UNA VEZ SUBSANADA(S) LA(S) CAUSAL(ES) QUE MOTIVO LA NEGATIVA DE INSCRIPCION, FAVOR RADICAR NUEVAMENTE EN ESTA OFICINA, EL DOCUMENTO PARA SU CORRESPONDIENTE TRAMITE, ADJUNTANDO LA PRESENTE NOTA DEVOLUTIVA.

CUANDO LA CAUSAL O LAS CAUSALES QUE RECHAZA (N) LA INSCRIPCION DEL DOCUMENTO NO SEA (N) SUBSANABLE (S), SE CONFIGURE EL PAGO DE LO NO DEBIDO, SE PRODUZCAN PAGOS EN EXCESO O SE DESISTA DEL TRAMITE, EL TERMINO PARA SOLICITAR LA DEVOLUCION DE DINERO PARA LOS DERECHOS DE REGISTRO ES DE CUATRO (4) MESES CALENDARIO, CONTADOS A PARTIR DE LA FECHA DE EJECUTORIA DEL ACTO ADMINISTRATIVO QUE NIEGA EL REGISTRO O SE DESANOTE EL DOCUMENTO INSCRITO.

PARA SOLICITAR LA DEVOLUCION DE DINERO POR CONCEPTO DE IMPUESTO DE REGISTRO, DEBE DIRIGIRSE A LA GOBERNACION DEL DEPARTAMENTO DE CUNDINAMARCA, EN LOS TERMINOS DEFINIDOS POR EL ARTICULO 15 DEL DECRETO 650 DE 1996

LOS ACTOS O NEGOCIOS JURIDICOS A QÜE SE REFIERE EL ARTICULO 4 DE LA LEY 1579 DE 2012, DEBERAN PRESENTARSE PARA SU INSCRIPCION, DENTRO DE LOS DOS (2) MESES CALENDARIO, SIGUIENTES A LA FECHA DE SU OTORGAMIENTO PARA ACTOS NOTARIALES O LA FECHA DE EJECUTORIA PARA PROVIDENCIAS JUDICIALES O ADMINISTRATIVAS; VENCIDOS LOS CUALES, SE COBRARAN INTERESES MORATORIOS POR IMPUESTO DE REGISTRO, PREVISTOS EN LA LEY 223 DE 1995 Y SU DECRETO REGLAMENTARIO 695 DE 1996 ARTICULO 14.

EXCEPTEUSE DE LO ANTERIOR, LOS CASOS RELACIONADOS CON EL NEGOCIO JURIDICO DE HIPOTECA Y EL ACTO DE CONSTITUCION DE PATRIMONIO DE FAMILIA DE QUE TRATA EL ARTICULO 32 DEL DECRETO LEY 1250 DE 1970, LOS CUALES SE DEBEN REGISTRAR DENTRO DE LOS NOVENTA (90) DIAS HABILES SIGUIENTES A SU AUTORIZACION. VENCIDO EL TERMINO REGISTRAL ANTES SENALADO, DEBERAN CONSTITUIRSE DE CONFORMIDAD CON EL PRECITADO ARTICULO.

CONTRA EL PRESENTE ACTO ADMINISTRATIVO, PROCEDA EL RECURSO DE REPOSICION ANTE EL REGISTRADOR DE INSTRUMENTOS PUBLICOS Y EN SUBSIDIO, EL DE APELACION ANTE EL DIRECTOR DE REGISTRO DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO DENTRO DE LOS DIEZ (10) DIAS HABILES SIGUIENTES A SU NOTIFICACION, EN VIRTUD DE LO PREVISTO POR EL NUMERAL DOS (2) DEL ARTICULO 24 DEL DECRETO 2163 DEL 17 DE JUNIO DE 2011, PREVIO EL CUMPLIMENTO DE LOS REQUISITOS ESTABLECIDOS EN LOS ARTICULOS 76 Y 77, LEY 1437 DE 2011 (CODIGO DE PROCEDIMIENTO ADMINISTRATIVO Y DE LO CONTENCIOSO ADMINISTRATIVO).

FUNCIONARIO CALIFICADOR

REGISTRADOR DE INSTRUMENTOS PUBLICOS

FOLIO

=====



SUPERINTENDENCIA DE NOTARIADO OFICINA REGISTRO INSTRUMENTOS PUBLICOS BOGOTA ZONA CENTRO

NOTA DEVOLUTIVA

Página 2

Impresa el 23 de Octubre de 2012 a las 01:52:30 p.m.

El documento ESCRITURA Nro. 4513 del 09-11-2011 de NOTARIA CINCUENTA BOGOTA D.C. fue presentado para su inscripción como solicitud de registro de documentos con Radicación: 2011-72022 vinculado a la Matrícula Inmobiliaria: 50C-3677

NOTIFICACION PERSONAL

CONFORME LO DISPUESTO EN EL ARTICULO 67 DEL CODIGO DE PROCEDIMIENTO ADMINISTRATIVO Y DE LO CONTENCIOSO ADMINISTRATIVO EN LA FECHA \_\_\_\_\_ SE NOTIFICO PERSONALMENTE EL PRESENTE ACTO ADMINISTRATIVO A \_\_\_\_\_ QUIEN SE IDENTIFICO CON \_\_\_\_\_ No. \_\_\_\_\_

FUNCIONARIO NOTIFICADOR

El documento ESCRITURA Nro. 4513 del 09-11-2011 de NOTARIA CINCUENTA BOGOTA D.C. Radicación: 2011-72022

EL NOTIFICADO

SUPERINTENDENCIA  
DE NOTARIADO  
Y REGISTRO  
LA GUARDIA DE LA PUEBLA

NOTAS DEVOLUTIVAS EXISTENTES (MODIFICADAS)- APPLICATIVO SIR

ID_TIPO_NOTA	DESCRIPCION
100	* * * ASPECTOS GENERALES * * *
101	EL PREDIO NO CORRESPONDE A ESTE CIRCULO REGISTRAL (ARTICULO 5 LEY 1579 DE 2012).
102	EL DOCUMENTO SOMETIDO A REGISTRO CONTIENE UN ACTO CUYA NATURALEZA

	JURIDICA NO ES SUSCEPTIBLE DE INSCRIPCION (ARTICULO 4 LEY 1579 DE 2012).
103	LAS PROTOCOLIZACIONES NO SON OBJETO DE REGISTRO (ARTICULO 4 LEY 1579 DE 2012).
104	EL PRESENTE DOCUMENTO YA FUE REGISTRADO (ARTICULO 20, 21 y 23 LEY 1579 DE 2012).
105	LAS DECLARACIONES DE MEJORAS EN SUELO AJENO NO SON OBJETO DE REGISTRO (ARTICULO 4 LEY 1579 DE 2012), INSTRUCCIONES ADMINISTRATIVAS 15 Y 23 DE 1994 DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO).
106	EL DOCUMENTO PRESENTADO PARA REGISTRO ES UNA FOTOCOPIA O COPIA SIMPLE Y NO CUMPLE CON LOS REQUISITOS ESTABLECIDOS EN EL NUMERAL PRIMERO DEL ARTICULO 254 DEL CODIGO DE PROCEDIMIENTO CIVIL.
107	EL DOCUMENTO PRESENTADO PARA REGISTRO ESTA INCOMPLETO (ARTICULOS. 20 Y 23 DEL DECRETO LEY. 960/70, PARAGRAFO 1 DEL ARTICULO 14 Y PARAGRAFO 1 DEL ARTICULO 16 DE LA LEY 1579 DE 2012).
108	A LA PRESENTE PROVIDENCIA LE FALTA LA CONSTANCIA DE EJECUTORIA (ARTICULOS 333 Y 334 DEL CODIGO DE PROCEDIMIENTO CIVIL).
109	FALTA COPIA AUTENTICA PARA EL ARCHIVO DE ESTA OFICINA (PARAGRAFO 1 del ARTICULO 14 LEY 1579 DE 2012).
110	EL DOCUMENTO SOMETIDO A REGISTRO NO CITA TITULO ANTECEDENTE Y/O ADQUISITIVO DE DOMINIO (ARTICULO 29 LEY 1579 DE 2012).
111	EL TITULO CITADO COMO ANTECEDENTE REGISTRAL NO CORRESPONDE AL INSCRITO EN EL FOLIO DE MATRICULA INMOBILIARIA (ARTICULO 32 DEL DECRETO LEY. 960/70 Y ARTICULO 29 LEY 1579 DE 2012).
112	NO CITA NUMERO DE MATRICULA INMOBILIARIA U OTROS DATOS DEL SISTEMA REGISTRAL ANTIGUO QUE IDENTIFIQUE EL PREDIO EN NUESTROS ARCHIVOS (ARTICULO 8 Y PARAGRAFO 1 DEL ARTICULO 16 LEY 1579 DE 2012).
113	LA(S) MATRICULA(S) CITADA(S), COMO IDENTIFICACION DEL PREDIO(S) NO ES (SON) CORRECTAS (ARTICULO 8 Y PARAGRAFO 1 DEL ARTICULO 16 DE LEY 1579 DE 2012).
114	FALTA PAGO DEL IMPUESTO DE REGISTRO (LEY 223/1995 Y DECRETO 650/96).
115	FALTA PAGO DE INTERES MORATORIO POR EXTEMPORANEIDAD EN EL REGISTRO (DOS MESES SI EL DOCUMENTO ES OTORGADO O EXPEDIDO EN EL PAIS Y TRES MESES SI ES OTORGADO O EXPEDIDO EN EL EXTERIOR (LEY 223/1995 Y DECRETO 650/96).
116	NO SE ESTABLECE EL VALOR DE LA CONSTITUCION DE SERVIDUMBRE PARA EFECTOS DEL COBRO DE DERECHOS DE REGISTRO (PARAGRAFO 5 ARTICULO 1 DECRETO 2280 DE 2008).
117	NO SE HA INSERTADO PAZ Y SALVO DEL IMPUESTO PREDIAL DEL INMUEBLE. (ARTICULO 1 LEY 33 1896, ARTICULOS 43 Y 44 DEL DECRETO 960 DE 1970, ARTICULO 45 DEL DECRETO 2163 DE 1970, ARTICULO 26 DEL DECRETO 2148 DE 1983).
118	NO SE CITA EL AVALUO CATASTRAL PARA EFECTOS DEL COBRO DE LOS DERECHOS DE REGISTRO (ARTICULO 1 DECRETO 2280 DE 2008).
119	FALTA PAGO DE DERECHOS DE REGISTRO (ARTICULO 1 DECRETO 2280 DE 2008).
120	DEBE DETERMINARSE EL DERECHO DE CUOTA QUE SE ENAJENA (SENTENCIA DEL 29-11-1967 DE LA CORTE SUPREMA DE JUSTICIA). PARA EL COBRO DE LOS DERECHOS DE REGISTRO DEBE DETERMINARSE EL DERECHO DE CUOTA TRANSFERIDO (PARAGRAFO 2 DEL ARTICULO 1 DEL DECRETO 2280 DE 2008).
121	EN LA RESCILIACION, RESCISION O RESOLUCION DEL CONTRATO SE OMITIO INDICAR EL VALOR O PRECIO DEL CONTRATO POR EL CUAL FUE CONSTITUIDO. (PARAGRAFO DEL ARTICULO 8 DEL DECRETO 2280 DE 2008).
122	NO SE CITA LA TOTALIDAD DE LAS MATRICULAS EN LAS CUALES DEBE INSCRIBIRSE EL

	DOCUMENTO (ARTICULO 8 LEY 1579 DE 2012).
123	EN CASO DE DESISTIMIENTO DEL REGISTRO O QUE ESTE NO SEA PROCEDENTE, CUENTAN LAS PARTES CON CINCO (5) DIAS HABILES PARA SOLICITAR LA DEVOLUCION DE LOS DINEROS PAGADOS POR CONCEPTO DE IMPUESTO DE REGISTRO (DECRETO 650 DE 1996).
124	CON LOS DATOS SUMINISTRADOS EN EL FORMULARIO, MAS LA INFORMACION EXISTENTE EN LA OFICINA DE REGISTRO DE INSTRUMENTOS PUBLICOS Y LA APORTADA POR OTRAS ENTIDADES O EL NO ENVIO DE LA MISMA, NO FUE POSIBLE UBICAR EL FOLIO DE MATRICULA INMOBILIARIA QUE IDENTIFIQUE EL PREDIO QUE DECLARA EN ABANDONO (ARTICULO 8, PARAGRAFO 1 DEL ARTICULO 16 Y ARTICULO 50 LEY 1579 DE 2012). NO OBSTANTE, EL NO REGISTRO DE LA MEDIDA DE PROTECCION EN EL FOLIO DE MATRICULA INMOBILIARIA, SU SOLICITUD SERA INGRESADA AL REGISTRO UNICO DE PREDIOS ABANDONADOS, PARA QUE ESA INSCRIPCION SIRVA COMO MEDIO PROBATORIO ANTE LAS AUTORIDADES JUDICIALES Y ADMINISTRATIVAS.
125	EL PREDIO SOBRE EL QUE SE SOLICITA LA PROTECCION, ES UN BALDIO QUE PERTENECE A LA NACION, NO CUENTA CON UN FOLIO DE MATRICULA INMOBILIARIA Y LA PROPIEDAD DE LOS TERRITORIOS BALDIOS SE ADJUDICA POR TITULO TRASLATICIO DE DOMINIO A TRAVES DEL INCODER (ARTICULOS 63 CN; ARTICULO 8, PARAGRAFO 1 DEL ARTICULO 16 Y ARTICULO 50 LEY 1579 DE 2012; ARTICULO 65 LEY 160/94) NO OBSTANTE EL NO REGISTRO DE LA MEDIDA DE PROTECCION EN EL FOLIO DE MATRICULA INMOBILIARIA, SU SOLICITUD SERA INGRESADA AL REGISTRO UNICO DE PREDIOS ABANDONADOS , PARA QUE ESA INSCRIPCION SIRVA COMO MEDIO PROBATORIO ANTE LAS AUTORIDADES JUDICIALES Y ADMINISTRATIVAS.
126	EL PREDIO DEL CUAL ES PROPIETARIO SE ENCUENTRA PROTEGIDO POR LA PROHIBICION DE ENAJENAR, PROFERIDA POR EL COMITE DEPARTAMENTAL O MUNICIPAL DE ATENCION A LA POBLACION DESPLAZADA. ( NUMERAL 2 ARTICULO 1 DECRETO 2007 DE 2001). NO OBSTANTE EL NO REGISTRO DE LA MEDIDA DE PROTECCION EN EL FOLIO DE MATRICULA INMOBILIARIA, SU SOLICITUD SERA INGRESADA AL REGISTRO UNICO DE PREDIOS ABANDONADOS, PARA QUE ESA INSCRIPCION SIRVA COMO MEDIO PROBATORIO ANTE LAS AUTORIDADES JUDICIALES Y ADMINISTRATIVAS.
127	EL PREDIO NO ES SUSCEPTIBLE DE APROPIACION PRIVADA PUES SE TRATA DE UN BIEN FISCAL ___, PARQUE NACIONAL ___, TERRITORIO DE COMUNIDAD ETNICA ___; ZONA DE RIESGO ___ Y EN RELACION CON ESTOS NO ES POSIBLE A LOS PARTICULARES, ESTABLECER DERECHOS PATRIMONIALES, POR SER IMPREScriptIBLES INENAJENABLES E INEMBARGABLES (ARTICULO .63 CONSTITucion NACIONAL). NO OBSTANTE EL NO REGISTRO DE LA MEDIDA DE PROTECCION EN EL FOLIO DE MATRICULA INMOBILIARIA, SU SOLICITUD SERA INGRESADA AL REGISTRO UNICO DE PREDIOS ABANDONADOS, PARA QUE ESA INSCRIPCION SIRVA COMO MEDIO PROBATORIO ANTE LAS AUTORIDADES JUDICIALES Y ADMINISTRATIVAS.
128	EN EL FOLIO DE MATRICULA QUE IDENTIFICA EL PREDIO QUE USTED DECLARA HABER ABANDONADO POR CAUSA DE LA VIOLENCIA, PRESENTA INSCRITA CON ANTERIORIDAD A SU SOLICITUD UNA TRANSFERENCIA DE DOMINIO. ARTICULOS 669 Y 756 DEL CODIGO CIVIL. NO OBSTANTE EL NO REGISTRO DE LA PROHIBICION DE ENAJENAR EN EL FOLIO DE MATRICULA INMOBILIARIA, SU SOLICITUD SERA INGRESADA AL REGISTRO UNICO DE PREDIOS ABANDONADOS PARA QUE ESA INSCRIPCION SIRVA COMO MEDIO PROBATORIO ANTE LAS AUTORIDADES JUDICIALES Y ADMINISTRATIVAS.
129	EL PREDIO QUE PRETENDE ENAJENARSE SE ENCUENTRA COBIJADO CON DECLARATORIA DE DESPLAZAMIENTO FORZADO O DE INMINENCIA DE DESPLAZAMIENTO. DEBE ANEXAR LA AUTORIZACION DEL RESPECTIVO COMITE DE ATENCION INTEGRAL A LA POBLACION DESPLAZADA (DECRETO 2007 DE 2001).

130	LOS DERECHOS QUE SE PRETENDEN ENAJENAR, SOBRE EL PREDIO, CUENTAN CON UNA MEDIDA DE PROTECCION SOLICITADA POR EL DESPLAZADO TITULAR DE DICHOS DERECHOS Y ESTA DEBE SER CANCELADA PREVIAMENTE (LEY 387 DE 2007 Y ARTICULOS 61 Y 62 DE LA LEY 1579 DE 2012)
131	LA NOMENCLATURA Y DETERMINACION DEL INMUEBLE NO CORRESPONDE A LA REGISTRADA (ARTICULO 8 Y PARAGRAFO 1 DEL ARTICULO 16 DE LA LEY 1579 DE 2012).
132	FALTA PAZ Y SALVO DE ADMINISTRACION PARA LA INSCRIPCION DE LOS ACTOS TRASLATICIOS DE DOMINIO (ARTICULO 29 LEY 675/2001).
133	EL PODER OTORGADO EN EL EXTERIOR DEBE EXHIBIR EL APOSTILLE ANTE EL MINISTERIO DE RELACIONES EXTERIORES. (ARTICULO 259 DEL CODIGO .DE PROCEDIMIENTO CIVIL).
134	EN EL FOLIO DE MATRICULA INMOBILIARIA SE ENCUENTRA REGISTRADA LIQUIDACION DEL EFECTO PLUSVALIA VIGENTE. (ARTICULO 81 LEY 388/97, INSTRUCCION ADMINISTRATIVA 31 DEL 28-12-2004 DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO).
135	LAS PROVIDENCIAS DEBEN SER ACLARADAS POR LA MISMA AUTORIDAD QUE LAS PROFIRIO (ART. 310 CODIGO DE PROCEDIMIENTO CIVIL).
136	CONFORME A LO DISPUESTO EN EL ARTICULO 18 DEL DECRETO LEY 960/70 EN EL TEXTO DE LA ESCRITURA NO DEBE HABER ESPACIOS EN BLANCO.
137	<b>INACTIVA</b>
138	PARA PROCEDER AL REGISTRO PARCIAL DEL DOCUMENTO, ESTE DEBE SER SOLICITADO POR TODAS LAS PARTES QUE INTERVIENEN EN EL ACTO O CONTRATO (ARTICULO 17 LEY 1579 DE 2012).
139	EL DOCUMENTO DEL CUAL SE SOLICITA REPRODUCCION DE SELLOS NO SE ENCUENTRA INSCRITO (ARTICULOS 20 Y 21 LEY 1579 DE 2012).
140	EL FOLIO DE MATRICULA INMOBILIARIA ES INEXISTENTE POR ENCONTRARSE JURIDICAMENTE CERRADO (ARTICULO 8 LEY 1579 DE 2012).
141	FALTA CEDULA CATASTRAL DEL BIEN INMUEBLE (ARTICULO 18, 19 Y 20 DECRETO 2148 DE 1983, ARTICULO 31 DECRETO LEY 960/70, INSTRUCCION ADMINISTRATIVA 04 DE 2007 DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO, ARTICULO 50 LEY 1579 DE 2012)
142	DEBEN ANEXAR EL FORMULARIO DE CALIFICACION NOTARIAL EN LA COPIA DEL DOCUMENTO CON DESTINO A REGISTRO. (ARTICULO 94 DECRETO LEY 2150/95, PARAGRAFO 4DEL ARTICULO 8 LEY 1579 DE 2012).
143	PAZ Y SALVO DEL IMPUESTO PREDIAL PROTOCOLIZADO EN EL DOCUMENTO NO CORRESPONDE AL INMUEBLE OBJETO DEL NEGOCIO (ARTICULO 8 Y 50 LEY 1579 DE 2012).
144	EN EL DOCUMENTO NO SE ANEXO EL FORMATO DE CALIFICACION DEBIDAMENTE DILIGENCIADO (PARAGRAFO 4 DEL ARTICULO 8 LEY 1579 DE 2012)
150	* * * * AFECTACION A VIVIENDA FAMILIAR * * * *
151	EL INMUEBLE SE ENCUENTRA AFECTADO A VIVIENDA FAMILIAR (ARTICULO 3 LEY 258/96).
152	NO PROCEDE LA AFECTACION A VIVIENDA FAMILIAR POR CUANTO NO SE OSTENTA LA CALIDAD DE PLENO PROPIETARIO (LEY 258 DE 1996).
153	EL NOTARIO OMITIO INDAGAR SOBRE LA AFECTACION A VIVIENDA FAMILIAR (LEY 258 DE 1996 Y LEY 854 DE 2003).
154	PARA CANCELAR LA AFECTACION A VIVIENDA FAMILIAR DEBEN COMPARRECER Y SUSCRIBIR LA ESCRITURA AMBOS CONYUGES O COMPANEROS PERMANENTES (LEY 258/96 Y LEY 854/2003).

155	EL ADQUIRENTE OMITE LA CONSTITUCION DE LA AFECTACION A VIVIENDA FAMILIAR EN EL ACTO ESCRITURARIO DE LA COMPROVVENTA (ARTICULOS. 1 Y 2 LEY 258/96).
156	QUEDAN VICIADOS DE NULIDAD ABSOLUTA LOS ACTOS JURIDICOS QUE DESCONOCEN LA AFECTACION A VIVIENDA FAMILIAR. (INCISO 3 DEL ARTICULO 6 DE LA LEY 258/96).
157	EL INMUEBLE SOBRE EL CUAL SE PRETENDE CONSTITUIR AFECTACION A VIVIENDA FAMILIAR NO CUMPLE CON LOS REQUISITOS CONTEMPLADOS EN LA LEY 258/96.
158	EL ADQUIRIENTE OMITE LA CONSTITUCION DE LA AFECTACION A VIVIENDA FAMILIAR EN EL ACTO ESCRITURARIO DE LA COMPROVVENTA (ARTICULOS 1 Y 2 LEY 258/96).
159	LOS ACTOS JURIDICOS QUEDAN VICIADOS DE NULIDAD ABSOLUTA, CUANDO DESCONOCEN LA AFECTACION A VIVIENDA FAMILIAR. INCISO 3. ART. 6 LEY 258/96.
200	ESCRITURAS - OTORGAMIENTOS- AUTORIZACIONES - ACLARACIONES
201	FALTA FECHA DE OTORGAMIENTO DE LA ESCRITURA (ARTICULO 9 DECRETO 2148/83).
202	FALTA FIRMA DEL NOTARIO O SU DELEGADO EN LA COPIA DE LA ESCRITURA OBJETO DE REGISTRO (ARTICULO 85 DECRETO LEY 960/70).
203	FALTA LA AUTORIZACION (FIRMA) DEL NOTARIO EN EL ORIGINAL (MATRIZ) DE LA ESCRITURA PUBLICA (ARTICULO 40 DECRETO LEY 960/70).
204	LA REPRODUCCION MECANICA DE LAS COPIAS DEL DOCUMENTO QUE SE PRETENDE REGISTRAR, ES ILEGIBLE EN TODO O EN ALGUNOS DE SUS APARTES (PARAGRAFO 1DEL ARTICULO 14 LEY 1579 DE 2012).
205	FALTA OTORGAMIENTO, FIRMA DE UNO O VARIOS DE LOS COMPARECIENTES EN EL ORIGINAL DE LA ESCRITURA (ARTICULO 35 DECRETO LEY 960/70).
206	EL DOCUMENTO OBJETO DE ACLARACION DEBE RADICARSE PRIMERO QUE EL DOCUMENTO ACLARATORIO (LITERAL C DEL ARTICULO 3, ARTICULO 14 Y ARTICULO 49 DE LA LEY 1579 DE 2012).
207	EL DOCUMENTO QUE SE PRETENDE ACLARAR NO HA SIDO REGISTRADO (ARTICULO 29 LEY 1579 DE 2012).
208	LAS ENMENDADURAS DEBEN VENIR AVALADAS POR EL ENTE EMISOR (ARTICULO 101 DECRETO LEY 960/70, INSTRUCCION ADMINISTRATIVA 034 DE 1998 DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO).
209	UNA VEZ AUTORIZADA LA ESCRITURA, CUALQUIER CORRECCION QUE PRETENDAN HACER LOS OTORGANTES DEBERA CONSIGNARSE EN DOCUMENTO SEPARADO, CON TODAS LAS FORMALIDADES Y POR TODAS LAS PERSONAS QUE INTERVINIERON EN EL INSTRUMENTO CORREGIDO. (ARTICULO 102 DECRETO LEY 960/70).
210	<b>INACTIVO</b>
211	FALTA CITAR DOCUMENTO DE IDENTIFICACION DE LOS OTORGANTES (ARTICULO 24 Y 25 DECRETO LEY 960/70).
212	NO SE PUEDE POR ESCRITURA ACLARATORIA, VARIAR LOS ELEMENTOS ESENCIALES DEL CONTRATO INICIAL. (DECRETO 2148 DE 1983, DECRETO 231 DE 1985).
213	LA ESCRITURA QUE SE PRETENDE REGISTRAR NO CONTIENE LICENCIA URBANISTICA QUE AUTORICE LA CONSTRUCCION, AMPLIACION, MEJoras, SUBDIVISION, PARCELACION, LOTEo, DEL INMUEBLE. (ARTICULO. 7 DE LA LEY 810 DE 2003 Y 99 DE LA LEY 388 DE 1997).
214	EL TERMINO DE VIGENCIA DE LA LICENCIA OTORGADA PARA LA CONSTRUCCION, AMPLIACION, LOTEo O SUBDIVISION DEL PREDIO SE ENCUENTRA VENCIDA (ARTICULOS 47 Y SS DEL DECRETO 1469 DE 2010).
215	EL DOCUMENTO SOMETIDO A REGISTRO NO ESTABLECE CON CLARIDAD EL ACTO O CONTRATO QUE SE PRETENDE INSCRIBIR (ARTICULO 15 DECRETO LEY 960/70).
216	LA ESCRITURA SOMETIDA A REGISTRO NO CONTIENE EL PODER QUE ACREDITA LA CALIDAD EN QUE SE COMPARECE (ARTICULOS 25 Y 28 DEL DECRETO LEY 960/70).

217	TODAS LAS HOJAS DEL DOCUMENTO SOMETIDO A REGISTRO DEBEN SER RUBRICADAS Y SELLADAS POR EL EMISOR. (ARTICULO 85 DECRETO LEY 960/70)
218	LA LICENCIA DE PLANEACION Y OBRAS PUBLICAS MUNICIPALES NO CUMPLE CON LOS REQUISITOS ESTABLECIDOS EN EL DECRETO 1469 DE 2010.
219	TODA ACTA DE CONCILIACION QUE CONLLEVE TRANSFERENCIA DE DOMINIO DEBE ELEVARSE A ESCRITURA PUBLICA (INSTRUCCION ADMINISTRATIVA 05 DE FEBRERO 5 DE 2004 SUPERINTENDENCIA DE NOTARIADO Y REGISTRO, PARAGRAFO 1ARTICULO 4 LEY 1579 DE 2012).
300	* * * * PREDIO - LINDEROS - AREA * * * *
301	SE CAMBIA EL INMUEBLE OBJETO DEL NEGOCIO JURIDICO (ARTICULO 48 DECRETO 2148 DE 1983).
302	EXISTE INCONGRUENCIA ENTRE EL AREA Y/O LOS LINDEROS DEL PREDIO CITADOS EN EL PRESENTE DOCUMENTO Y LOS INSCRITOS EN EL FOLIO DE MATRICULA INMOBILIARIA Y/O ANTECEDENTES QUE SE ENCUENTRAN EN ESTA OFICINA DE REGISTRO (ARTICULO 8, PARAGRAFO 1 DEL ARTICULO 16, ARTICULO 29 Y ARTICULO 49 LEY 1579 DE 2012 E INSTRUCCION ADMINISTRATIVA CONJUNTA 01 Y 11 DEL IGAC Y SNR).
303	EL INMUEBLE NO SE DETERMINO POR SU AREA Y LINDEROS (ARTICULO 31 DECRETO 960/70, PARAGRAFO 1ARTICULO 16 LEY 1579 DE 2012).
304	LAS MEDIDAS DEL PREDIO DEBEN EXPRESARSE EN EL SISTEMA METRICO DECIMAL (DECRETO LEY 960/70, ARTICULOS 10 Y 30 RESOLUCION 1126 DE 1967 DEL MINISTERIO DE DESARROLLO ECONOMICO).
305	NO SE IDENTIFICA EL PREDIO RESULTANTE DEL ENGLOBE Y/O LOS PREDIOS OBJETO DE ENGLOBE (ARTICULO 19 DECRETO 2148 DE 1983).
306	NO SE IDENTIFICAN Y/O DETERMINAN LOS LOTES SEGREGADOS (ARTICULOS 1 Y 8 DECRETO 2148 DE 1983 Y 8 DEL DECRETO 2157 DE 1995).
307	SOLO SE PUEDE ENGLOBAR TENIENDO PLENO DOMINIO SOBRE LOS PREDIOS (ARTICULO 669 DEL CODIGO CIVIL).
308	EL CAMBIO DE AREA Y LINDEROS QUE CONLLEVE AUMENTO RESPECTO DE LOS INSCRITOS, DEBE HACERSE CON BASE EN CERTIFICACION O RESOLUCION, DE CATASTRO Y/O INSTITUTO GEOGRAFICO AGUSTIN CODAZZI-I.G.A.C., ELEVADA A ESCRITURA PUBLICA. (ARTICULO 5 DECRETO 1711 DE 1984).
309	FALTA IDENTIFICAR Y ALINDERAR LA PARTE RESTANTE DEL INMUEBLE (ARTICULO 8 DECRETO 2157 DE 1995).
310	LA CAUSAL QUE DIO ORIGEN A LA NEGATIVA DE REGISTRO DEL PRESENTE DOCUMENTO, DEBERA CORREGIRSE POR ESCRITURA PUBLICA ACLARATORIA (INCISO 1. Y PARTE INICIAL INCISO 2. ART.49 DCTO. 2148/83)
311	<b>INACTIVA</b>
312	LOS CENTROS DE CONCILIACION DEBEN ANEXAR CERTIFICADO DE VIGENCIA Y LA RESOLUCION QUE AUTORIZA SU FUNCIONAMIENTO. (ART. 67 LEY 23 DE 1991).
350	* * * DEL TITULAR DEL DERECHO - REPRESENTACION * * *
351	QUIEN TRANSFIERE NO ES TITULAR DEL DERECHO REAL DE DOMINIO (ARTICULO 29 LEY 1579 DE 2012 Y 669 DEL CODIGO CIVIL).
352	PARA EFECTUAR DIVISION MATERIAL, LOTEO, SEGREGACION O CUALQUIER OTRO ACTO DE DISPOSICION, DEBE OSTENTAR PLENO DOMINIO Y/O COMPARCER TODOS LOS TITULARES DEL DERECHO REAL DE DOMINIO. (ARTICULOS 669 Y 2340 DEL CODIGO CIVIL).
353	QUIEN COMPARCE COMO HEREDERO NO ESTA HABILITADO PARA DISPONER DEL INMUEBLE (DECRETO DE LA POSESION EFECTIVA DE LA HERENCIA - ARTICULO 757 CODIGO CIVIL).

354	QUIEN DISPONE DE LA TOTALIDAD DEL INMUEBLE SOLO ES TITULAR DE DERECHOS DE CUOTA (ARTICULO 669 CODIGO CIVIL).
355	QUIEN DISPONE DEL INMUEBLE NO ES PROPIETARIO, SOLAMENTE ES TITULAR DE DERECHOS Y ACCIONES (ARTICULO 669 CODIGO CIVIL.).
356	PARA EFECTUAR LA LIQUIDACION DE LA COMUNIDAD DEBEN COMPARRECER LA TOTALIDAD DE LOS COMUNEROS (NUMERAL 3 ARTICULO 2340 CODIGO CIVIL).
357	NO SE APORTA AUTORIZACION EXPRESA DEL MANDANTE, CONFORME LO ORDENADO EN EL ARTICULO 2170 DEL CODIGO CIVIL.
358	FALTA AUTORIZACION EXPRESA DE LA JUNTA DE SOCIOS PARA QUE EL REPRESENTANTE ADQUIERA EL BIEN INMUEBLE PARA SI MISMO. (ARTICULO 839 DEL CODIGO DE COMERCIO).
359	<b>INACTIVA</b>
360	FALTA EXPRESA AUTORIZACION DE LA JUNTA DE SOCIOS PARA QUE EL REPRESENTANTE LEGAL ADQUIERA PARA SI MISMO (ART. 839 CODIGO DEL COMERCIO).
361	<b>INACTIVA</b>
400	* * * EMBARGOS - VALORIZACION - PROHIBICIONES * * *
401	EL REGISTRADOR DE INSTRUMENTOS PUBLICOS NO REGISTRARA ESCRITURA ALGUNA DE ENAJENACION O DE CONSTITUCION DE HIPOTECA, CUANDO SOBRE EL PREDIO SE ENCUENTRE VIGENTE EMBARGO (ARTICULO 43 LEY 57 DE 1887 Y ARTICULO 34 LEY 1579 DE 2012).
402	LA MEDIDA CAUTELAR QUE ORDENA CANCELAR NO FIGURA REGISTRADA EN EL FOLIO DE MATRICULA INMOBILIARIA.(ARTICULOS 34 Y 62 LEY 1579 DE 2012).
403	EL EJECUTADO NO ES TITULAR INSCRITO (ARTICULO 839-1 ESTATUTO TRIBUTARIO).
404	POR CUANTO NO SE TRATA DE UN EJECUTIVO CON GARANTIA HIPOTECARIA, NO ES POSIBLE INSCRIBIR EL EMBARGO, YA QUE EL DEMANDADO NO ES TITULAR DEL DERECHO REAL DE DOMINIO (ARTICULO 681 DEL CODIGO DE PROCEDIMIENTO CIVIL.).
405	SOBRE EL BIEN INMUEBLE OBJETO DE EMBARGO SE ENCUENTRA VIGENTE PATRIMONIO DE FAMILIA. (ARTICULOS 21 Y 23 DE LA LEY 70 DE 1931 Y LEY 425 DE 1999).
406	SOBRE EL BIEN INMUEBLE OBJETO DE EMBARGO SE ENCUENTRA VIGENTE AFECTACION A VIVIENDA FAMILIAR (ARTICULO 7 LEY 258 DE 1996).
407	EN EL FOLIO DE MATRICULA INMOBILIARIA CITADO SE ENCUENTRA INSCRITO OTRO EMBARGO (ARTICULO 558 DEL CODIGO DE PROCEDIMIENTO CIVIL.).
408	NO SE APORTA CONSTANCIA DEL JUZGADO QUE ACREDITE INEXISTENCIA DE REMANENTES. (INCISOS 3 Y 5 DEL ARTICULO 543 DEL CODIGO DE PROCEDIMIENTO CIVIL).
409	FALTA AUTORIZACION DEL JUEZ O ACREDITOR PARA REGISTRAR EL DOCUMENTO. (ARTICULO 1521 DEL CODIGO CIVIL).
410	FALTA CERTIFICACION DEL JUZGADO QUE ACREDITE QUE NO SE HA DICTADO SENTENCIA FAVORABLE AL DEMANDANTE, DEBIDAMENTE EJECUTORIADA. (NUMERAL 2 ART. 691 C. DE P.C.).
411	LOS LUGARES DESTINADOS A CEMENTERIOS SON INEMBARGABLES. (NUMERAL 8 ART. 684 DEL C. DE P.C.).
412	EL EMBARGO QUE SE ORDENA CANCELAR FUE LEVANTADO OFICIOSAMENTE DE CONFORMIDAD CON LO DISPUESTO CON EL NUMERAL 1 DEL ARTICULO 558 DEL C. DE P.C.
413	LOS DERECHOS Y ACCIONES SON INEMBARGABLES. (INCISO 1 ART. 681 DEL C. DE P.C., INSTRUCCION ADMINISTRATIVA 21 DEL 15-09-2005 DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO).

414	EL EMBARGO DE MEJORAS NO ES OBJETO DE REGISTRO (NUMERAL 2 ART. 681 DEL C. DE P.C.).
415	LOS BIENES DE USO PUBLICO SON INEMBARGABLES (NUMERAL 1 ART. 684 DEL C. DE P.C.).
416	LOS DERECHOS PERSONALISIMOS E INTRANSFERIBLES DE USO Y HABITACION SON INEMBARGABLES (NUMERAL 14 ART. 684 DEL C. DE P.C.).
417	FALTA CITAR LOS DATOS DEL OFICIO CON EL CUAL SE COMUNICO LA MEDIDA CAUTELAR QUE SE CANCELA.(ARTICULOS 31,61 y 62 LEY 1579 DE 2012).
418	EN EL FOLIO DE MATRICULA INMOBILIARIA NO FIGURA INSCRITA GARANTIA REAL A FAVOR DEL DEMANDANTE. (ART. 558 DELC. DE P.C.).
419	EN EL FOLIO DE MATRICULA INMOBILIARIA SE ENCUENTRA VIGENTE PATRIMONIO DE FAMILIA Y/O AFECTACION A VIVIENDA FAMILIAR, NO SIENDO PROCEDENTE LA INSCRIPCION DEL EMBARGO DE ACCION MIXTA POR NO EXISTIR GARANTIA REAL VIGENTE INSCRITA. (LEY 258 DE 1996, LEY 70 DE 1931).
420	NO SON INSCRIBIBLES LOS EMBARGOS DECRETADOS SOBRE UN INMUEBLE AFECTADO A VIVIENDA FAMILIAR, SALVO QUE LA MEDIDA CAUTELAR TENGA SU ORIGEN EN UNA OBLIGACION HIPOTECARIA CUYA CONSTITUCION SEA ANTERIOR A LA AFECTACION A VIVIENDA FAMILIAR O EL INMUEBLE SE HUBIERA GRAVADO PARA GARANTIZAR PRESTAMOS DE ADQUISICION, CONSTRUCCION O MEJORA DE LA VIVIENDA FAMILIAR. (ARTS 7 LEY 258 DE 1996 Y 1 LEY 854 DE 2003).
421	EL PREDIO SE ENCUENTRA AFECTADO CON VALORIZACION (ART. 13 DECRETO 1604 DE 1966).
422	EL FOLIO DE MATRICULA INMOBILIARIA CITADO SE ENCUENTRA EN CUSTODIA POR ORDEN DE AUTORIDAD COMPETENTE RAZON POR LA CUAL NO ES VIABLE INSCRIBIR DOCUMENTO ALGUNO O EXPEDIR CERTIFICADO DE LIBERTAD Y TRADICION.
423	EN EL FOLIO DE MATRICULA INMOBILIARIA CITADO SE ENCUENTRA INSCRITA PROHIBICION JUDICIAL DE INSCRIPCION DE DOCUMENTOS (ARTICULO 32 LEY 1579 DE 2012).
424	EN EL FOLIO DE MATRICULA INMOBILIARIA CITADO SE ENCUENTRA INSCRITO OFICIO DE OFERTA DE COMPRA QUE DEJA EL BIEN FUERA DEL COMERCIO. (ART. 7 DECRETO 2400 DE 1989 REFORMA URBANA).
425	PREVIA LA INSCRIPCION DE LA ESCRITURA DE COMPROVENTA DEBE CANCELARSE LA OFERTA DE COMPRA. (INCISO 2 ART. 14 LEY 9 DE 1989).
426	NO ES PROCEDENTE LA CANCELACION DE LA MEDIDA CAUTELAR HASTA TANTO SE SUBSAN LA CAUSAL QUE ORIGINÓ LA DEVOLUCION DE LA PROVIDENCIA QUE ORDENA EL REMATE. (SENTENCIA DEL 13-06-1996 DEL TRIBUNAL SUPERIOR DEL DISTRITO JUDICIAL DE BOGOTA).
427	NO ES PROCEDENTE LA INSCRIPCION DEL EMBARGO SOBRE LOS INMUEBLES AFECTADOS POR RESOLUCION DE LA DIRECCION NACIONAL DE ESTUPEFACIENTES QUE ORDENA LA DESTINACION PROVISIONAL DE QUE TRATA EL ART. 55 DEL DECRETO LEY 99 DE 1991.
428	<b>INACTIVAS</b>
429	DE ACUERDO CON LO INFORMADO POR EL GRUPO OPERATIVO NO SE ENCONTRO MATRICULA INMOBILIARIA QUE IDENTIFIQUE BIEN(ES) INMUEBLE(S) A NOMBRE DE LA(S) PERSONA(S) CITADA(S) EN SU OFICIO. SE PROCEDIO A INCLUIR EL NOMBRE EN LA TABLA DE PROHIBICIONES PARA ENAJENAR.(ARTICULO 8 LEY 1579 DE 2012).
430	NO SON INSCRIBIBLES LOS TITULOS REFERIDOS A UN INMUEBLE AFECTADO EN PROCESO DE EXTINCIÓN DE COMINIO (ARTICULO 12 LEY 793 DE 2002).
500	* * * HIPOTECAS - PATRIMONIO DE FAMILIA * * *

501	QUIEN HIPOTECA NO ES TITULAR INSCRITO (ART. 2443 DEL C.C.)
502	LA HIPOTECA DEBE SER CANCELADA POR EL ACREDOR (ART. 49 Y 50 DECRETO LEY 960 DE 1970).
503	QUIEN CANCELA LA HIPOTECA NO CITA LA CALIDAD DE CESIONARIO (ART. 49 Y 50 DECRETO LEY 960 DE 1970).
504	FALTA DOCUMENTO O CARTA DONDE CONSTE EL MONTO DEL CREDITO APROBADO PARA LA HIPOTECA INDETERMINADA (ART. 15 DECRETO 1572 DE 1994, PARRAFO 2 ARTICULO 6 DECRETO 2289 DEL 23-06-2008).
505	SE ENCUENTRA VENCIDO EL TERMINO DE 90 DIAS PARA LA INSCRIPCION DE LA HIPOTECA Y/O PATRIMONIO DE FAMILIA, POR LO TANTO DEBERA CONSTITUIRSE NUEVAMENTE POR OTRO INSTRUMENTO PUBLICO. (ARTICULO 28 LEY 1579 DE 2012).
506	SE OMITEN PROTOCOLIZAR CON LA ESCRITURA EL DOCUMENTO CON EL CUAL LOS ENTES CREDITICIOS AUTORIZAN LA INSCRIPCION DE UNA SEGUNDA HIPOTECA (VIS). (ART. 38 LEY 3 DE 1991).
507	LA HIPOTECA QUE SE ORDENA CANCELAR NO APARECE REGISTRADA (ARTICULOS 61 Y 62 LEY 1579 DE 2012).
508	FALTA CONSIGNAR EN LA PRIMERA COPIA DEL INSTRUMENTO, QUE ESTA PRESTA MERITO EJECUTIVO. (ART.80 DECRETO LEY 960 DE 1970 Y PARAGRAFO 2 DEL ARTICULO 16 LEY 1579 DE 2012).
509	LA CESION DEL CREDITO HIPOTECARIO NO ES OBJETO DE REGISTRO (ARTICULO 4 LEY 1579 DE 2012).
510	PARA PROCEDER A LA CANCELACION DEL PATRIMONIO DE FAMILIA DEBE ADJUNTAR LA AUTORIZACION DEL ACREDOR HIPOTECARIO INSCRITO (ART. 22 LEY 546 DE 1999).
511	PARA PROCEDER A LA INSCRIPCION DE UNA NUEVA HIPOTECA SOBRE UN BIEN AFECTADO CON PATRIMONIO DE FAMILIA, DEBE EXPRESAR QUE ESTA SERA DESTINADA PARA LA MEJORA DE LA VIVIENDA O PAGO DE UN GRAVAMEN HIPOTECARIO ANTERIOR. (ART. 38 LEY 3 DE 1991).
512	<b>INACTIVA</b>
513	EL DOCUMENTO NO CONTIENE LOS DATOS COMPLETOS DEL INMUEBLE SOBRE EL CUAL SE SOLICITA CANCELAR LA HIPOTECA (PARAGRAFO 1 ARTICULO 16 Y ARTICULO 62 DE LA LEY 1579 DE 2012 Y ARTICULO 54 DECRETO LEY 960 DE 1970).
514	LA ESCRITURA DE CANCELACION DE HIPOTECA NO CONTIENE EL VALOR POR EL CUAL SE CONSTITUYO EL GRAVAMEN. (PARRAFO 5 ART. 6 DECRETO 2280 DE 2008).
515	LAS HIPOTECAS SOBRE DERECHOS Y ACCIONES NO SON REGISTRABLES. (ART. 669 DEL C.C.).
516	LOS ADQUIRENTES OMITEN CONSTITUIR PATRIMONIO DE FAMILIA EN EL ACTO ESCRITURARIO DE LA COMPRA DE VIVIENDA DE INTERES SOCIAL (ART. 60 LEY 9 DE 1989).
517	SOBRE EL PREDIO SE ENCUENTRA VIGENTE PATRIMONIO DE FAMILIA (ART. 22 LEY 70 DE 1931).
518	QUIEN CANCELA EL PATRIMONIO DE FAMILIA NO ES BENEFICIARIO DE EL NI PROPIETARIO DEL INMUEBLE SOBRE EL CUAL SE CONSTITUYO. (ART. 7 Y 23 LEY 70 DE 1931).
519	LA ORDEN JUDICIAL Y/O EXHORTO QUE ORDENA LA CANCELACION DEL GRAVAMEN HIPOTECARIO DEBE ELEVARSE A ESCRITURA PUBLICA (ART.52 DECRETO LEY 960 DE 1970).
550	* * * REGLAMENTO DE PROPIEDAD HORIZONTAL * * *
551	EL REGLAMENTO DE PROPIEDAD HORIZONTAL DEBE ELEVARSE A ESCRITURA PUBLICA (ART. 4 LEY 675 DE 2001).

552	SE OMITE PROTOCOLIZAR EL ACTA DE COPROPIETARIOS QUE AUTORIZA LA MODIFICACION AL REGLAMENTO (ART. 21 CONCORDANTE CON EL NUMERAL 6 DEL ART. 38 DE LA LEY 675 DE 2001).
553	NO SE CUMPLE CON LA TOTALIDAD DE LOS REQUISITOS PARA LA CONSTITUCION, MODIFICACION Y/O ADICION DEL REGLAMENTO DE PROPIEDAD HORIZONTAL (ARTS. 5 Y 6 DE LA LEY 675 DE 2001).
554	LA SUMATORIA TOTAL DE LOS COEFICIENTES DE COPROPIEDAD DEBE SER IGUAL AL 100% (ARTS. 4 Y 5 DE LA LEY 675 DE 2001).
555	COMO QUIERA QUE EL CONJUNTO SE DESARROLLARA POR ETAPAS, EL PRESENTE DOCUMENTO DEBE CONTENER LOS REQUISITOS ESTABLECIDOS EN EL ARTICULO 7 DE LA LEY 675 DE 2001.
556	SE OMITE PROTOCOLIZAR LA LICENCIA DE CONSTRUCCION O EL DOCUMENTO QUE (ART. 6 LEY 675 DE 2001).
557	FALTA CITAR LA TOTALIDAD DE LAS MATRICULAS INMOBILIARIAS QUE CONFORMAN LA PROPIEDAD HORIZONTAL SOBRE LA QUE SE PRETENDE REGISTRAR MODIFICACION, ACLARACION Y/O ADICION DEL MISMO (ARTICULO 8 LEY 1579 DE 2012).
558	NO SE INSERTO EL PAZ Y SALVO POR CONCEPTO DE ADMINISTRACION, NI SE DEJO CONSTANCIA DE LA SOLIDARIDAD DEL NUEVO PROPIETARIO EN RELACION CON LAS DEUDAS QUE EXISTAN CON LA COPROPIEDAD (ART. 29 LEY 675 DE 2001).
559	NO SE ANEXA LA APROBACION DE LOS PLANOS Y ALINDERAMIENTO, CUADRO DE AREAS APROBADOS POR LA CURADURIA URBANA. (DECRETO 1469 DE 2010).
560	LOS LOTES DONDE SE CONSTITUYE EL PRESENTE REGLAMENTO DE PROPIEDAD HORIZONTAL DEBEN SER PREVIAMENTE ENGLOBADOS (ART. 19 DECRETO 2148 DE 1983).
561	EN EL PRESENTE DOCUMENTO NO SE DETERMINAN LAS ZONAS OBJETO DE CESION OBLIGATORIA (LEY 388 DE 1997).
562	NO COMPARCE EL ACREDITOR HIPOTECARIO AUTORIZANDO LA EXTINCIÓN DE LA PROPIEDAD HORIZONTAL (ART. 9 LEY 675 DE 2001).
563	NO CUMPLE CON LOS REQUISITOS EXIGIDOS POR EL ARTICULO 9 DE LA LEY 675 DE 2001 PARA LA EXTINCIÓN DE LA PROPIEDAD HORIZONTAL.
564	NO SE CUMPLE CON EL QUORUM ESTABLECIDO POR LA LEY 675 DE 2006 EN SUS ARTICULOS 45 Y 46.
600	* * * PREDIOS RURALES - BALDIOS * * *
601	EL DOCUMENTO QUE SE PRESENTA A REGISTRO NO CUENTA CON LA AUTORIZACION EXPRESA DEL INCODER PARA ENAJENAR, GRAVAR O ARRIENDAR LA UAF, NI HAY DECLARACION JURAMENTADA DEL ADJUDICATARIO DE NO HABERLE SIDO NOTIFICADA LA DECISION DE AUTORIZACION DEL INCODER, TENIENDO EN CUENTA QUE ES UN PREDIO RURAL ADJUDICADO POR EL INCODER Y NO HAN TRANSCURRIDO LOS QUINCE ANOS CONTADOS DESDE LA PRIMERA ADJUDICACION QUE SE HIZO SOBRE LA RESPECTIVA PARCELA. ARTICULO 39 INCISO 3 Y 4 DE LA LEY 160 DE 1994.
602	EL DOCUMENTO QUE SE PRESENTA PARA REGISTRO CONTIENE LA TRANSMISION DE DOMINIO DE UN PREDIO RURAL ADQUIRIDO CON SUBSIDIO, SIN QUE HAYAN TRANSCURRIDO LOS DOCE ANOS SIGUIENTES A LA ADQUISICION Y NO SE PROTOCOLIZA AUTORIZACION EXPRESA Y ESCRITA DEL INCODER. PARAGRAFO DEL ARTICULO 25 LEY 160 DE 1994.
603	EL DOCUMENTO QUE SE PRESENTA PARA REGISTRO CONTIENE UNA ENAJENACION O GRAVAMEN DE UNA PARCELA ADJUDICADA BAJO EL REGIMEN DE LA UAF Y NO SE ACREDITA LA OFERTA PREVIA AL INCODER Y LA CONSTANCIA O PRUEBA DESU RECHAZO. ARTICULO 39, INCISO 6 Y 7 LEY 160 DE 1994.
604	EL DOCUMENTO QUE SE PRESENTA A REGISTRO CONTIENE UNA CESION DE DERECHOS

	DE CUOTA QUE SE TIENEN SOBRE UNA UAF Y NO SE CUENTA CON LA AUTORIZACION DEL INCODER PARA EFECTUARLA. INCISO 2 DEL NUMERAL 4 DEL ARTICULO 40 LEY 160 DE 1994
605	EL DOCUMENTO QUE SE PRESENTA A REGISTRO CONTIENE UNA DIVISION O FRACCIONAMIENTO DE UN INMUEBLE RURAL, CUYOS PREDIOS RESULTANTES TIENEN UNA CABIDA INFERIOR A LA DETERMINADA POR EL INCODER COMO UAF PARA ESE MUNICIPIO, TAMPoco HAY CONSTANCIA QUE LA ESCRITURA FUE AUTORIZADA POR EL INCODER POR TRATARSE DE UNA DE LAS EXCEPCLO 45 DE LA LEY 160/94. ARTICULO 44 LEY 160 DE 1994
606	NO SE PUEDE ADQUIRIR LA PROPIEDAD SOBRE TERRENOS INICIALMENTE ADJUDICADOS COMO BALDIOS SI LA CABIDA DEL PREDIO SUPERA LOS LIMITES MAXIMOS DE LA UNIDAD AGRICOLA FAMILIAR UAF. INCISO 9 ARTICULO 72 LEY 160 DE 1994.
607	NO SE PUEDE APORTAR A SOCIEDADES O COMUNIDADES LA PROPIEDAD DE TIERRAS QUE SE HUBIERAN ADJUDICADO COMO BALDIOS, SI CON ESE APORTE LA SOCIEDAD CONSOLIDA PROPIEDAD EN SUPERFICIE QUE EXCEDAN LA UNIDAD AGRICOLA FAMILIAR UAF. (INCISO 9 ARTICULO 72 LEY 160 DE 1994).
608	NO ES VIABLE LA INSCRIPCION DEL DOCUMENTO POR CUANTO NO HAN TRANSCURRIDO CINCO ANOS DESPUES DE LA ADJUDICACION Y LA HIPOTECA QUE CONTIENE NO EXPRESA QUE SEA PARA GARANTIZAR LAS OBLIGACIONES DERIVADAS DE CREDITOS AGROPECUARIOS OTORGADOS POR ENTIDADES FINANCIERAS. ARTICULO 73 LEY 160 DE 1994.
609	<b>INACTIVAR</b>
610	UNA VEZ ABIERTO UN FOLIO DE MATRICULA PROVENIENTE DE ADJUDICACION DE BALDIOS, SOLO PROCEDERA LA INSCRIPCION DE CUALQUIER ACTO SUJETO A REGISTRO CON LA AUTORIZACION EXPRESA DEL INCODER O QUIEN HAGA SUS VECES. (ARTICULO 58 LEY 1579 DEL 2012).
650	* * * FIDUCIA - LEASING * * *
651	NO SE INDICA EL VALOR ESTIPULADO COMO REMUNERACION DEL FIDUCIARIO PARA DETERMINAR COBRO DE DERECHOS DE REGISTRO (I.A. 17 DE 1994 SUPERNOTARIADO )
700	* * * * DEVOLUCION DOCUMENTOS * * * *
701	SEÑOR USUARIO TIENE UN PLAZO LIMITE DE DOS MESES CONTADOS A PARTIR DE LA FECHA DE RADICACION DEL DOCUMENTO PARA CANCELAR EL MAYOR VALOR. (RESOLUCION 5123 DEL 09-11-2000 Y DECRETO 2280 DE 2008).
702	SE REITERA EL CONTENIDO DE LA CAUSAL QUE ORIGINÓ LA NEGATIVA DEL REGISTRO DE ESTE DOCUMENTO CONSIGNADA EN LA DEVOLUCION ANTERIOR. COMO QUIERA QUE CONTRA LA DEVOLUCION ANTERIOR NO SE INTERPUSIERON LOS RECURSOS PREVISTOS EN EL ARTICULO 50 DEL CODIGO CONTENCIOSO ADMINISTRATIVO, SE ENTIENDE QUE LA DEVOLUCION POR ESA MISMA CAUSAL HA QUEDADO EN FIRME Y POR LO TANTO, CONTRA ESTA DECISION NO PROcede RECURSO ALGUNO, EN LA VIA GUBERNATIVA. LOS TERMINOS PARA SOLICITAR DEVOLUCION DE DINEROS RECAUDADOS POR CONCEPTO DE IMPUESTOS Y DERECHOS DE REGISTRO, CORREN A PARTIR DE LA PRIMERA DEVOLUCION.
703	NO SE HAN SUBSANADO LA TOTALIDAD DE LAS CAUSALES QUE DIERON ORIGEN A LA NEGATIVA DEL REGISTRO DEL PRESENTE DOCUMENTO CONSIGNADAS EN LA DEVOLUCION ANTERIOR.
704	EL DOCUMENTO SOMETIDO A REGISTRO NO CONTIENE LA LICENCIA JUDICIAL PARA EJERCER ACTOS DE DISPOSICION DE BIENES INMUEBLES DEL MENOR DE EDAD (ART. 649 Y SIGUIENTES C. DE P.C.).
800	* * * * OTROS * * * *

950	* * * ASUNTOS VARIOS (NO INCLUIDOS EN LAS CAUSALES ANTERIORES) * * *
951	SE REITERA EL CONTENIDO DE LA CAUSAL QUE ORIGINÓ LA NEGATIVA DEL REGISTRO DE ESTE DOCUMENTO CONSIGNADA EN DEVOLUCIÓN ANTERIOR.
952	NO SE HAN SUBSANADO LA TOTALIDAD DE LAS CAUSALES QUE DIERON LUGAR A LA NEGATIVA DEL REGISTRO DE ESTE DOCUMENTO CONSIGNADAS EN LA DEVOLUCIÓN ANTERIOR.
953	EXISTE MEDIDA CAUTELAR, ABSTENERSE DE TRAMITAR NUEVOS REGISTROS
999	SIN INFORMACIÓN

## DESPLIEGUE – 2.7.7

05/10/2012

### 002\_453\_ESTADISTICA DE CORRECCIONES EN TRAMITE-vimm

#### REQUERIMIENTO: ESTADISTICAS DE CORRECCIONES EN TRÁMITE

Se requiere crear un reporte con las siguientes características:

- 1) Nombre y Código del Círculo Registral el aplicativo debe traerlo por defecto
- 2) El título de este reporte es “ESTADÍSTICAS DE CORRECCIONES EN TRAMITE”
- 3) Rango de fecha Inicial: dd-mm-aaaa Fecha final dd-mm-aaaa el aplicativo debe permitir ingresar una fecha inicial y una final de la cual se requiere el reporte; la fecha inicial y la fecha final pueden ser la misma.
- 4) El aplicativo debe permitir seleccionar uno, varios o todos los usuarios del listado de usuarios de correcciones.
- 5) Fecha y hora de impresión
- 6) Usuario que genera el reporte – nombre y ID
- 7) Página
- 8) La información se debe mostrar agrupada por usuario, con Título “USUARIO :nombre
- 9) Debe tener las siguientes columnas con los siguientes títulos:  
 -Fases y/o Dependencias  
 -En el rango de fecha: En esta columna deben consolidar los totales de los turnos de correcciones que se encuentran en trámite y listados en cada una de las fases del proceso de correcciones y /o Dependencias, así mismo se debe tener en cuenta la fecha de llegada del turno a la fase y que se encuentre comprendida dentro del rango de fecha seleccionado. No se deben tener en cuenta para esta estadística las fases y/o dependencias ya cumplidas ó superadas.  
 -Fuera del rango de fecha seleccionado: En esta columna deben consolidar los totales de los turnos de corrección que se encuentran en trámite y listados en cada una de las fases del proceso de correcciones y/o Dependencias, sin tener en cuenta la fecha de llegada del turno a la fase, tampoco el rango de fechas seleccionado. No se deben tener en cuenta para esta estadística las fases y/o dependencias ya cumplidas ó superadas. El rango de fecha para este ítem no debe superar la fecha inicial del rango seleccionado. Ej: El rango de fecha seleccionado es del 1 de marzo al 15 de marzo de 2011, para este ítem debe generarse hasta el 28 de febrero de 2011.
- 10) Debe tener las siguientes filas con los siguientes títulos y totales de cada ítem:  
 -Correcciones en revisión y análisis  
 -Correcciones enviadas a Actuación administrativa  
 -Correcciones enviadas a Mayores Valores  
 -Correcciones enviadas a Antiguo Sistema  
 -Correcciones en Ejecutar Corrección Simple  
 -Correcciones en Revisar y Aprobar  
 -Correcciones en Mesa de Control de Correcciones  
 -Sub-Total de Correcciones en Trámite.  
 -Total Correcciones en Trámite: en esta fila se presentará el total correspondiente a la suma de los datos que se encuentran en la fila Sub-Total de Correcciones en Trámite en las columnas: “En el Rango de fechas” más “Fuera del Rango de Fechas”.

En cada una de los Items correspondiente a las fases y/o Dependencias se debe reflejar el total de los turnos de correcciones que se encuentran en trámite en cada una de las fases y /o dependencias a las cuales se han re-direccionado los turnos de corrección. Se debe tener en cuenta lo establecido en el punto nueve (9) para su respectiva presentación y totales.

- 11) Se debe presentar un resumen con título “Total Correcciones en Trámite por Fases”.

Debe contener las siguientes filas:

- Correcciones en Revisión y Análisis
- Correcciones enviadas a Actuación administrativa
- Correcciones enviadas a Mayores Valores
- Correcciones enviadas a Antiguo Sistema
- Correcciones en Ejecutar Corrección Simple
- Correcciones en Revisar y Aprobar
- Correcciones en Mesa de Control de Correcciones
- Sub-Total de Correcciones en Trámite**

En cada una de los Items correspondiente a las fases y/o Dependencias se debe reflejar el total de los turnos de correcciones que se encuentren en cada una de las fases y /o Dependencias a las cuales se han re direccinado los turnos de corrección; en las columnas correspondientes se debe presentar la sumatoria de los turnos de corrección que se encuentren en las diferentes fases, para los diferentes usuarios.

- 12) Se debe presentar un resumen con título “Totales Generales”; debe tener tres filas, con los siguientes títulos y en cada uno se debe presentar el total correspondiente al ítem de cada fila.

- Total Correcciones en Trámite:** en esta fila se presentará el total correspondiente a la suma de lo datos que se encuentran en la fila **Sub-Total de Correcciones en Trámite** de las columnas: “En el Rango de fechas” más “Fuera del Rango de Fechas”, del bloque identificado como “Total Correcciones en Trámite por Fases”.
- Total Correcciones Desanotadas Aprobadas en Este Rango de Fechas.**
- Total Correcciones Desanotadas Negadas en Este Rango de Fechas.**

Para los totales de correcciones DESANOTADAS APROBADAS y NO APROBADAS, se deben totalizar únicamente los turnos de correcciones que se encuentren en la fase CORRECCIÓN DESANOTADA dentro del rango de fechas seleccionado, y debe agruparse de acuerdo a la RESOLUCION (aprobada o negada).

**Se anexa cuadro guía para el diseño del reporte.**

**SUPERINTENDENCIA DE NOTARIADO Y REGISTRO  
MINISTERIO DE INTERIOR Y DE JUSTICIA  
REPÚBLICA DE COLOMBIA**

OFICINA DE REGISTRO DE II.PP DE BUCARAMANGA CÍRCULO REGISTRAL 300

INFORME DE DOCUMENTOS EN CORRECCIÓN

DEL: 01-02-2011 AL: 16-03-2011

Impreso el 16 de Marzo de 2011 a las 16:06:06

Usuario que genera reporte: ROSENDO.ROSAS

Página 1

Radicación	Fecha-Radica	Solicitante	Motivo-Solicitud	Matrícula	Fase	Usuario
2011-999-3-666	29/01/2011	JOSÉ BACCA MELO	CORREGIR NRO. ESCRITURA 546 POR ESCRITURA 546.	999-666666	REVISIÓN Y ANÁLISIS ROMO.HERRERA	

**TOTALES GENERALES**

<b>TOTAL CORRECCIONES EN TRÁMITE</b>	<b>51</b>	
<b>TOTAL CORRECCIONES DESANOTADAS APROBADAS EN ESTE RANGO DE FECHAS</b>	<b>20</b>	
<b>TOTAL CORRECCIONES DESANOTADAS NEGADAS EN ESTE RANGO DE FECHAS</b>	<b>3</b>	

#### **012\_453\_Error en el pago de turnos de registro\_MA**

GLPI 30488

Motivo Inconsistencia: Se comunica la funcionaria Ligia Suarez para reportar que en el reporte 25 generado el día 05/10/2011 para la usuaria YOLLY.URREA aparece una diferencia de - 50.000 para el turno 2011-314-6-7969.

Se genera el Reporte 25 del día 05/10/2011 para la usuaria YOLLY.URREA y se observa que en el turno 2011-314-6-7969. Aparece una diferencia de - \$ 50000 en el campo DIFERENCIA Vr. Docu. Pago / Vr. Aplicado, favor verificar el inconveniente que se está presentando al generar el reporte y revisar si existen más turnos asociado del día 05/10/2011 que puedan ocasionar el inconveniente, se anexa reporte.

#### **018\_453\_Desbloqueo\_folios\_con\_turnos\_fases\_desanotado**

Se solicita verificar en el sistema todos aquellos turnos de los procesos de Registro y correcciones que se encuentren en la fase desanotado y los folios asociados aun se encuentren bloqueados. Proceder a actualizar el bloqueo para que con estos folios se puedan expedir Certificados de Tradición.

Validaciones:

Para aquellos turnos de registro que se encuentren en la fase desanotado, sus folios asociados deben quedar desbloqueados.

#### **021\_453\_Desactivar\_Círculo\_Reparto\_Notarial**

Justificación: Para la ciudad de Villa de Leyva no se debe ejecutar reparto notarial, por cuanto allí solo existe una notaría

Descripción:

Se solicita la desactivación de la ejecución del reparto notarial a la ciudad de Villa de Leyva, la cual corresponde al círculo de registro de Tunja, lo anterior teniendo en cuenta que para este círculo notarial solamente existe una notaría y no aplica su ejecución.

#### **022\_453\_Campo\_documento\_identificación\_procesos\_registrales**

Justificación: Teniendo en cuenta el caso 32374 abierto en Mesa de Ayuda por la ORIP de Armenia, en el cuál informa que en la fase de Testamentos, en el bloque testador, al ingresar el número del Tipo de Documento de Identificación Pasaporte, para el testador, el Aplicativo SIR no permite ingresar la información alfanumérica del Tipo de Documento Pasaporte. Por lo expuesto se requiere que el sistema permita agregar la información alfanumérica del tipo de documento identificación Pasaporte en todos los procesos registrales.

Descripción:

Se requiere que el aplicativo SIR en cualquiera de los procesos registrales, al seleccionar el tipo de documento de identificación PASAPORTE, permita ingresar información alfanumérica.

Validaciones a Realizar: El aplicativo debe permitir, para el tipo de documento de identificación PASAPORTE,

ingresar información alfanumérica, en todos los Procesos Registrales.

#### **024\_453\_Creación de Usuario SNR\_Reparto Notarial**

##### **Justificación:**

Llevar el control sobre los repartos notariales ejecutados a nivel nacional en las Oficinas de Registro que operan con el aplicativo SIR.

##### **Descripción:**

Se solicita la creación de un usuario para la doctora Zoraida Celis Carrillo, cédula de ciudadanía 63301457 Directora de Gestión Notarial, que permita la consulta de los turnos de minutos e impresión de todos los reportes de reparto notarial de todas las Oficinas de Registro del país que ejecuta Reparto Notarial.

##### **Validaciones Que Realizar:**

El sistema debe permitir únicamente la consulta de minutos e impresión de reportes reparto notarial a nivel nacional.

#### **026\_453\_Historico\_Cambios\_Firma\_Registrador**

##### **Descripción:**

Se solicita que para la tabla SIR\_NE\_FIRMA\_REGISTRADOR, se actualice el campo FRRG\_FECHA\_CREACION y además se debe llevar un histórico del cambio de estados con las fechas de activación, desactivación y desactivación definitiva. Para llevar el histórico del cambio de estados se creara una tabla SIR\_NE\_FIRMA\_REG\_ESTADO con los siguientes campos:

ID\_FIRMA\_REGISTRADOR, FRRG\_ACTIVO (0-Inactivo, 1-Activo, 2-Inactivo Definitivo) y FRRG\_FECHA\_REG

##### **Validaciones a Realizar:**

Todas las firmas deben tener el campo FRRG\_FECHA\_CREACION diligenciado. Esta fecha se registrara al momento de hacer el cambio de estado de una firma y se reflejara a partir de que el desarrollo se despliegue en ambiente de producción.

#### **027\_453\_Imprimible\_Liquidacion\_Fotocopias**

##### **Justificación:**

Reflejar la normatividad correcta por la cual se establecen las tarifas para la Expedición de documentos que reposan en los archivos de las oficinas de registro.

##### **Descripción:**

Se requiere que el imprimible del proceso de Fotocopias “Liquidación de Documentos a ser Fotocopiados”, refleje la normatividad correcta por la cual se establecen las tarifas aplicadas para la expedición de documentos que reposan en los archivos de la Oficina de Registro.

En el párrafo de la normatividad, el texto correcto es:

“Según Resolución 0175 del 11 de enero de 2012, se cobran las siguientes tarifas:

- Documentos almacenados en medio óptico o microfilmado \$ 940.00 por cada página reproducida.
- Documentos que reposen en los archivos físicos de la respectiva oficina de registro \$590.00 por cada página fotocopiada.

De acuerdo con la Resolución 5123 del 2.000. Artículo 1. Parágrafo y Resolución 4907 de 2009. Artículo 1. “Se entendería que el interesado desiste de su petición cuando transcurridos dos meses contados a partir de la solicitud no acredite el pago respectivo de lo cual el Registrador dejaría expresa constancia en memorial petitorio de las copias”.

## **029\_453\_Salvedades\_Digitador\_Masivo\_Correcciones**

### **Justificación:**

A folios en estado temporal, asociados a turnos de correcciones que se encuentren en el proceso de Digitador Masivo, el sistema no debe obligar a incluir salvedad, cuando se ha agregado información en los campos de complementación, dirección y linderos.

### **Descripción:**

En la fase, Digitador Masivo del proceso de Correcciones, el sistema no debe obligar a incluir salvedad, cuando se incluye información de complementación, linderos y/o dirección en folios con estado temporal y que se encuentren asociados a turnos de correcciones en trámite en la fase mencionada.

### **Validaciones a Realizar:**

Que el folio se encuentre en estado temporal, asociado al turno de corrección mediante el cual se genero su apertura y en trámite en la fase Digitador Masivo del proceso de correcciones.

## **034\_453\_Número\_Documento\_Datos\_Alfanumericos**

### **Justificación:**

El aplicativo debe permitir la captura de datos alfanuméricos en el campo correspondiente al número del documento asociado a los diferentes tipos de documentos que se registran, en los procesos registrales:

- Proceso Registro - Rol Liquidador Registro - Pre liquidación y radicación turno de Registro,
- Proceso Correcciones - Rol Auxiliar correcciones – Al corregir el documento del folio o las anotaciones.
- Proceso Certificados - Rol Usuario Operativo - Radicación turno Certificados Exentos

### **Descripción:**

Se requiere que en la fase de solicitud de radicación Registro, en el bloque “Encabezado del Documento-Datos Básicos”, en el campo identificado como NUMERO, el sistema debe permitir ingresar datos alfanuméricos, dado que el número del documento asociado a los tipos de Documentos que se registran, pueden contener información alfanumérica y esta debe ser capturada como se encuentra registrada en los documentos físicos.

Este mismo ajuste debe ser aplicado en los siguientes procesos:

- Proceso Registro - Rol Liquidador Registro - Pre liquidación y radicación turno de Registro,
- Proceso Correcciones - Rol Auxiliar correcciones – Al corregir el documento del folio o las anotaciones.
- Proceso Certificados - Rol Usuario Operativo - Radicación turno Certificados Exentos

## **DESPLIEGUE – 2.7.6.1**

**19/07/2012**

## **071 Cambio\_textos\_notas\_devolutivas**

### **JUSTIFICACIÓN:**

De acuerdo a la solicitud enviada mediante comunicado SNRIE8646 del 06 de julio del año en curso, enviado por el doctor Néstor Raúl Espitia. Director de Registro, el cual se anexa, es necesario modificar el texto predefinido en el reporte de la NOTA DEVOLUTIVA.

### **DESCRIPCION DEL REQUERIMIENTO**

Actualizar el reporte de NOTAS DEVOLUTIVAS, que actualmente cita:

“CONTRA EL PRESENTE ACTO ADMINISTRATIVO, PROCEDE EL RECURSO DE REPOSICION ANTE EL REGISTRADOR DE INSTRUMENTOS PUBLICOS Y EN SUBSIDIO, EL DE APELACION ANTE EL DIRECTOR DE REGISTRO DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO DENTRO DE LOS CINCO (5) DIAS HABILES SIGUIENTES A SU NOTIFICACION, EN VIRTUD DE LO PREVISTO POR EL NUMERAL 2 DEL

ARTICULO 24 DEL DECRETO 2163 DEL 17 DE JUNIO DE 2011, PREVIO EL CUMPLIMIENTO DE LOS REQUISITOS ESTABLECIDOS EN LOS ARTICULOS 51 Y 52 DEL DECRETO 01 DE 1984 (CODIGO CONTENCIOSO ADMINISTRATIVO)."

Por el siguiente texto:

"CONTRA EL PRESENTE ACTO ADMINISTRATIVO, PROCEDE EL RECURSO ANTE EL REGISTRADOR DE INSTRUMENTOS PUBLICOS Y EN SUBSIDIO, EL DE APELACION ANTE EL DIRECTOR DE REGISTRO DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO DENTRO DE LOS DIEZ (10) DIAS HABILES SIGUIENTES A SU NOTIFICACION, EN VIRTUD DE LO PREVISTO POR EL NUMERAL 2 DEL ARTICULO 24 DEL DECRETO 2163 DEL 17 DE JUNIO DE 2011, PREVIO EL CUMPLIMIENTO DE LOS REQUISITOS ESTABLECIDOS EN LOS ARTICULOS 76 Y 77, LEY 1437 DE 2011 (CODIGO DE PROCEDIMIENTO ADMINISTRATIVO Y DE LO CONTENCIOSO ADMINISTRATIVO)."

## DESPLIEGUE – 2.7.6

22/06/2012

### 076 Implementación\_Transacciones

Se solicita implementar en la aplicación SIR el cálculo del tiempo de las acciones que se ejecutan en la radicación de turnos en los procesos de: certificados, consultas, correcciones, devoluciones, fotocopias, registro y reparto notarial.

Este cálculo se mostrara en el encabezado de los formularios de radicación y se debe almacenar en base de datos si se termina el proceso de radicación del turno.

Se debe crear un rol el cual tendrá acceso a un reporte que recibirá como parámetro el número del turno, si el turno tiene registros de transacción se desplegará el reporte con un detalle de las acciones ejecutadas en el proceso de radicación con su tiempo y al final un total del tiempo de las acciones.

#### Desarrollo del Requerimiento:

Se ejecutaron los siguientes Scripts:

- 1.IDX\_REP\_169\_SEC.sql
- 2.SCRIPTS\_REQ076\_151.sql
- 3.SEQ\_Reporte\_169.sql
- 4.SIR\_OP\_Reporte\_169.sql
- 5.SIR\_OP\_TRANSACCION.sql
- 6.SP\_Reporte\_169.sql

#### src\gov\sir\core\web\acciones\comun\AWPago.java

```
import co.com.iridium.generalSIR.comun.modelo.Transaccion;
import co.com.iridium.generalSIR.transacciones.TransaccionSIR;
import java.util.logging.Logger;
import java.util.logging.Level;
```

```
public Evento perform(HttpServletRequest request) throws AccionWebException {
```

#### LINEA 273

```
/*14/05/2012
TRANSACCION*/
request.getSession().setAttribute("TIEMPO_INICIO_TRANSACCION", new Date());
```

```
public void doEnd(HttpServletRequest request,
EventoRespuesta eventoRespuesta) {
```

#### LINEA 781

```
/*14/05/2012
TRANSACCION
REQ 076_151*/
Turno turno = null;
```

#### LINEA 794

```
Turno turno = respuesta.getTurno();
```

#### POR

```
/*14/05/2012
TRANSACCION
REQ 076_151*/
turno = respuesta.getTurno();
```

#### LINEA 965 - 1005

```
/*AHERRENO
14/05/2012
```

```

TRANSACCION*
Date fechalni = (Date) request.getSession().getAttribute("TIEMPO_INICIO_TRANSACCION");
Date fechaFin = new Date();
TransaccionSIR transaccion = new TransaccionSIR();
List <Transaccion> acciones = (List <Transaccion>) request.getSession().getAttribute("LISTA_TRANSACCION");
double tiempoSession = (Double) request.getSession().getAttribute("TIEMPO_TRANSACCION");
long calTiempo = 0;
try {
    calTiempo = transaccion.calculoTransaccion(fechalni, fechaFin);
} catch (Exception ex) {
    Logger.getLogger(AWPago.class.getName()).log(Level.SEVERE, null, ex);
}
DecimalFormat df = new DecimalFormat("0.000");
double calculo = Double.valueOf(df.format(tiempoSession+((double)calTiempo/1000)).replace(',', '.'));
System.out.println("El tiempo de la accion "+request.getParameter(WebKeys.ACCTION)+" en milisegundos "+calTiempo );
request.getSession().setAttribute("TIEMPO_TRANSACCION",calculo);

Transaccion transaccionReg = new Transaccion();
transaccionReg.setFechaTransaccion(fechaFin);
transaccionReg.setAccionWeb(accion);
transaccionReg.setTiempo(calTiempo);
acciones.add(transaccionReg);

Rol rol = (Rol) request.getSession().getAttribute(WebKeys.ROL);
if(rol.getNombre().equals("CAJERO CERTIFICADOS") || rol.getNombre().equals("LIQUIDADOR REGISTRO")||rol.getNombre().equals("CAJERO CONSULTAS")){
    if(accion.equals(PROCESAR)){
        /*Se recorre la lista para almacenar la informacion del turno*/
        for (Transaccion transacion: acciones) {
            transacion.setAnio(turno.getAnio());
            transacion.setIdCirculo(turno.getIdCirculo());
            transacion.setIdProceso(turno.getIdProceso());
            transacion.setIdTurno(turno.getIdTurno());
        }
        transaccion.guardarTransaccion(acciones);
        acciones.clear();
        request.getSession().setAttribute("LISTA_TRANSACCION",acciones);
        request.getSession().setAttribute("TIEMPO_TRANSACCION",Long.valueOf(0));
    }
}
}
}

```

#### **\src\gov\sir\core\web\acciones\comun\AWTurno.java**

```

import co.com.iridium.generalSIR.comun.modelo.Transaccion;
import java.util.logging.Level;
import java.util.logging.Logger;
import co.com.iridium.generalSIR.transacciones.TransaccionSIR;

```

#### **LINEA 142**

```

    /** Identifica la hora de inicio de la Transaccion de la acccion*/
    public static final String TIEMPO_INICIO_TRANSACCION = "TIEMPO_INICIO_TRANSACCION";

```

```

    /** Identifica el tiempo de la Transaccion de la acccion*/
    public static final String TIEMPO_TRANSACCION = "TIEMPO_TRANSACCION";

```

```

    /** Esta variable identifica el objeto con la lista de las acciones que
     * se han ejecuta en el proceso de radicacion de un turno*/
    public static final String LISTA_TRANSACCION = "LISTA_TRANSACCION";

```

```

    public Evento perform(HttpServletRequest request) throws AccionWebException {

```

#### **LINEA 162**

```

    /*AHERRENO
    14/05/2012
    REQ 076_151 TRANSACCION*/
    request.getSession().setAttribute(AWTurno.TIEMPO_INICIO_TRANSACCION, new Date());

```

```

    public void doEnd(HttpServletRequest request, EventoRespuesta evento) {

```

#### **LINEA 1069 - 1088**

```

    /*AHERRENO
    14/05/2012
    REQ 076_151 TRANSACCION*/
    Date fechalni = (Date) request.getSession().getAttribute("TIEMPO_INICIO_TRANSACCION");
    Date fechaFin = new Date();
    TransaccionSIR transaccion = new TransaccionSIR();
    List <Transaccion> acciones = (List <Transaccion>) request.getSession().getAttribute("LISTA_TRANSACCION");
    if (acciones == null){
        acciones = new ArrayList<Transaccion>();
    }
}
```

```

long calTiempo = 0;
try {
    calTiempo = transaccion.calculoTransaccion(fechalni, fechaFin);
} catch (Exception ex) {
    Logger.getLogger(AWTurno.class.getName()).log(Level.SEVERE, null, ex);
}
DecimalFormat df = new DecimalFormat("0.000");
double calculo = Double.valueOf(df.format((double)calTiempo/1000).replace('.', ','));
System.out.println("El tiempo de la accion "+request.getParameter(WebKeys.ACCION)+" en milisegundos "+calTiempo );
session.setAttribute(AWTurno.TIEMPO_TRANSACCION,calculo);
Transaccion transaccionReg = new Transaccion();
transaccionReg.setFechaTransaccion(fechaFin);
transaccionReg.setAccionWeb(acciona);
transaccionReg.setTiempo(calTiempo);
acciones.add(transaccionReg);
session.setAttribute(AWTurno.LISTA_TRANSACCION,acciones);

```

### **src\gov\sir\core\web\acciones\certificado\AWCertificado.java**

```

import co.com.iridium.generalSIR.comun.modelo.Transaccion;
import co.com.iridium.generalSIR.transacciones.TransaccionSIR;
import java.text.DecimalFormat;
import java.util.logging.Level;
import java.util.logging.Logger;
public Evento perform(HttpServletRequest request) throws AccionWebException {

```

**LINEA 232**

```

/*AHERRENO
14/05/2012
REQ 076_151 TRANSACCION*/
request.getSession().setAttribute("TIEMPO_INICIO_TRANSACCION", new Date());

```

### **public void doEnd(HttpServletRequest request, EventoRespuesta evento) {**

**LINEA 1130 - 1152**

```

/* AHERRENO
14/05/2012
REQ 076_151 TRANSACCION*/
Date fechalni = (Date) request.getSession().getAttribute("TIEMPO_INICIO_TRANSACCION");
double tiempoSession = (Double) request.getSession().getAttribute("TIEMPO_TRANSACCION");
Date fechaFin = new Date();
TransaccionSIR transaccion = new TransaccionSIR();
List <Transaccion> acciones = (List <Transaccion>) request.getSession().getAttribute("LISTA_TRANSACCION");
long calTiempo = 0;
try {
    calTiempo = transaccion.calculoTransaccion(fechalni, fechaFin);
} catch (Exception ex) {
    Logger.getLogger(AWCertificado.class.getName()).log(Level.SEVERE, null, ex);
}
DecimalFormat df = new DecimalFormat("0.000");
double calculo = Double.valueOf(df.format(tiempoSession+((double)calTiempo/1000)).replace('.', ','));
System.out.println("El tiempo de la accion "+request.getParameter("ACCION")+" en milisegundos "+calTiempo );
request.getSession().setAttribute("TIEMPO_TRANSACCION",calculo);
Transaccion transaccionReg = new Transaccion();
transaccionReg.setFechaTransaccion(fechaFin);
transaccionReg.setAccionWeb(acciona);
transaccionReg.setTiempo(calTiempo);
acciones.add(transaccionReg);
request.getSession().setAttribute("LISTA_TRANSACCION",acciones);

```

### **src\gov\sir\core\web\acciones\certificado\AWLiquidacionCertificado.java**

```

import co.com.iridium.generalSIR.comun.modelo.Transaccion;
import co.com.iridium.generalSIR.transacciones.TransaccionSIR;
import java.text.DecimalFormat;
import java.util.logging.Level;
import java.util.logging.Logger;

```

### **public Evento perform(HttpServletRequest request) throws AccionWebException {**

**LINEA 230**

```

/*AHERRENO
14/05/2012
REQ 076_151 TRANSACCION*/
request.getSession().setAttribute("TIEMPO_INICIO_TRANSACCION", new Date());

```

### **public void doEnd(HttpServletRequest request, EventoRespuesta evento) {**

**LINEA 1656 - 1693**

```

/* AHERRENO
14/05/2012
REQ 076_151 TRANSACCION*/
Date fechalni = (Date) request.getSession().getAttribute("TIEMPO_INICIO_TRANSACCION");
Date fechaFin = new Date();

```

```

TransaccionSIR transaccion = new TransaccionSIR();
List <Transaccion> acciones = (List <Transaccion>) request.getSession().getAttribute("LISTA_TRANSACCION");
double tiempoSession = (Double) request.getSession().getAttribute("TIEMPO_TRANSACCION");
long calTiempo = 0;
try {
    calTiempo = transaccion.calculoTransaccion(fechalni, fechaFin);
} catch (Exception ex) {
    Logger.getLogger(AWLiquidacionCertificado.class.getName()).log(Level.SEVERE, null, ex);
}
DecimalFormat df = new DecimalFormat("0.000");
double calculo = Double.valueOf(df.format(tiempoSession+((double)calTiempo/1000)).replace(',', '.'));
System.out.println("El tiempo de la accion "+accion+" en milisegundos "+calTiempo );
request.getSession().setAttribute("TIEMPO_TRANSACCION",calculo);
Transaccion transaccionReg = new Transaccion();
transaccionReg.setFechaTransaccion(fechaFin);
transaccionReg.setAccionWeb(accion);
transaccionReg.setTiempo(calTiempo);
acciones.add(transaccionReg);
request.getSession().setAttribute("LISTA_TRANSACCION",acciones);

if(accion.equals(RADICAR)){
    EvnRespLiquidacion turno = (EvnRespLiquidacion) evento;
    /*Se recorre la lista para almacenar la informacion del turno*/
    for (Transaccion transacion: acciones) {
        transacion.setAnio(turno.getTurno().getAnio());
        transacion.setIdCirculo(turno.getTurno().getIdCirculo());
        transacion.setIdProceso(turno.getTurno().getIdProceso());
        transacion.setIdTurno(turno.getTurno().getIdTurno());
    }
    transaccion.guardarTransaccion(acciones);
    acciones.clear();
    request.getSession().setAttribute("LISTA_TRANSACCION",acciones);
    request.getSession().setAttribute("TIEMPO_TRANSACCION",Long.valueOf(0));
}

```

### **src\gov\sir\core\web\helpers\comun\DatosPagoHelper.java**

```

import org.auriga.core.modelo.transferObjects.Rol;
LINEA 336
/*AHERRENO 17/05/2012
REQ 076_151 TRANSACCIONES*/
Rol rol = (Rol) request.getSession().getAttribute(WebKeys.ROL);
if(rol.getNombre().equals("CAJERO CERTIFICADOS") || rol.getNombre().equals("LIQUIDADOR REGISTRO")){
    out.write("<td width=\"120\" align=\"center\" valign=\"top\" background=\"" +
        request.getContextPath() +
        "/jsp/images/tabla_center_bgn002.gif\">");
    out.write("<table width=\"100%\" border=\"0\" cellspacing=\"0\" cellpadding=\"0\">");
    out.write("<tr >");
    out.write("<td ><img src=\"" +request.getContextPath()+
        "/jsp/images/ico_reloj.gif\" width=\"16\" height=\"21\"></td >");
    out.write("<td class=\"titulobcentral\">" +
        request.getSession().getAttribute("TIEMPO_TRANSACCION")+
        " Seg.</td>");
    out.write("</tr >");
    out.write("</table></td >");
}

```

### **src\gov\sir\core\web\acciones\consulta\AWLiquidacionConsulta**

```

import co.com.iridium.generalSIR.comun.modelo.Transaccion;
import co.com.iridium.generalSIR.transacciones.TransaccionSIR;
import java.text.DecimalFormat;
import java.util.Date;
import java.util.logging.Level;
import java.util.logging.Logger;
LINEA 104
public Evento perform(HttpServletRequest request) throws AccionWebException {
    /*AHERRENO
    24/05/2012
    REQ 076_151 TRANSACCION*/    request.getSession().setAttribute("TIEMPO_INICIO_TRANSACCION", new Date());
    public void doEnd(HttpServletRequest request, EventoRespuesta evento) {
LINEA 803 – 840
    /*AHERRENO
    24/05/2012
    REQ 076_151 TRANSACCION*/
    Date fechalni = (Date) request.getSession().getAttribute("TIEMPO_INICIO_TRANSACCION");
    double tiempoSession = (Double) request.getSession().getAttribute("TIEMPO_TRANSACCION");
    Date fechaFin = new Date();
    TransaccionSIR transaccion = new TransaccionSIR();
}

```

```

List <Transaccion> acciones = (List <Transaccion>) request.getSession().getAttribute("LISTA_TRANSACCION");
long calTiempo = 0;
try {
    calTiempo = transaccion.calculoTransaccion(fechalni, fechaFin);
} catch (Exception ex) {
    Logger.getLogger(AWLiquidacionConsulta.class.getName()).log(Level.SEVERE, null, ex);
}
DecimalFormat df = new DecimalFormat("0.000");
double calculo = Double.valueOf(df.format(tiempoSession+((double)calTiempo/1000)).replace(',', '.'));
System.out.println("El tiempo de la accion "+request.getParameter("ACCION")+" en milisegundos " +calTiempo );
request.getSession().setAttribute("TIEMPO_TRANSACCION",calculo);
Transaccion transaccionReg = new Transaccion();
transaccionReg.setFechaTransaccion(fechaFin);
transaccionReg.setAcciónWeb(request.getParameter("ACCION"));
transaccionReg.setTiempo(calTiempo);
acciones.add(transaccionReg);
request.getSession().setAttribute("LISTA_TRANSACCION",acciones);

if(request.getParameter("ACCION").equals("LIQUIDAR_CONSULTA_SIMPLE_SIMPLIFICADA")){
EvnRespLiquidacion turno = (EvnRespLiquidacion) evento;
/*Se recorre la lista para almacenar la informacion del turno*/
for (Transaccion transacion: acciones) {
    transacion.setAnio(turno.getTurno().getAnio());
    transacion.setIdCirculo(turno.getTurno().getIdCirculo());
    transacion.setIdProceso(turno.getTurno().getIdProceso());
    transacion.setIdTurno(turno.getTurno().getIdTurno());
}
transaccion.guardarTransaccion(acciones);
acciones.clear();
request.getSession().setAttribute("LISTA_TRANSACCION",acciones);
request.getSession().setAttribute("TIEMPO_TRANSACCION",Long.valueOf(0));
}

```

### src\gov\sir\core\web\acciones\corrección\AWLliquidacionCorrección.java

```

import co.com.iridium.generalSIR.comun.modelo.Transaccion;
import co.com.iridium.generalSIR.transacciones.TransaccionSIR;
import java.text.DecimalFormat;
import java.util.logging.Level;
import java.util.logging.Logger;
public Evento perform(HttpServletRequest request) throws AccionWebException {
LINEA138
/*AHERRENO
28/05/2012
REQ 076_151 TRANSACCION*/
request.getSession().setAttribute("TIEMPO_INICIO_TRANSACCION", new Date());
public void doEnd(HttpServletRequest request, EventoRespuesta evento) {
LINEA 1316 - 1353
/* AHERRENO
28/05/2012
REQ 076_151 TRANSACCION*/
Date fechalni = (Date) request.getSession().getAttribute("TIEMPO_INICIO_TRANSACCION");
double tiempoSession = (Double) request.getSession().getAttribute("TIEMPO_TRANSACCION");
Date fechaFin = new Date();
TransaccionSIR transaccion = new TransaccionSIR();
List <Transaccion> acciones = (List <Transaccion>) request.getSession().getAttribute("LISTA_TRANSACCION");
long calTiempo = 0;
try {
    calTiempo = transaccion.calculoTransaccion(fechalni, fechaFin);
} catch (Exception ex) {
    Logger.getLogger(AWLiquidacionCorrección.class.getName()).log(Level.SEVERE, null, ex);
}
DecimalFormat df = new DecimalFormat("0.000");
double calculo = Double.valueOf(df.format(tiempoSession+((double)calTiempo/1000)).replace(',', '.'));
System.out.println("El tiempo de la accion "+request.getParameter("ACCION")+" en milisegundos " +calTiempo );
request.getSession().setAttribute("TIEMPO_TRANSACCION",calculo);
Transaccion transaccionReg = new Transaccion();
transaccionReg.setFechaTransaccion(fechaFin);
transaccionReg.setAcciónWeb(request.getParameter("ACCION"));
transaccionReg.setTiempo(calTiempo);
acciones.add(transaccionReg);
request.getSession().setAttribute("LISTA_TRANSACCION",acciones);

if(request.getParameter("ACCION").equals("LIQUIDAR")){
EvnRespCorrección turno = (EvnRespCorrección) evento;
/*Se recorre la lista para almacenar la informacion del turno*/
for (Transaccion transacion: acciones) {

```

```

        transacion.setAnio(turno.getTurno().getAnio());
        transacion.setIdCirculo(turno.getTurno().getIdCirculo());
        transacion.setIdProceso(turno.getTurno().getIdProceso());
        transacion.setIdTurno(turno.getTurno().getIdTurno());
    }
    transaccion.guardarTransaccion(acciones);
    acciones.clear();
    request.getSession().setAttribute("LISTA_TRANSACCION",acciones);
    request.getSession().setAttribute("TIEMPO_TRANSACCION",Long.valueOf(0));
}

```

### **src\gov\sir\core\web\acciones\devolucion\AWLiquidacionDevolucion**

```

import co.com.iridium.generalSIR.comun.modelo.Transaccion;
import co.com.iridium.generalSIR.transacciones.TransaccionSIR;
import java.text.DecimalFormat;
import java.util.logging.Level;
import java.util.logging.Logger;
public Evento perform(HttpServletRequest request) throws AccionWebException {
LINEA 198
/*AHERRENO
28/05/2012
REQ 076_151 TRANSACCION*/
request.getSession().setAttribute("TIEMPO_INICIO_TRANSACCION", new Date());
public void doEnd(HttpServletRequest request, EventoRespuesta evento) {
LINEA 965 – 1002
/*AHERRENO
28/05/2012
REQ 076_151 TRANSACCION*/
Date fechalni = (Date) request.getSession().getAttribute("TIEMPO_INICIO_TRANSACCION");
double tiempoSession = (Double) request.getSession().getAttribute("TIEMPO_TRANSACCION");
Date fechaFin = new Date();
TransaccionSIR transaccion = new TransaccionSIR();
List <Transaccion> acciones = (List <Transaccion>) request.getSession().getAttribute("LISTA_TRANSACCION");
long calTiempo = 0;
try {
    calTiempo = transaccion.calculoTransaccion(fechalni, fechaFin);
} catch (Exception ex) {
    Logger.getLogger(AWLiquidacionDevolucion.class.getName()).log(Level.SEVERE, null, ex);
}
DecimalFormat df = new DecimalFormat("0.000");
double calculo = Double.valueOf(df.format(tiempoSession+((double)calTiempo/1000)).replace(',', '.'));
System.out.println("El tiempo de la accion "+request.getParameter("ACCION)+" en milisegundos "+calTiempo );
request.getSession().setAttribute("TIEMPO_TRANSACCION",calculo);
Transaccion transaccionReg = new Transaccion();
transaccionReg.setFechaTransaccion(fechaFin);
transaccionReg.setAccionWeb(request.getParameter("ACCION"));
transaccionReg.setTiempo(calTiempo);
acciones.add(transaccionReg);
request.getSession().setAttribute("LISTA_TRANSACCION",acciones);

if(request.getParameter("ACCION").equals("LIQUIDAR")){
    EvnRespPago turno = (EvnRespPago) evento;
    /*Se recorre la lista para almacenar la informacion del turno*/
    for (Transaccion transacion: acciones) {
        transacion.setAnio(turno.getTurno().getAnio());
        transacion.setIdCirculo(turno.getTurno().getIdCirculo());
        transacion.setIdProceso(turno.getTurno().getIdProceso());
        transacion.setIdTurno(turno.getTurno().getIdTurno());
    }
    transaccion.guardarTransaccion(acciones);
    acciones.clear();
    request.getSession().setAttribute("LISTA_TRANSACCION",acciones);
    request.getSession().setAttribute("TIEMPO_TRANSACCION",Long.valueOf(0));
}

```

### **src\gov\sir\core\web\acciones\fotocopia\AWLiquidacionFotocopia.java**

```

import co.com.iridium.generalSIR.comun.modelo.Transaccion;
import co.com.iridium.generalSIR.transacciones.TransaccionSIR;
import java.text.DecimalFormat;
import java.util.Date;
import java.util.logging.Level;
import java.util.logging.Logger;
public Evento perform( HttpServletRequest request ) throws AccionWebException {
LINEA 110
/*AHERRENO

```

<pre> 28/05/2012 REQ 076_151TRANSACCION/ request.getSession().setAttribute("TIEMPO_INICIO_TRANSACCION", new Date()); <b>public void doEnd(HttpServletRequest request, EventoRespuesta evento) {</b> </pre>
<b>LINEA 866 – 904</b> <pre> /*AHERRENO 28/05/2012 REQ 076_151 TRANSACCION/ Date fechalni = (Date) request.getSession().getAttribute("TIEMPO_INICIO_TRANSACCION"); double tiempoSession = (Double) request.getSession().getAttribute("TIEMPO_TRANSACCION"); Date fechaFin = new Date(); TransaccionSIR transaccion = new TransaccionSIR(); List &lt;Transaccion&gt; acciones = (List &lt;Transaccion&gt;) request.getSession().getAttribute("LISTA_TRANSACCION"); long calTiempo = 0; try {     calTiempo = transaccion.calculoTransaccion(fechalni, fechaFin); } catch (Exception ex) {     Logger.getLogger(AWLiquidacionFotocopia.class.getName()).log(Level.SEVERE, null, ex); } DecimalFormat df = new DecimalFormat("0.000"); double calculo = Double.valueOf(df.format(tiempoSession+((double)calTiempo/1000)).replace(',', '.')); System.out.println("El tiempo de la accion "+request.getParameter("ACCION")+" en milisegundos " +calTiempo ); request.getSession().setAttribute("TIEMPO_TRANSACCION",calculo); Transaccion transaccionReg = new Transaccion(); transaccionReg.setFechaTransaccion(fechaFin); transaccionReg.setAcciónWeb(request.getParameter("ACCION")); transaccionReg.setTiempo(calTiempo); acciones.add(transaccionReg); request.getSession().setAttribute("LISTA_TRANSACCION",acciones);  if(request.getParameter("ACCION").equals("LIQUIDAR")){     EvnRespFotocopia turno = (EvnRespFotocopia) evento;     /*Se recorre la lista para almacenar la informacion del turno*/     for (Transaccion transacion: acciones) {         transacion.setAnio(turno.getTurno().getAnio());         transacion.setIdCirculo(turno.getTurno().getIdCirculo());         transacion.setIdProceso(turno.getTurno().getIdProceso());         transacion.setIdTurno(turno.getTurno().getIdTurno());     }     transaccion.guardarTransaccion(acciones);     acciones.clear();     request.getSession().setAttribute("LISTA_TRANSACCION",acciones);     request.getSession().setAttribute("TIEMPO_TRANSACCION",Long.valueOf(0)); } } </pre>
<b>src\gov\sir\core\web\acciones\registro\AWLiquidacionRegistro.java</b> <pre> import co.com.iridium.generalSIR.comun.modelo.Transaccion; import co.com.iridium.generalSIR.transacciones.TransaccionSIR; import java.text.DecimalFormat; import java.util.logging.Level; import java.util.logging.Logger; <b>public Evento perform(HttpServletRequest request) throws AccionWebException {</b> </pre>
<b>LINEA 507</b> <pre> /*AHERRENO 29/05/2012 REQ 076_151 TRANSACCION/ request.getSession().setAttribute("TIEMPO_INICIO_TRANSACCION", new Date()); <b>public void doEnd(HttpServletRequest request, EventoRespuesta respuesta) {</b> </pre>
<b>LINEA 4526 – 4567</b> <pre> /*AHERRENO 28/05/2012 REQ 076_151 TRANSACCION/ Date fechalni = (Date) request.getSession().getAttribute("TIEMPO_INICIO_TRANSACCION"); double tiempoSession = (Double) request.getSession().getAttribute("TIEMPO_TRANSACCION"); Date fechaFin = new Date(); TransaccionSIR transaccion = new TransaccionSIR(); List &lt;Transaccion&gt; acciones = (List &lt;Transaccion&gt;) request.getSession().getAttribute("LISTA_TRANSACCION"); long calTiempo = 0; try {     calTiempo = transaccion.calculoTransaccion(fechalni, fechaFin); } catch (Exception ex) {     Logger.getLogger(AWLiquidacionRegistro.class.getName()).log(Level.SEVERE, null, ex); } DecimalFormat df = new DecimalFormat("0.000"); double calculo = Double.valueOf(df.format(tiempoSession+((double)calTiempo/1000)).replace(',', '.')); </pre>

```

System.out.println("El tiempo de la accion "+request.getParameter("ACCION")+" en milisegundos " +calTiempo );
request.getSession().setAttribute("TIEMPO_TRANSACCION",calculo);
Transaccion transaccionReg = new Transaccion();
transaccionReg.setFechaTransaccion(fechaFin);
transaccionReg.setAcciónWeb(request.getParameter("ACCION"));
transaccionReg.setTiempo(calTiempo);
acciones.add(transaccionReg);
request.getSession().setAttribute("LISTA_TRANSACCION",acciones);

if(request.getParameter("ACCION").equals("CREAR_TURNO")){
EvnRespLiquidacionRegistro eventoResp = (EvnRespLiquidacionRegistro) respuesta;
Turno turno = null;
if(eventoResp.getPayload() instanceof Turno){
turno = (Turno)eventoResp.getPayload();
/*Se recorre la lista para almacenar la informacion del turno*/
for (Transaccion transacion: acciones) {
transacion.setAnio(turno.getAnio());
transacion.setIdCirculo(turno.getIdCirculo());
transacion.setIdProceso(turno.getIdProceso());
transacion.setIdTurno(turno.getIdTurno());
}
transaccion.guardarTransaccion(acciones);
acciones.clear();
request.getSession().setAttribute("LISTA_TRANSACCION",acciones);
request.getSession().setAttribute("TIEMPO_TRANSACCION",Long.valueOf(0));
}
}
}

```

### src\gov\sir\core\web\acciones\repartonotarial\AWLliquidacionRepartoNotarial.java

```

import co.com.iridium.generalSIR.comun.modelo.Transaccion;
import co.com.iridium.generalSIR.transacciones.TransaccionSIR;
import gov.sir.core.negocio.modelo.Turno;
import java.text.DecimalFormat;
import java.util.Date;
import java.util.logging.Level;
import java.util.logging.Logger;
public Evento perform(HttpServletRequest request) throws AccionWebException {
LINEA 120
/* AHERRENO
29/05/2012
REQ 076_151 TRANSACCION*/
request.getSession().setAttribute("TIEMPO_INICIO_TRANSACCION", new Date());
public void doEnd(HttpServletRequest request, EventoRespuesta evento) {
LINEA 947 – 988
/*AHERRENO
28/05/2012
REQ 076_151 TRANSACCION*/
Date fechalni = (Date) request.getSession().getAttribute("TIEMPO_INICIO_TRANSACCION");
double tiempoSession = (Double) request.getSession().getAttribute("TIEMPO_TRANSACCION");
Date fechaFin = new Date();
TransaccionSIR transaccion = new TransaccionSIR();
List <Transaccion> acciones = (List <Transaccion>) request.getSession().getAttribute("LISTA_TRANSACCION");
long calTiempo = 0;
try {
calTiempo = transaccion.calculoTransaccion(fechalni, fechaFin);
} catch (Exception ex) {
Logger.getLogger(AWLiquidacionRepartoNotarial.class.getName()).log(Level.SEVERE, null, ex);
}
DecimalFormat df = new DecimalFormat("0.000");
double calculo = Double.valueOf(df.format(tiempoSession+((double)calTiempo/1000)).replace(',', '.'));
System.out.println("El tiempo de la accion "+request.getParameter("ACCION")+" en milisegundos " +calTiempo );
request.getSession().setAttribute("TIEMPO_TRANSACCION",calculo);
Transaccion transaccionReg = new Transaccion();
transaccionReg.setFechaTransaccion(fechaFin);
transaccionReg.setAcciónWeb(request.getParameter("ACCION"));
transaccionReg.setTiempo(calTiempo);
acciones.add(transaccionReg);
request.getSession().setAttribute("LISTA_TRANSACCION",acciones);

if(request.getParameter("ACCION").equals("LIQUIDAR")){
EvnRespRepartoNotarial eventoResp = (EvnRespRepartoNotarial) respuesta;
Turno turno = null;
if(eventoResp instanceof EvnRespRepartoNotarial){
turno = (Turno)eventoResp.getTurno();
/*Se recorre la lista para almacenar la informacion del turno*/
for (Transaccion transacion: acciones) {
}
}
}
}

```

```

        transacion.setAnio(turno.getAnio());
        transacion.setIdCirculo(turno.getIdCirculo());
        transacion.setIdProceso(turno.getIdProceso());
        transacion.setIdTurno(turno.getIdTurno());
    }
    transaccion.guardarTransaccion(acciones);
    acciones.clear();
    request.getSession().setAttribute("LISTA_TRANSACCION",acciones);
    request.getSession().setAttribute("TIEMPO_TRANSACCION",Long.valueOf(0));
}
}

```

## \SIR-APP\web\WEB-INF\jspSeguro\certificado\ SOLICITUD\_CERTIFICADO.jsp

### LINEA 351

```

<!--AHERRENO 17/05/2012
REQ 076_151 TRANSACCIONES-->
<td width="120" align="center" valign="top" background="<%=" request.getContextPath()%>/jsp/images/tabla_central_bgn002.gif">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td>/jsp/images/ico_reloj.gif" width="16" height="21"></td>
<td class="titulotbcentral"><%= request.getSession().getAttribute("TIEMPO_TRANSACCION")%> Seg.</td>
</tr>
</table></td>

```

## \SIR-APP\web\WEB-INF\jspSeguro\certificado\SOLICITUD\_CERTIFICADO\_SIMPLIFICADO.jsp

### SE MODIFICA LINEA 302

```

<!--AHERRENO 17/05/2012
REQ 076_151 TRANSACCIONES-->
<td width="120" align="center" valign="top" background="<%=" request.getContextPath()%>/jsp/images/tabla_central_bgn002.gif">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td>/jsp/images/ico_reloj.gif" width="16" height="21"></td>
<td class="titulotbcentral"><%= request.getSession().getAttribute("TIEMPO_TRANSACCION")%> Seg.</td>
</tr>
</table></td>

```

## \SIR-APP\web\WEB-INF\jspSeguro\consulta\CONSULTA.jsp

### LINEA 173

```

<!--AHERRENO 24/05/2012
REQ 076_151 TRANSACCIONES-->
<td width="120" align="center" valign="top" background="<%=" request.getContextPath()%>/jsp/images/tabla_central_bgn002.gif">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td>/jsp/images/ico_reloj.gif" width="16" height="21"></td>
<td class="titulotbcentral"><%= request.getSession().getAttribute("TIEMPO_TRANSACCION")%> Seg.</td>
</tr>
</table></td>

```

## \SIR-APP\web\WEB-INF\jspSeguro\consulta\ CONSULTA\_SIMPLIFICADA.jsp

### LINEA 154

```

<!--AHERRENO 24/05/2012
REQ 076_151 TRANSACCIONES-->
<td width="120" align="center" valign="top" background="<%=" request.getContextPath()%>/jsp/images/tabla_central_bgn002.gif">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td>/jsp/images/ico_reloj.gif" width="16" height="21"></td>
<td class="titulotbcentral"><%= request.getSession().getAttribute("TIEMPO_TRANSACCION")%> Seg.</td>
</tr>
</table></td>

```

## \SIR-APP\web\WEB-INF\jspSeguro\correccion\ SOLICITUD\_CORRECCION.jsp

### LINEA 124

```

<!--AHERRENO 28/05/2012
REQ 076_151 TRANSACCIONES-->
<td width="120" align="center" valign="top" background="<%=" request.getContextPath()%>/jsp/images/tabla_central_bgn002.gif">
<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td>/jsp/images/ico_reloj.gif" width="16" height="21"></td>
<td class="titulotbcentral"><%= request.getSession().getAttribute("TIEMPO_TRANSACCION")%> Seg.</td>
</tr>
</table></td>

```

## \SIR-APP\web\WEB-INF\jspSeguro\devolucion\ SOLICITUD\_DEVOLUCION.jsp

<b>LINEA 180</b>	<pre> &lt;!--AHERRENO 28/05/2012 REQ 076_151 TRANSACCIONES--&gt; &lt;td width="120" align="center" valign="top" background="&lt;%= request.getContextPath()%&gt;/jsp/images/tabla_central_bgn002.gif"&gt; &lt;table width="100%" border="0" cellspacing="0" cellpadding="0"&gt; &lt;tr&gt; &lt;td&gt;&lt;img src="&lt;%= request.getContextPath()%&gt;/jsp/images/ico_reloj.gif" width="16" height="21"&gt;&lt;/td&gt; &lt;td class="titulotbcentral"&gt;&lt;%= request.getSession().getAttribute("TIEMPO_TRANSACCION")%&gt; Seg.&lt;/td&gt; &lt;/tr&gt; &lt;/table&gt;&lt;/td&gt; </pre>
<b>\SIR-APP\web\WEB-INF\jspSeguro\fotocopia\ LIQUIDACION_FOTOCOPIA.jsp</b>	
<b>LINEA 232</b>	<pre> &lt;!--AHERRENO 28/05/2012 REQ 076_151 TRANSACCIONES--&gt; &lt;td width="120" align="center" valign="top" background="&lt;%= request.getContextPath()%&gt;/jsp/images/tabla_central_bgn002.gif"&gt; &lt;table width="100%" border="0" cellspacing="0" cellpadding="0"&gt; &lt;tr&gt; &lt;td&gt;&lt;img src="&lt;%= request.getContextPath()%&gt;/jsp/images/ico_reloj.gif" width="16" height="21"&gt;&lt;/td&gt; &lt;td class="titulotbcentral"&gt;&lt;%= request.getSession().getAttribute("TIEMPO_TRANSACCION")%&gt; Seg.&lt;/td&gt; &lt;/tr&gt; &lt;/table&gt;&lt;/td&gt; </pre>
<b>\SIR-APP\web\WEB-INF\jspSeguro\registro\REGISTRO_DOCUMENTOS.jsp</b>	
<b>LINEA 494</b>	<pre> &lt;!--AHERRENO 17/05/2012 REQ 076_151 TRANSACCIONES--&gt; &lt;td width="120" align="center" valign="top" background="&lt;%= request.getContextPath()%&gt;/jsp/images/tabla_central_bgn002.gif"&gt; &lt;table width="100%" border="0" cellspacing="0" cellpadding="0"&gt; &lt;tr&gt; &lt;td&gt;&lt;img src="&lt;%= request.getContextPath()%&gt;/jsp/images/ico_reloj.gif" width="16" height="21"&gt;&lt;/td&gt; &lt;td class="titulotbcentral"&gt;&lt;%= request.getSession().getAttribute("TIEMPO_TRANSACCION")%&gt; Seg.&lt;/td&gt; &lt;/tr&gt; &lt;/table&gt;&lt;/td&gt; </pre>
<b>\SIR-APP\web\WEB-INF\jspSeguro\registro\REGISTRO_LIQUIDAR.jsp</b>	
<b>LINEA 157</b>	<pre> &lt;!--AHERRENO 17/05/2012 REQ 076_151 TRANSACCIONES--&gt; &lt;td width="120" align="center" valign="top" background="&lt;%= request.getContextPath()%&gt;/jsp/images/tabla_central_bgn002.gif"&gt; &lt;table width="100%" border="0" cellspacing="0" cellpadding="0"&gt; &lt;tr&gt; &lt;td&gt;&lt;img src="&lt;%= request.getContextPath()%&gt;/jsp/images/ico_reloj.gif" width="16" height="21"&gt;&lt;/td&gt; &lt;td class="titulotbcentral"&gt;&lt;%= request.getSession().getAttribute("TIEMPO_TRANSACCION")%&gt; Seg.&lt;/td&gt; &lt;/tr&gt; &lt;/table&gt;&lt;/td&gt; </pre>
<b>\SIR-APP\web\WEB-INF\jspSeguro\registro\REGISTRO_DOCUMENTOS_RADICACION.jsp</b>	
<b>LINEA 171</b>	<pre> &lt;!--AHERRENO 03/06/2012 REQ 076_151 TRANSACCIONES--&gt; &lt;td width="120" align="center" valign="top" background="&lt;%= request.getContextPath()%&gt;/jsp/images/tabla_central_bgn002.gif"&gt; &lt;table width="100%" border="0" cellspacing="0" cellpadding="0"&gt; &lt;tr&gt; &lt;td&gt;&lt;img src="&lt;%= request.getContextPath()%&gt;/jsp/images/ico_reloj.gif" width="16" height="21"&gt;&lt;/td&gt; &lt;td class="titulotbcentral"&gt;&lt;%= request.getSession().getAttribute("TIEMPO_TRANSACCION")%&gt; Seg.&lt;/td&gt; &lt;/tr&gt; &lt;/table&gt;&lt;/td&gt; </pre>
<b>\SIR-APP\web\WEB-INF\jspSeguro\repartonotarial\SOLICITUD_REPARTO_NOTARIAL.jsp</b>	
<b>LINEA 225</b>	<pre> &lt;!--AHERRENO 17/05/2012 REQ 076_151 TRANSACCIONES--&gt; &lt;td width="120" align="center" valign="top" background="&lt;%= request.getContextPath()%&gt;/jsp/images/tabla_central_bgn002.gif"&gt; &lt;table width="100%" border="0" cellspacing="0" cellpadding="0"&gt; &lt;tr&gt; &lt;td&gt;&lt;img src="&lt;%= request.getContextPath()%&gt;/jsp/images/ico_reloj.gif" width="16" height="21"&gt;&lt;/td&gt; </pre>

<td class="titulotbcentral"><%= request.getSession().getAttribute("TIEMPO_TRANSACCION")%> Seg.</td>
</tr>
</table></td>
<b>NUEVO REPORTE - INFORME TRANSACCION POR TURNO</b>
Se adjunta al caso reporte-169.jrxml
<b>SIR-APP\src\org.auriga.smart.vistas.xml</b>
<!-- AHERRENO 01/06/2011 report-169 -->
<vista> <autenticacion>true</autenticacion>  <excepcion>gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector</excepcion> <excepcion>gov.sir.core.web.acciones.excepciones.ValidacionParametrosException</excepcion> <nOMBRE>admin.reportes.report-169.param-form.view</nOMBRE> <parametro><llave>error</llave><valor>/WEB-INF/jspSeguro/comun/ERROR_MULTIPLE.jsp</valor></parametro> <parametro><llave>head</llave><valor>/WEB-INF/jspSeguro/comun/HEADER_LOGOUT.jsp</valor></parametro> <parametro><llave>body</llave><valor>/WEB- INF/jspSeguro/administracion/ADMINREPORTES_REPORT169_PARAMFORM.jsp</valor></parametro> <parametro><llave>bottom</llave><valor>/WEB-INF/jspSeguro/comun/FOOTER_DEFAULT.jsp</valor></parametro> </vista>
<b>SIR-APP\web\WEB-INF\jspSeguro\administracion\ADMINREPORTES_REPORT169_PARAMFORM</b>
Archivo Nuevo
<b>SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java</b>
<b>LINEA 321</b>
// AHERRENO 01/06/2012 // Reporte 169 public static final String REPORTE_169_SEND_ACTION = "REPORTE_169_SEND_ACTION"; public static final String REPORTE_169_BACK_ACTION = "REPORTE_169_BACK_ACTION";
<b>LINEA 1188</b>
// AHERRENO 01/06/2012 // Declaración de las Variables para el reporte 169 public static final String REPORTE_169_PARAM_CMDKEY = "reporte-169.jasper"; public static final String REPORTE_169_PARAM_TURNO = "P_TURNO"; public static final String REPORTE_169_PARAM_USUARIOLOG = "P_USUARIO_LOG";
<b>LINEA 2568</b>
// AHERRENO 01/06/2012 // Reporte 169 if (REPORTE_169_SEND_ACTION.equals(accion)) { do_Reporte169_Send_Action(request); } if (REPORTE_169_BACK_ACTION.equals(accion)) { do_Reporte169_Back_Action(request); }
<b>LINEA 13815 - 13910</b>
<pre>/*  * @param request  * Reporte 169 - INFORME TRANSACCIONES POR TURNO  * AHERRENO 01/06/2012  */ private void do_Reporte169_Send_Action(HttpServletRequest request) throws ValidacionParametrosException {      ValidacionParametrosException exception =         new ValidacionParametrosException();      String param_01 = request.getParameter(REPORTE_169_PARAM_TURNO).trim();     String param_02 = request.getParameter(REPORTE_169_PARAM_USUARIOLOG);      // apply validators     if ((null == param_01)    ("".equalsIgnoreCase(param_01))) {         exception.addError("parametro: Turno: Debe digitar un valor");     }      // apply validators     if ((null == param_02)    ("".equalsIgnoreCase(param_02))) {         exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");     } else {         if (WebKeys.SIN_SELECCIONAR.equals(param_02)) {             exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");         }     } }</pre>

```

        }

        do_Reporte169_SaveState(request);

        if (exception.getErrores().size() > 0) {
            throw exception;
        }

        String[] params = new String[]{
            REPORTE_XX_PARAM_CMDKEY,
            REPORTE_169_PARAM_TURNO,
            REPORTE_169_PARAM_USUARIOLOG
        };

        String[] values = new String[]{
            REPORTE_169_PARAM_CMDKEY,
            param_01,
            param_02
        };

        String url = makeUrl(params, values);
        reportsServices_ConfigureParameterUri(request, url);

        do_Reporte169_RemoveState(request);
        do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);

    }

    private void do_Reporte169_Back_Action(HttpServletRequest request) {

        do_Reporte169_RemoveState(request);
        do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);

    } // :do_Reporte169_Back_Action

    private void do_Reporte169_SaveState(HttpServletRequest request) {

        HttpSession session;

        String[] itemIds = new String[]{
            REPORTE_169_PARAM_TURNO,
            REPORTE_169_PARAM_USUARIOLOG
        };

        session = request.getSession();
        save_PageltemsState(itemIds, request, session);

    } // :do_Reporte169_SaveState

    private void do_Reporte169_RemoveState(HttpServletRequest request) {

        HttpSession session;

        String[] itemIds = new String[]{
            REPORTE_169_PARAM_TURNO,
            REPORTE_169_PARAM_USUARIOLOG
        };

        session = request.getSession();

        session.removeAttribute(AWReportes.USUARIOS_CONSULTAR_POR_CIRCULO);
        session.removeAttribute("consultaUsuariosCajeros");

        delete_PageltemsState(itemIds, request, session);

    } // :do_Reporte169_RemoveState

```

**SIR-APP\src\gov\sir\core\web\navegacion\administracion\ControlNavegacionReportes.java**

**LINEA 506**  
//AHERRENO 01/06/2012  
//Reporte 169  
listaReportesJasper.add(AWReportes.REPORTE\_169\_SEND\_ACTION);

<b>LINEA 1339</b>	<pre>//AHERRENO 01/06/2012 //Reporte 169 if( AWReportes.REPORTE_169_BACK_ACTION.equals( accion ) ) {     return REPORTE_XX_BACK_EVENTRESP_OK; }</pre>
<b>SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDORreportesJasperDAO.java</b>	
<b>LINEA 553</b>	<pre>//AHERRENO 01/06/2012 //Reporte 169 //Se obtienen los parámetros para enviar al procedimiento listaParametros = new ArrayList(); listaParametros.add(AWReportes.REPORTE_169.PARAM_TURNO); mapaReportes.put(AWReportes.REPORTE_169.PARAM_CMDKEY, new Object[]{});     "SELECT seq_reporte_169.NEXTVAL FROM dual",     "sp_reporte_169(?, ?)", listaParametros,     "DELETE FROM sir_op_reporte_169 WHERE sec_proceso = "});</pre>
	<b>generalSIR</b>
	<b>co.com.iridium.generalSIR.comun.modelo.Transaccion.java</b>
<b>Clase nueva</b>	<b>co.com.iridium.generalSIR.dao.TransaccionSIRDALO.java</b>
<b>Clase nueva</b>	<b>co.com.iridium.generalSIR.transacciones.TransaccionSIR.java</b>
<b>Clase nueva</b>	<b>DESPLIEGUE – 2.7.5.1</b>
	<b>05/06/2012</b>
<b>066 Proceso_Correcciones_ajustar_cambio_nat_juridica_validación_fecha_documento_por_fecha_readicación_documento</b>	
Teniendo en cuenta que la Registradora de Ayapel, reporta incidencia, en la cual informa no poder realizar la corrección de un código de Naturaleza Jurídica de tres dígitos por uno de cuatro dígitos en el proceso de correcciones y después de verificar con la Mesa de Ayuda de ETB, donde se corrobora que la validación la realiza por la fecha del documento, se solicita que la validación de la fecha se realice con la de radicación del documento y no por la fecha del documento, tal como ocurre actualmente. El sistema debe permitir el cambio de códigos de Naturaleza Jurídica de tres dígitos por cuatro dígitos para aquellas anotaciones que tengan fecha de radicación a partir del 03 de julio de 2001, teniendo en cuenta la resolución 1695 del 31 de mayo de 2001.	
Para aquellos folios que tengan fecha de apertura superior a 03 de julio de 2001, el sistema debe permitir ingresar códigos de tres y cuatro dígitos.	
<b>DESARROLLO</b>	
	<b>Clase</b>
	<b>gov.sir.core.web.acciones.correccion.AWModificarFolio.java</b>
	<b>Método doFolioEdit_Anotacion_SaveChanges</b>
Líneas 611 – 6158	
Remplazar:	
<pre>Documento documentoAnotacion = t0_folio_anotacion.getDocumento(); if ( documentoAnotacion != null) {     String [] paramsFecha = CNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS.split("/");     Calendar cal = Calendar.getInstance();     if (paramsFecha.length == 3)     {         cal.set(Calendar.DAY_OF_MONTH, Integer.parseInt(paramsFecha[0]));         cal.set(Calendar.MONTH, Integer.parseInt(paramsFecha[1]) -1);         cal.set(Calendar.YEAR, Integer.parseInt(paramsFecha[2]));     } else</pre>	

```

{
    cal.set(Calendar.DAY_OF_MONTH, 15);
    cal.set(Calendar.MONTH, 07);
    cal.set(Calendar.YEAR, 2001);
}

Date fechaComparacion = cal.getTime();
boolean mostrarAntiguas = true;
boolean mostrarNuevas = true;
    if (documentoAnotacion.getFecha() != null)
    {
        NaturalezaJuridica nat = t0_folio_anotacion.getNaturalezaJuridica();
        if (documentoAnotacion.getFecha().before(fechaComparacion))
        {

            if (nat != null && idNaturaleza!=null &&
!idNaturaleza.equals(nat.getIdNaturalezaJuridica())
                && (idNaturaleza.length() == 4
                && !( nat.getIdNaturalezaJuridica().equals("915") ||
nat.getIdNaturalezaJuridica().equals("999"))
                    ))
            )
        {
            exception.addError("Naturaleza juridica no coincide con fecha de radicación
de documento: El código de la naturaleza Juridica debe ser de 3 dígitos");
        }
        } else if (nat != null && nat.getIdNaturalezaJuridica()!=null &&
idNaturaleza!=null && !idNaturaleza.equals(nat.getIdNaturalezaJuridica())&&
idNaturaleza.length() != 4)
        {
            exception.addError("Naturaleza juridica no coincide con fecha de radicación de
documento: El código de la naturaleza Juridica debe ser de 4 dígitos");
        }
    }
}

```

Por:

```

/**
 * @Autor: Edgar Lora
 * @Mantis: 0010688
 * @Requerimiento: 066_151
 * @Descripcion: Se cambio la fecha para la validacion de cambio de naturaleza juridica.
 */
Date fechaRadicacion = t0_folio_anotacion.getFechaRadicacion();
String [] paramsFecha = CNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS.split("/");
Calendar cal = Calendar.getInstance();
if (paramsFecha.length == 3)
{
    cal.set(Calendar.DAY_OF_MONTH, Integer.parseInt(paramsFecha[0]));
    cal.set(Calendar.MONTH, Integer.parseInt(paramsFecha[1]) -1);
    cal.set(Calendar.YEAR, Integer.parseInt(paramsFecha[2]));
}

```

```

} else
{
    cal.set(Calendar.DAY_OF_MONTH, 15);
    cal.set(Calendar.MONTH, 07);
    cal.set(Calendar.YEAR, 2001);
}

Date fechaComparacion = cal.getTime();
boolean mostrarAntiguas = true;
boolean mostrarNuevas = true;
if (fechaRadicacion != null){
    NaturalezaJuridica nat = t0_folio_anotacion.getNaturalezaJuridica();
    if (fechaRadicacion.before(fechaComparacion)){

        if (nat != null && idNaturaleza!=null &&
        !idNaturaleza.equals(nat.getIdNaturalezaJuridica())
            && (idNaturaleza.length() == 4
            && !( nat.getIdNaturalezaJuridica().equals("915") ||
        nat.getIdNaturalezaJuridica().equals("999")
            )))
    ){
        exception.addError("Naturaleza juridica no coincide con fecha de radicación de
documento: El código de la naturaleza Juridica debe ser de 3 dígitos");
    }
}
}

```

#### Clase

gov.sir.core.negocio.modelo.constantes.CNaturalezaJuridica.java

#### Variable FECHA\_NUEVAS\_NATURALEZAS

Remplazar:

public static final String FECHA\_NUEVAS\_NATURALEZAS = "15/08/2001";

Por:

public static final String FECHA\_NUEVAS\_NATURALEZAS = "03/07/2001";

#### JSP

#### WEB-

**INF/jspSeguro/corrección/CORRECC\_FOLIOEDIT\_ANOTACIONCANCELADORA.jsp**

#### Function juridica

Remplazar Línea – 262 a 264

```

/*var fechaRad = document.getElementById('<%=
AWModificarFolio.ANOTACION_FECHA_DOCUMENTO %>').value;
    var url = nombre + '&<%= AWModificarFolio.ANOTACION_FECHA_RADICACION
%>='+ fechaRad;*/
    popup=window.open(nombre,valor,dimensiones);

```

Por:

```
var fechaRad = null;
```

```
if(document.getElementById('<%=AWModificarFolio.ANOTACION_FECHA_DOCUMENTO%>') != null){  
    fechaRad =  
    document.getElementById('<%=AWModificarFolio.ANOTACION_FECHA_DOCUMENTO%>').  
    value;  
} else{  
    fechaRad =  
    document.getElementById('<%=AWModificarFolio.ANOTACION_FECHA_DOCUMENTO%>_h  
idden').value;  
}  
  
var url = nombre +  
'&<%=AWModificarFolio.ANOTACION_FECHA_RADICACION%>=' + fechaRad;  
popup=window.open(url,valor,dimensiones);
```

Agregar código nuevo Línea – 856

```
<input type="hidden" id="<%={AWModificarFolio.ANOTACION_FECHA_DOCUMENTO}%>_hidden" name="<%={AWModificarFolio.ANOTACION_FECHA_DOCUMENTO}%>_hidden" value="<%={session.getAttribute(AWModificarFolio.ANOTACION_FECHA_DOCUMENTO)}%>" />
```

Clase
gov.sir.core.web.helpers.comun. SeleccionarNaturalezaJuridicaHelper Metodo setProperties

Se modifco bloque de codigo – Línea 118 a 127:

```
fechaRadAnotacion =  
(Date)session.getAttribute(AWNaturalezaJuridica.ANOTACION_FECHA_RADICACION);
```

```

if (anotacion != null){
    Date fechaAnota = fechaRadAnotacion;
    NaturalezaJuridica nat = anotacion.getNaturalezaJuridica();
    // se verifica si la fecha de radicacion es anterior a FECHA_NUEVAS_NATURALEZAS
    // y si el código actual no es 915 o 999.
    if (fechaAnota != null &&
        fechaAnota.before(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS)){
        this.mostrarNuevas = false;
        this.mostrarAntiguas = true;
    } else {
        this.mostrarNuevas = true;
        this.mostrarAntiguas = false;
    }
    if (nat != null && (nat.getIdNaturalezaJuridica().equals("915") ||
        nat.getIdNaturalezaJuridica().equals("999"))){
        this.mostrarNuevas = true;
    }
}

```

Por

fechaRadAnotacion =

```

(Date)session.getAttribute(AWNaturalezaJuridica.ANOTACION_FECHA_RADICACION);
SimpleDateFormat formatoDelTexto = new SimpleDateFormat("dd/MM/yyyy");
String fechaAperturaFolio = request.getParameter(CDocumento.FECHA_RADICACION);
if(fechaAperturaFolio != null){
    try {
        fechaRadFolio = formatoDelTexto.parse(fechaAperturaFolio);
        session.setAttribute(CDocumento.FECHA_RADICACION, fechaRadFolio);
    } catch (ParseException ex) {

Logger.getLogger(SeleccionarNaturalezaJuridicaHelper.class.getName()).log(Level.SEVERE, null,
ex);
    }
}

if (anotacion != null){
    Date fechaAnota = fechaRadAnotacion;
    Date fechaFolio = (Date)session.getAttribute(CDocumento.FECHA_RADICACION);
    NaturalezaJuridica nat = anotacion.getNaturalezaJuridica();
    // se verifica si la fecha de radicacion es anterior a FECHA_NUEVAS_NATURALEZAS
    // y si el código actual no es 915 o 999.
    if (fechaAnota != null &&
fechaAnota.before(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS)){
        if(fechaFolio != null &&
fechaFolio.before(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS)){
            this.mostrarNuevas = true;
            this.mostrarAntiguas = true;
        }else{
            this.mostrarNuevas = true;
            this.mostrarAntiguas = false;
        }
    } else {
        this.mostrarNuevas = true;
        this.mostrarAntiguas = false;
    }
    if (nat != null && (nat.getIdNaturalezaJuridica().equals("915") ||
nat.getIdNaturalezaJuridica().equals("999"))){
        this.mostrarNuevas = true;
    }
}
}

```

### Clase

**gov.sir.core.web.acciones.correccion.AWModificarFolio.java**

**MetododoFolioEdit\_Anotacion\_SaveChanges**

```

/**
 * @Autor: Edgar Lora
 * @Mantis: 0010688
 * @Requerimiento: 066_151
 * @Descripcion: Se cambio la fecha para la validacion de cambio de naturaleza juridica.
 */
Date fechaRadicacion = t0_folio_anotacion.getFechaRadicacion();
String [] paramsFecha = CNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS.split("/");

```

```

Calendar cal = Calendar.getInstance();
if (paramsFecha.length == 3)
{
    cal.set(Calendar.DAY_OF_MONTH, Integer.parseInt(paramsFecha[0]));
    cal.set(Calendar.MONTH, Integer.parseInt(paramsFecha[1]) -1);
    cal.set(Calendar.YEAR, Integer.parseInt(paramsFecha[2]));
} else
{
    cal.set(Calendar.DAY_OF_MONTH, 15);
    cal.set(Calendar.MONTH, 07);
    cal.set(Calendar.YEAR, 2001);
}

Date fechaComparacion = cal.getTime();
booleanmostrarAntiguas = true;
booleanmostrarNuevas = true;
if (fechaRadicacion != null){
    NaturalezaJuridicanat = t0_folio_anotacion.getNaturalezaJuridica();
    if (fechaRadicacion.before(fechaComparacion)){

        if (nat != null&&idNaturaleza!=null&& !idNaturaleza.equals(nat.getIdNaturalezaJuridica()))
        && (idNaturaleza.length() == 4
        && !( nat.getIdNaturalezaJuridica().equals("915") || nat.getIdNaturalezaJuridica().equals("999"))
            ))
        ){
            exception.addError("Naturaleza juridica no coincide con fecha de radicación de documento: El
código de la naturaleza Juridica debe ser de 3 dígitos");
        }
    }
}

```

Línea 6123 a 6170 Cambiar codigo anterior Por:

```

/**
 * @Autor: Edgar Lora
 * @Mantis: 0010688
 * @Requerimiento: 066_151
 * @Descripcion: Se cambio la fecha para la validacion de cambio de naturaleza juridica.
 */
Date fechaRadicacion = t0_folio_anotacion.getFechaRadicacion();
Date fechaRadicacionFolio = null;
SimpleDateFormat formatoDelTexto = new SimpleDateFormat("dd/MM/yyyy");
String fechaAperturaFolio =
(String)session.getAttribute(CDocumento.FECHA_RADICACION);
if(fechaAperturaFolio != null){
    try {
        fechaRadicacionFolio = formatoDelTexto.parse(fechaAperturaFolio);
    } catch (ParseException ex) {
        Logger.getLogger(AWModificarFolio.class.getName()).log(Level.SEVERE, null, ex);
        fechaRadicacionFolio = null;
    }
}

```

```

}

String [] paramsFecha = CNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS.split("/");
Calendar cal = Calendar.getInstance();
if (paramsFecha.length == 3)
{
    cal.set(Calendar.DAY_OF_MONTH, Integer.parseInt(paramsFecha[0]));
    cal.set(Calendar.MONTH, Integer.parseInt(paramsFecha[1]) -1);
    cal.set(Calendar.YEAR, Integer.parseInt(paramsFecha[2]));
} else
{
    cal.set(Calendar.DAY_OF_MONTH, 15);
    cal.set(Calendar.MONTH, 07);
    cal.set(Calendar.YEAR, 2001);
}

Date fechaComparacion = cal.getTime();
boolean mostrarAntiguas = true;
boolean mostrarNuevas = true;
if (fechaRadicacion != null){
    NaturalezaJuridica nat = t0_folio_anotacion.getNaturalezaJuridica();
    if (fechaRadicacion.before(fechaComparacion)){
        if(fechaAperturaFolio != null && fechaRadicacionFolio.after(fechaComparacion)&&
!idNaturaleza.equals(nat.getIdNaturalezaJuridica()) && idNaturaleza.length() == 3){
            exception.addError("Naturaleza juridica no coincide con fecha de radicación: El código
de la naturaleza Juridica debe ser de 4 dígitos");
        }
    }else{
        if(fechaAperturaFolio != null && !idNaturaleza.equals(nat.getIdNaturalezaJuridica()) &&
idNaturaleza.length() == 3){
            exception.addError("Naturaleza juridica no coincide con fecha de radicación: El código
de la naturaleza Juridica debe ser de 4 dígitos");
        }
    }
}
}

```

Clase

gov.sir.core.web.acciones.correccion.AWModificarFolio.java

Método doFolioEdit\_Anotacion\_SaveChanges

Remplazar código de la línea 6119 a 6158:

```

Documento documentoAnotacion = t0_folio_anotacion.getDocumento();
if (documentoAnotacion != null)
{
    String [] paramsFecha =
CNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS.split("/");
    Calendar cal = Calendar.getInstance();
    if (paramsFecha.length == 3)
    {
        cal.set(Calendar.DAY_OF_MONTH, Integer.parseInt(paramsFecha[0]));

```

```

        cal.set(Calendar.MONTH, Integer.parseInt(paramsFecha[1]) -1);
        cal.set(Calendar.YEAR, Integer.parseInt(paramsFecha[2]));
    } else
    {
        cal.set(Calendar.DAY_OF_MONTH, 15);
        cal.set(Calendar.MONTH, 07);
        cal.set(Calendar.YEAR, 2001);
    }

    Date fechaComparacion = cal.getTime();
    boolean mostrarAntiguas = true;
    boolean mostrarNuevas = true;
    if (documentoAnotacion.getFecha() != null)
    {
        NaturalezaJuridica nat = t0_folio_anotacion.getNaturalezaJuridica();
        if (documentoAnotacion.getFecha().before(fechaComparacion))
        {

            if (nat != null && idNaturaleza!=null &&
!idNaturaleza.equals(nat.getIdNaturalezaJuridica()))
                && (idNaturaleza.length() == 4
                && !(

nat.getIdNaturalezaJuridica().equals("915") || nat.getIdNaturalezaJuridica().equals("999")
                    ))
            )
        {
            exception.addError("Naturaleza juridica no coincide con
fecha de radicación de documento: El código de la naturaleza Juridica debe ser de 3 dígitos");
        }
        } else if (nat != null && nat.getIdNaturalezaJuridica()!=null &&
idNaturaleza!=null && !idNaturaleza.equals(nat.getIdNaturalezaJuridica())&&
idNaturaleza.length() != 4)
        {
            exception.addError("Naturaleza juridica no coincide con fecha de
radicación de documento: El código de la naturaleza Juridica debe ser de 4 dígitos");
        }
    }
}

```

Por el siguiente codigo:

```

/**
 * @Autor: Edgar Lora
 * @Mantis: 0010688
 * @Requerimiento: 066_151
 * @Descripcion: Se cambio la fecha para la validacion de cambio de naturaleza juridica.
 */
Date fechaRadicacion = t0_folio_anotacion.getFechaRadicacion();
Date fechaRadicacionFolio = null;
SimpleDateFormat formatoDelTexto = new SimpleDateFormat("dd/MM/yyyy");

```

```

String fechaAperturaFolio =
(String)session.getAttribute(CDocumento.FECHA_RADICACION);
if(fechaAperturaFolio != null){
    try {
        fechaRadicacionFolio = formatoDelTexto.parse(fechaAperturaFolio);
    } catch (ParseException ex) {
        Logger.getLogger(AWModificarFolio.class.getName()).log(Level.SEVERE, null,
ex);
        fechaRadicacionFolio = null;
    }
}

String [] paramsFecha =
CNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS.split("/");
Calendar cal = Calendar.getInstance();
if (paramsFecha.length == 3)
{
    cal.set(Calendar.DAY_OF_MONTH, Integer.parseInt(paramsFecha[0]));
    cal.set(Calendar.MONTH, Integer.parseInt(paramsFecha[1]) -1);
    cal.set(Calendar.YEAR, Integer.parseInt(paramsFecha[2]));
} else
{
    cal.set(Calendar.DAY_OF_MONTH, 15);
    cal.set(Calendar.MONTH, 07);
    cal.set(Calendar.YEAR, 2001);
}

Date fechaComparacion = cal.getTime();
boolean mostrarAntiguas = true;
boolean mostrarNuevas = true;
if (fechaRadicacion != null){
    NaturalezaJuridica nat = t0_folio_anotacion.getNaturalezaJuridica();
    if(fechaAperturaFolio != null){
        if (fechaRadicacion.before(fechaComparacion)){
            if(fechaRadicacionFolio.before(fechaComparacion)&&
!idNaturaleza.equals(nat.getIdNaturalezaJuridica()) && idNaturaleza.length() == 4){
                exception.addError("Naturaleza juridica no coincide con fecha de radicación: El
código de la naturaleza Juridica debe ser de 3 dígitos");
            }
        }else{
            if(!idNaturaleza.equals(nat.getIdNaturalezaJuridica()) && idNaturaleza.length() ==
3){
                exception.addError("Naturaleza juridica no coincide con fecha de radicación: El
código de la naturaleza Juridica debe ser de 4 dígitos");
            }
        }
    }else{
        if (fechaRadicacion != null){
            if (fechaRadicacion.before(fechaComparacion)){

```

```

        if (nat != null && idNaturaleza!=null &&
    !idNaturaleza.equals(nat.getIdNaturalezaJuridica())
        && (idNaturaleza.length() == 4
        && !( nat.getIdNaturalezaJuridica().equals("915") ||
    nat.getIdNaturalezaJuridica().equals("999")
            ))
        ){
            exception.addError("Naturaleza juridica no coincide con fecha de
radicación de documento: El código de la naturaleza Juridica debe ser de 3 dígitos");
        }
    }
}
}
}

```

Clase

gov.sir.core.web.acciones.correccion.AWNaturalezaJuridica.java

Remplazar el valor de la variable:

public static final String FECHA\_NUEVAS\_NATURALEZAS\_STR = "15/08/2001";

Por:

public static final String FECHA\_NUEVAS\_NATURALEZAS\_STR = "03/07/2001";

Declarar variable nueva:

public static final String FOLIO\_FECHA\_APERTURA\_NJ =
" FOLIO\_FECHA\_APERTURA\_NJ";

Clase

gov.sir.core.web.helpers.comun.SeleccionarNaturalezaJuridicaHelper.java

Método setProperties

Este método se le hicieron bastantes cambios, por favor remplazar el código de éste por el siguiente:

```

public void setProperties(HttpServletRequest request)
throws HelperException {
HttpSession session = request.getSession();

List elementosGrupo = new Vector();
List elementosNat = new Vector();

List grupoNaturalezas = null;
Turno turno = (Turno)session.getAttribute(WebKeys.TURNO);
String cancelar = (String) session.getAttribute("cancelar");
String soloCancelar = (String) session.getAttribute("soloCancelar");

if (listaNaturalezaJuridicas == null || (turno.getIdProceso() ==
Long.parseLong(CProceso.PROCESO_CORRECCION)) && cancelar != null &&
cancelar.equals("true")) {
    grupoNaturalezas = (List)
}

```

```

session.getServletContext().getAttribute(WebKeys.LISTA_GRUPOS_NATURALEZAS_JURI
DICAS);
} else {
    grupoNaturalezas = listaNaturalezaJuridicas;
}

if (grupoNaturalezas == null) {
    grupoNaturalezas = new Vector();

    GrupoNaturalezaJuridica gNat;

    for (int i = 0; i < 5; i++)
    {
        gNat = new GrupoNaturalezaJuridica();
        gNat.setIdGrupoNatJuridica(i + "id");
        gNat.setNombre(i + "Nom");

        NaturalezaJuridica nat;

        for (int j = 0; j < 7; j++) {
            nat = new NaturalezaJuridica();
            nat.setIdNaturalezaJuridica("id" + i + j);
            nat.setNombre("Nom" + i + j);
            nat.setGrupoNaturalezaJuridica(gNat);
            gNat.addNaturalezaJuridica(nat);
        }
        grupoNaturalezas.add(gNat);
    }
}

if (cancelar == null)
{
    cancelar = "";
}

if (soloCancelar == null)
{
    soloCancelar = "";
}

anotacion =
(Anotacion)session.getAttribute(AWNaturalezaJuridica.ANOTACION_MODICADA);
// se está creando una nueva anotación
fechaRadAnotacion =
(Date)session.getAttribute(AWNaturalezaJuridica.ANOTACION_FECHA_RADICACION);
SimpleDateFormat formatoDelTexto = new SimpleDateFormat("dd/MM/yyyy");
String fechaAperturaFolio =
request.getParameter(AWNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ);
if(fechaAperturaFolio != null){
    try {

```

```

fechaRadFolio = formatoDelTexto.parse(fechaAperturaFolio);
session.setAttribute(AWNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ,
fechaRadFolio);
} catch (ParseException ex) {

Logger.getLogger(SeleccionarNaturalezaJuridicaHelper.class.getName()).log(Level.SEVERE,
null, ex);
}

if (anotacion != null){
    Date fechaAnota = fechaRadAnotacion;
    Date fechaFolio = null;
    NaturalezaJuridica nat = anotacion.getNaturalezaJuridica();

    if(turno.getIdProceso() == Long.parseLong(CProceso.PROCESO_CORRECCION)){
        fechaFolio =
(Date)session.getAttribute(AWNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ);

        if(fechaAnota != null &&
fechaAnota.before(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS) &&
fechaAnota.compareTo(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS) != 0){
            if(fechaFolio != null &&
fechaFolio.before(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS) &&
fechaFolio.compareTo(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS) != 0){
                this.mostrarNuevas = false;
                this.mostrarAntiguas = true;
            }else{
                this.mostrarNuevas = true;
                this.mostrarAntiguas = true;
            }
        }else{
            this.mostrarNuevas = true;
            this.mostrarAntiguas = false;
        }
    }else{
        // se verifica si la fecha de radicacion es anterior a
FECHA_NUEVAS_NATURALEZAS
        // y si el código actual no es 915 o 999.
        if (fechaAnota != null &&
fechaAnota.before(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS) &&
fechaAnota.compareTo(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS) != 0){
            this.mostrarNuevas = false;
            this.mostrarAntiguas = true;
        } else {
            this.mostrarNuevas = true;
            this.mostrarAntiguas = false;
        }
        if (nat != null && (nat.getIdNaturalezaJuridica().equals("915") ||
nat.getIdNaturalezaJuridica().equals("999"))){

```

```

        this.mostrarNuevas = true;
    }
}

} else if (fechaRadAnotacion != null &&
fechaRadAnotacion.before(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS) &&
fechaRadAnotacion.compareTo(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS)
!= 0){
    if(turno.getIdProceso() == Long.parseLong(CProceso.PROCESO_CORRECCION)){
        Date fechaFolio =
(Date)session.getAttribute(AWNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ);
        if(fechaFolio != null &&
fechaFolio.before(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS) &&
fechaFolio.compareTo(AWNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS) != 0){
            this.mostrarNuevas = false;
            this.mostrarAntiguas = true;
        }else{
            this.mostrarNuevas = true;
            this.mostrarAntiguas = true;
        }
    }else{
        this.mostrarNuevas = false;
        this.mostrarAntiguas = true;
    }
} else {
    this.mostrarNuevas = true;
    this.mostrarAntiguas = false;
}

if(turno==null || turno.getIdFase().equals(CFase.ANT_HOJA_RUTA)){
    this.mostrarNuevas = true;
    this.mostrarAntiguas = true;
}
if (!cancelar.equals("true"))
{
    Iterator itGrupos = grupoNaturalezas.iterator();
    idGrupoNat = request.getParameter(GRUPO_NATURALEZA_JURIDICA_ID);
    idNat = request.getParameter(NATURALEZA_JURIDICA_ID);

    while (itGrupos.hasNext()) {
        GrupoNaturalezaJuridica group = (GrupoNaturalezaJuridica) itGrupos.next();

        if (!this.mostrarCancelacion) {
            if
(!group.getIdGrupoNatJuridica().equalsIgnoreCase(CGrupoNaturalezaJuridica.ID_CANCELACION)) {
                elementosGrupo.add(new ElementoLista(
                    group.getIdGrupoNatJuridica(), group.getNombre()));
            }
        } else {

```

```

        if (!this.mostrarSoloCancelacion)
        {
            elementosGrupo.add(new ElementoLista(
                group.getIdGrupoNatJuridica(), group.getNombre()));
        }
        else
        {
            if
                (group.getIdGrupoNatJuridica().equalsIgnoreCase(CGrupoNaturalezaJuridica.ID_CANCELACION)) {
                    elementosGrupo.add(new ElementoLista(
                        group.getIdGrupoNatJuridica(), group.getNombre()));
                }
            }
        }

        if (idGrupoNat != null) {
            if (idGrupoNat.equals(group.getIdGrupoNatJuridica())) {
                nomGrupo = group.getNombre();
                session.setAttribute(GRUPO_NATURALEZA_JURIDICA_ID,
                    group.getIdGrupoNatJuridica());

                List naturalezas = group.getNaturalezaJuridicas();
                Iterator itNat = naturalezas.iterator();

                while (itNat.hasNext()) {
                    NaturalezaJuridica nat = (NaturalezaJuridica) itNat.next();
                    if ((this.mostrarAntiguas && nat.getIdNaturalezaJuridica().length() == 3)
                        || (this.mostrarNuevas && nat.getIdNaturalezaJuridica().length()
                        == 4))
                    {
                        elementosNat.add(new ElementoLista(
                            nat.getIdNaturalezaJuridica(),
                            nat.getNombre() + " - " +
                            nat.getIdNaturalezaJuridica()));
                    }

                    if (idNat != null) {
                        if (idNat.equals(nat.getIdNaturalezaJuridica())) {
                            session.setAttribute(NATURALEZA_JURIDICA_ID,
                                nat.getIdNaturalezaJuridica());
                            nomNaturaleza = nat.getNombre();
                            mostrarBoton = true;
                        }
                    }
                }
            }
        }
    } else {

```

```

Iterator itGrupos = grupoNaturalezas.iterator();
idGrupoNat = request.getParameter(GRUPO_NATURALEZA_JURIDICA_ID);
idNat = request.getParameter(NATURALEZA_JURIDICA_ID);

while (itGrupos.hasNext()) {
    GrupoNaturalezaJuridica group = (GrupoNaturalezaJuridica) itGrupos.next();

    if
(group.getIdGrupoNatJuridica().equals(CGrupoNaturalezaJuridica.ID_CANCELACION)) {
        elementosGrupo.add(new ElementoLista(
            group.getIdGrupoNatJuridica(), group.getNombre()));
    }

    if (idGrupoNat != null) {
        if (idGrupoNat.equals(group.getIdGrupoNatJuridica())) {
            nomGrupo = group.getNombre();
            session.setAttribute(GRUPO_NATURALEZA_JURIDICA_ID,
                group.getIdGrupoNatJuridica());

            List naturalezas = group.getNaturalezaJuridicas();
            Iterator itNat = naturalezas.iterator();

            while (itNat.hasNext()) {
                NaturalezaJuridica nat = (NaturalezaJuridica) itNat.next();
                String idnat = nat.getIdNaturalezaJuridica()
                    .substring(0, 1);

                if ((this.mostrarAntiguas && nat.getIdNaturalezaJuridica().length() == 3)
                    || (this.mostrarNuevas && nat.getIdNaturalezaJuridica().length()
== 4))
                {
                    elementosNat.add(new ElementoLista(
                        nat.getIdNaturalezaJuridica(),
                        nat.getNombre() + " - " +
                        nat.getIdNaturalezaJuridica()));
                }

                if (idNat != null) {
                    if (idNat.equals(nat.getIdNaturalezaJuridica())) {
                        session.setAttribute(NATURALEZA_JURIDICA_ID,
                            nat.getIdNaturalezaJuridica());
                        nomNaturaleza = nat.getNombre();
                        mostrarBoton = true;
                    }
                }
            }
        }
    }
}
}

```

```

gruposHelper = new ListaElementoHelper();
gruposHelper.setCssClase("campoformlista");
gruposHelper.setId(GRUPO_NATURALEZA_JURIDICA_ID);
gruposHelper.setNombre(GRUPO_NATURALEZA_JURIDICA_ID);
gruposHelper.setTipos(elementosGrupo);
gruposHelper.setFuncion("onChange=\"submitform();\"");
}

naturalezaHelper = new ListaElementoHelper();
naturalezaHelper.setCssClase("campoformlista");
naturalezaHelper.setId(NATURALEZA_JURIDICA_ID);
naturalezaHelper.setNombre(NATURALEZA_JURIDICA_ID);
naturalezaHelper.setTipos(elementosNat);
naturalezaHelper.setFuncion("onChange=\"submitform();\"");
}

```

### JSP

WEB-INF/jspSeguro/correcion/CORRECC\_FOLIOEDIT\_ANOTACIONCANCELADORA.jsp  
Funcion juridica

El codigo de esta funcion en este archivo debe remplazarse por el siguiente:

```

function juridica(nombre,valor,dimensiones)
{
    /**
     * Fecha: 18-08-2011
     * Autor: Guillermo Cabrera
     * Propósito: Al editar una anotación canceladora, no estaba cargando el popup para
seleccionar
     * la naturaleza juridica.
     * Se modifica la función ya que se estaba generando mal la url al adicionarle la fecha del
documento
     * que para este caso no se necesita.
     * Se setea en la sesión el atributo cancelar en true "session.setAttribute("cancelar",
"true")" para que solo
     * se carguen las naturalezas juridicas de cancelación.
     * Mantis: 0008593: Acta - Requerimiento No 029_151 - Error al agregar naturaleza al
folio.
    */
    <% session.setAttribute("cancelar", "true"); %>
    document.getElementById('natjuridica_id').value=valor+"_ID";
    document.getElementById('natjuridica_nom').value=valor+"_NOM";
    var fechaRad = null;

if(document.getElementById('<%=AWModificarFolio.ANOTACION_FECHA_RADICACION
%>') != null){
    fechaRad =
document.getElementById('<%=AWModificarFolio.ANOTACION_FECHA_RADICACION%
>').value;
} else{
    fechaRad =

```

```

document.getElementById('<%=AWModificarFolio.ANOTACION_FECHA_RADICACION%>_hidden').value;
}
var url = nombre +
'&<%=AWModificarFolio.ANOTACION_FECHA_RADICACION%>=' + fechaRad;
url = url + '&<%=WNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ %>=' +
document.getElementById('<%=CDocumento.FECHA_RADICACION %>').value;
popup=window.open(url,valor,dimensiones);
popup.focus();
}

```

Insertar codigo nuevo en la línea 857:

```

<input type="hidden" id="<%=_hidden" name="<%=_hidden"
value="<%=_hidden" value="<%=_hidden" />
    <input type="hidden" id="<%=_hidden" name="<%=_hidden"
value="<%=_hidden" value="<%=_hidden" />

```

JSP

WEB-INF/jspSeguro/correcion/EDITAR\_ANOTACION.jsp

Funcion juridica

El codigo de esta funcion en este archivo debe remplazarse por el siguiente:

```

function juridica(nombre,valor,dimensiones)
{
    document.getElementById('natjuridica_id').value=valor+"_ID";
    document.getElementById('natjuridica_nom').value=valor+"_NOM";
    var fechaRad = null;

if(document.getElementById('<%=AWModificarFolio.ANOTACION_FECHA_RADICACION%>') != null){
    fechaRad =
document.getElementById('<%=AWModificarFolio.ANOTACION_FECHA_RADICACION%>').value;
} else{
    fechaRad =
document.getElementById('<%=AWModificarFolio.ANOTACION_FECHA_RADICACION%>_hidden').value;
}
    var url = nombre +
'&<%=AWModificarFolio.ANOTACION_FECHA_RADICACION%>=' + fechaRad;
    url = url + '&<%=WNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ %>=' +
document.getElementById('<%=CDocumento.FECHA_RADICACION %>').value;
    popup=window.open(url,valor,dimensiones);
    popup.focus();
}

```

Insertar codigo nuevo en la línea 540:

```

<input type="hidden" id="<%=_hidden" name="<%=_hidden"
value="<%=_hidden" />
<input type="hidden" id="<%=_hidden" name="<%=_hidden"
value="<%=_hidden" />

```

JSP

WEB-INF/jspSeguro/correcion/AGREGAR\_ANOTACION.jsp

Funcion juridica

El codigo de esta funcion en este archivo debe remplazarse por el siguiente:

```

function juridica(nombre,valor,dimensiones)
{
    document.getElementById('natjuridica_id').value=valor+_ID";
    document.getElementById('natjuridica_nom').value=valor+_NOM";
    var fechaRad = null;

if(document.getElementById('<%=_hidden' != null){
    fechaRad =
document.getElementById('<%=_hidden').value;
} else{
    fechaRad =
document.getElementById('<%=_hidden').value;
}
    var url = nombre +
'&<%=_hidden'='+ fechaRad;
    url = url + '&<%=_hidden'='+
document.getElementById('<%=_hidden').value;
    popup=window.open(url,valor,dimensiones);
    popup.focus();
}

```

Insertar codigo nuevo en la linea 220:

```

<input type="hidden" id="<%=_hidden" name="<%=_hidden"
value="<%=_hidden" />
<input type="hidden" id="<%=_hidden" name="<%=_hidden"
value="<%=_hidden" />

```

JSP

WEB-INF/jspSeguro/correcion/CORREC\_CREAR\_ANOTACIONCANCELACION.jsp

### Funcion juridica

El codigo de esta funcion en este archivo debe remplazarse por el siguiente:

```
function juridica(nombre,valor,dimensiones)
{
    <%session.setAttribute("cancelar", "true"); %>
    document.getElementById('natjuridica_id').value=valor+"_ID";
    document.getElementById('natjuridica_nom').value=valor+"_NOM";
    var fechaRad = null;

    if(document.getElementById('<%=AWModificarFolio.ANOTACION_FECHA_RADICACION%>') != null){
        fechaRad =
        document.getElementById('<%=AWModificarFolio.ANOTACION_FECHA_RADICACION%>').value;
    }else{
        fechaRad =
        document.getElementById('<%=AWModificarFolio.ANOTACION_FECHA_RADICACION%>_hidden').value;
    }
    var url = nombre +
    '&<%=AWModificarFolio.ANOTACION_FECHA_RADICACION%>=' + fechaRad;
    url = url + '&<%=AWNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ %>=' +
    document.getElementById('<%=CDocumento.FECHA_RADICACION %>').value;
    popup=window.open(url,valor,dimensiones);
    popup.focus();
}
```

Insertar codigo nuevo en la linea 404:

```
<input type="hidden" id="<%=_AWModificarFolio.ANOTACION_FECHA_RADICACION%>_hidden" name="<%=_AWModificarFolio.ANOTACION_FECHA_RADICACION%>_hidden"
value="<%=_session.getAttribute(AWModificarFolio.ANOTACION_FECHA_RADICACION)%> />
<input type="hidden" id="<%=_CDocumento.FECHA_RADICACION%>" name="<%=_CDocumento.FECHA_RADICACION%>" value="<%=_session.getAttribute(CDocumento.FECHA_RADICACION)%>"/>
```

### Clase

gov.sir.core.web.acciones.correccion.AWNaturalezaJuridica.java

Cambiar bloque de codigo, linea 29 a 53:

```
static {
    String []fechaSplit = FECHA_NUEVAS_NATURALEZAS_STR.split("/");
    if (fechaSplit.length == 3)
    {
        if (fechaSplit[0].length() == 2
            && fechaSplit[1].length() == 2
            && fechaSplit[2].length() == 4)
    {
```

```

        Calendar cal = Calendar.getInstance();
        int dia = Integer.parseInt(fechaSplit[0]);
        int mes = Integer.parseInt(fechaSplit[1]);
        int anio = Integer.parseInt(fechaSplit[2]);

        cal.set(Calendar.DAY_OF_MONTH, dia);
        cal.set(Calendar.MONTH, mes-1);
        cal.set(Calendar.YEAR, anio);

        cal.set(Calendar.HOUR_OF_DAY, 0);
        cal.set(Calendar.MINUTE, 0);
        cal.set(Calendar.MILLISECOND, 0);

        FECHA_NUEVAS_NATURALEZAS = cal.getTime();

    }
}
}

```

Por:

```

static {

    SimpleDateFormat formatoDelTexto = new SimpleDateFormat("dd/MM/yyyy");
    try {
        FECHA_NUEVAS_NATURALEZAS =
formatoDelTexto.parse(FECHA_NUEVAS_NATURALEZAS_STR);
    } catch (ParseException ex) {
        Logger.getLogger(AWNaturalezaJuridica.class.getName()).log(Level.SEVERE, null,
ex);
    }
}

```

### JSP

#### WEB-INF/jspSeguro/comun/NATURALEZA\_JURIDICA.jsp

Cambiar bloque de codigo, línea 68 a 90

```

String []fechaSplit = fechaRadStr.split("/");
    if (fechaSplit.length == 3)
    {
        if ((fechaSplit[0].length() == 2 || fechaSplit[0].length() == 1)
            && (fechaSplit[1].length() == 2 ||
fechaSplit[1].length() == 1)
            && fechaSplit[2].length() == 4)
        {
            Calendar cal = Calendar.getInstance();
            int dia = Integer.parseInt(fechaSplit[0]);
            int mes = Integer.parseInt(fechaSplit[1]);
            int anio = Integer.parseInt(fechaSplit[2]);

            cal.set(Calendar.DAY_OF_MONTH, dia);
            cal.set(Calendar.MONTH, mes-1);
            cal.set(Calendar.YEAR, anio);
        }
    }
}

```

```

        cal.set(Calendar.HOUR_OF_DAY, 0);
        cal.set(Calendar.MINUTE, 0);
        cal.set(Calendar.MILLISECOND, 0);

        fechaRad = cal.getTime();
    }
}

```

Por:

```

SimpleDateFormat formatoDelTexto = new SimpleDateFormat("dd/MM/yyyy");
fechaRad = formatoDelTexto.parse(fechaRadStr);

```

Clase
gov.sir.core.web.acciones.correccion.AWModificarFolio.java
Metodo doFolioEdit_Anotacion_SaveChanges

Cambiar bloque de codigo, línea 6142 a 6153:

```

String [] paramsFecha = CNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS.split("/");
Calendar cal = Calendar.getInstance();
if (paramsFecha.length == 3)
{
    cal.set(Calendar.DAY_OF_MONTH, Integer.parseInt(paramsFecha[0]));
    cal.set(Calendar.MONTH, Integer.parseInt(paramsFecha[1]) -1);
    cal.set(Calendar.YEAR, Integer.parseInt(paramsFecha[2]));
} else
{
    cal.set(Calendar.DAY_OF_MONTH, 15);
    cal.set(Calendar.MONTH, 07);
    cal.set(Calendar.YEAR, 2001);
}

```

Por:

```

Date fechaComparacion = null;
try {
    fechaComparacion =
formatoDelTexto.parse(CNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS);
} catch (ParseException ex) {
    Logger.getLogger(AWModificarFolio.class.getName()).log(Level.SEVERE, null, ex);
}

```

Borrar línea 6155 a 6156:

```

Date fechaComparacion = cal.getTime();

```

Cambiar bloque de codigo, línea 6159 a 6185:

```

if (fechaRadicacion != null){
    NaturalezaJuridica nat = t0_folio_anotacion.getNaturalezaJuridica();
    if(fechaAperturaFolio != null){

```

```

if (fechaRadicacion.before(fechaComparacion)){
    if(fechaRadicacionFolio.before(fechaComparacion)&&
!idNaturaleza.equals(nat.getIdNaturalezaJuridica()) && idNaturaleza.length() == 4){
        exception.addError("Naturaleza jurídica no coincide con fecha de radicación: El
código de la naturaleza Jurídica debe ser de 3 dígitos");
    }
}else{
    if(!idNaturaleza.equals(nat.getIdNaturalezaJuridica()) && idNaturaleza.length() ==
3){
        exception.addError("Naturaleza jurídica no coincide con fecha de radicación: El
código de la naturaleza Jurídica debe ser de 4 dígitos");
    }
}
}else{
    if (fechaRadicacion != null){
        if (fechaRadicacion.before(fechaComparacion)){

            if (nat != null && idNaturaleza!=null &&
!idNaturaleza.equals(nat.getIdNaturalezaJuridica())
                && (idNaturaleza.length() == 4
                && !( nat.getIdNaturalezaJuridica().equals("915") ||
nat.getIdNaturalezaJuridica().equals("999")
                    )))

            ){
                exception.addError("Naturaleza jurídica no coincide con fecha de
radicación de documento: El código de la naturaleza Jurídica debe ser de 3 dígitos");
            }
        }
    }
}
}

```

Por:

```

if (fechaRadicacion != null){
    NaturalezaJuridica nat = t0_folio_anotacion.getNaturalezaJuridica();
    if(local_Turno == null || local_Turno.getIdProceso() ==
Long.parseLong(CProceso.PROCESO_CORRECCION)){
        if (fechaRadicacion.before(fechaComparacion) &&
fechaRadicacion.compareTo(fechaComparacion) != 0){
            if(fechaRadicacionFolio.before(fechaComparacion) &&
fechaRadicacionFolio.compareTo(fechaComparacion) != 0 &&
!idNaturaleza.equals(nat.getIdNaturalezaJuridica()) && idNaturaleza.length() == 4){
                exception.addError("Naturaleza jurídica no coincide con fecha de radicación: El
código de la naturaleza Jurídica debe ser de 3 dígitos");
            }
        }else{
            if(!idNaturaleza.equals(nat.getIdNaturalezaJuridica()) && idNaturaleza.length() ==
3){
                exception.addError("Naturaleza jurídica no coincide con fecha de radicación: El
código de la naturaleza Jurídica debe ser de 4 dígitos");
            }
        }
    }
}

```

```

código de la naturaleza Juridica debe ser de 4 dígitos");
        }
    }
}else{
    if (fechaRadicacion != null){
        if (fechaRadicacion.before(fechaComparacion)){
            if (nat != null && idNaturaleza!=null &&
!idNaturaleza.equals(nat.getIdNaturalezaJuridica())
                && (idNaturaleza.length() == 4
                && !( nat.getIdNaturalezaJuridica().equals("915") ||
nat.getIdNaturalezaJuridica().equals("999")))){
                    exception.addError("Naturaleza juridica no coincide con fecha de
radicación de documento: El código de la naturaleza Juridica debe ser de 3 dígitos");
                }
            }
        }
    }
}

```

#### Metodo agregarAnotacion

Cambiar bloque de codigo, línea: 10374

Calendar fechaRadicacion = darFecha(valFechaRadicacion);

Por:

```

Date fechaRadicacion = null;
try {
    fechaRadicacion = toDate(valFechaRadicacion);
} catch (ParseException ex) {
    Logger.getLogger(AWModificarFolio.class.getName()).log(Level.SEVERE, null, ex);
}

```

Cambiar bloque de codigo, línea: 10395

Calendar fechaDocumento = darFecha(valFechaDocumento);

Por:

```

Date fechaDocumento = null;
try {
    fechaDocumento = toDate(valFechaDocumento);
} catch (ParseException ex) {
    Logger.getLogger(AWModificarFolio.class.getName()).log(Level.SEVERE, null, ex);
}

```

Cambiar bloque de codigo, línea: 10413 a 10415

```

if( fechaRadicacion.before( fechaDocumento ) ) {
    exception.addError( "fecha radicacion no puede ser menor a fecha de documento");
}

```

Por:

```
if( fechaRadicacion.before( fechaDocumento ) &&
fechaRadicacion.compareTo(fechaDocumento) != 0) {
    exception.addError( "fecha radicacion no puede ser menor a fecha de documento");
}
```

Cambiar bloque de código, línea 10488 a 10523:

```
if ( idNaturaleza != null)
{
    String [] paramsFecha =
CNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS.split("/");
    Calendar cal = Calendar.getInstance();
    if (paramsFecha.length == 3)
    {
        cal.set(Calendar.DAY_OF_MONTH,
Integer.parseInt(paramsFecha[0]));
        cal.set(Calendar.MONTH, Integer.parseInt(paramsFecha[1]) -1);
        cal.set(Calendar.YEAR, Integer.parseInt(paramsFecha[2]));
    } else
    {
        cal.set(Calendar.DAY_OF_MONTH, 15);
        cal.set(Calendar.MONTH, 07);
        cal.set(Calendar.YEAR, 2001);
    }

    Date fechaComparacion = cal.getTime();
    boolean mostrarAntiguas = true;
    boolean mostrarNuevas = true;
    if (fechaDocumento != null)
    {
        if (fechaDocumento.getTime().before(fechaComparacion))
        {

            if (idNaturaleza != null && !idNaturaleza.equals("")
                && idNaturaleza.length() == 4 )
            {
                exception.addError("Naturaleza jurídica no
coincide con fecha de radicación de documento: El código de la naturaleza Jurídica debe ser de
3 dígitos");
            }
        } else if (idNaturaleza != null && !idNaturaleza.equals("")
                    && idNaturaleza.length() != 4)
        {
            exception.addError("Naturaleza jurídica no coincide con
fecha de radicación de documento: El código de la naturaleza Jurídica debe ser de 4 dígitos");
        }
    }
}
```

```
}
```

Por:

```
Turno local_Turno;
    local_Turno = (Turno)session.getAttribute( WebKeys.TURNO );
    Date fechaRadicacionFolio = null;
        String fechaAperturaFolio =
(String)session.getAttribute(CDocumento.FECHA_RADICACION);
    SimpleDateFormat formatoDelTexto = new SimpleDateFormat("dd/MM/yyyy");
if(fechaAperturaFolio != null){
    try {
        fechaRadicacionFolio = formatoDelTexto.parse(fechaAperturaFolio);
    } catch (ParseException ex) {
        Logger.getLogger(AWModificarFolio.class.getName()).log(Level.SEVERE,
null, ex);
        fechaRadicacionFolio = null;
    }
}

if ( idNaturaleza != null)
{
    Date fechaComparacion = null;
try {
    fechaComparacion =
formatoDelTexto.parse(CNaturalezaJuridica.FECHA_NUEVAS_NATURALEZAS);
} catch (ParseException ex) {
    Logger.getLogger(AWModificarFolio.class.getName()).log(Level.SEVERE,
null, ex);
}
boolean mostrarAntiguas = true;
boolean mostrarNuevas = true;
if(local_Turno == null || local_Turno.getIdProceso() ==
Long.parseLong(CProceso.PROCESO_CORRECCION)){
    if(fechaRadicacion != null){
        if (fechaRadicacion.before(fechaComparacion) &&
fechaRadicacion.compareTo(fechaComparacion) != 0){
            if(fechaRadicacionFolio.before(fechaComparacion) &&
fechaRadicacionFolio.compareTo(fechaComparacion) != 0 && idNaturaleza.length() == 4){
                exception.addError("Naturaleza juridica no coincide con fecha de
radicación: El código de la naturaleza Juridica debe ser de 3 dígitos");
            }
        }else{
            if(idNaturaleza.length() == 3){
                exception.addError("Naturaleza juridica no coincide con fecha de
radicación: El código de la naturaleza Juridica debe ser de 4 dígitos");
            }
        }
    }else{
}
```

```

        if (fechaRadicacion != null){
            if (fechaRadicacion.before(fechaComparacion)){
                if (idNaturaleza != null && !idNaturaleza.equals("") &&
idNaturaleza.length() == 4 ){
                    exception.addError("Naturaleza juridica no coincide con fecha de
radicación de documento: El código de la naturaleza Juridica debe ser de 3 dígitos");
                }
            } else if (idNaturaleza != null && !idNaturaleza.equals("") &&
idNaturaleza.length() != 4){
                exception.addError("Naturaleza juridica no coincide con fecha de
radicación de documento: El código de la naturaleza Juridica debe ser de 4 dígitos");
            }
        }
    }
}

```

Borrar bloque de codigo, línea 10529 a 10530:

```

Turno local_Turno;
local_Turno = (Turno)session.getAttribute( WebKeys.TURNO );

```

Cambiar bloque de codigo, línea 10633:

```
anotacion.setFechaRadicacion(fechaRadicacion.getTime());
```

Por:

```
anotacion.setFechaRadicacion(fechaRadicacion);
```

Cambiar bloque de codigo, línea 10655:

```
documento.setFecha(fechaDocumento.getTime());
```

Por:

```
documento.setFecha(fechaDocumento);
```

#### Metodo nuevo toDate

```

/**
 * Método que convierte una cadena de texto en un objeto tipo Date
 * @param fechaInterfaz
 * @return
 */
private Date toDate(String fechaInterfaz) throws ParseException {
if((null == fechaInterfaz) || ("").equals(fechaInterfaz.trim())))
{
    return null;
}
SimpleDateFormat formatoDelTexto = new SimpleDateFormat("dd/MM/yyyy");
return formatoDelTexto.parse(fechaInterfaz);
}

```

#### Clase

gov.sir.core.web.helpers.comun.SeleccionarNaturalezaJuridicaHelper.java Metodo setProperties
Cambiar bloque de codigo, línea 163:
if(turno.getIdProceso() == Long.parseLong(CProceso.PROCESO_CORRECCION)){
Por:
if(turno == null    turno.getIdProceso() == Long.parseLong(CProceso.PROCESO_CORRECCION)){
JSP
WEB-INF/jspSeguro/correcion/CORRECC_FOLIOEDIT_ANOTACIONCANCELADORA.jsp
Cambiar línea 269:
url = url + '&<%=AWNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ %>=' + document.getElementById('<%=CDocumento.FECHA_RADICACION %>').value;
Por:
url = url + '&<%=AWNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ %>=' + document.getElementById('<%=AWModificarFolio.FOLIO_FECHA_APERTURA %>').value;
Cambiar línea 858:
<input type="hidden" id="<%="CDocumento.FECHA_RADICACION %>" name="<%="CDocumento.FECHA_RADICACION %>" value="<%="session.getAttribute(CDocumento.FECHA_RADICACION) %>"/>
Por:
<input type="hidden" id="<%="AWModificarFolio.FOLIO_FECHA_APERTURA %>" name="<%="AWModificarFolio.FOLIO_FECHA_APERTURA %>" value="<%="session.getAttribute(AWModificarFolio.FOLIO_FECHA_APERTURA) %>"/>
JSP
WEB-INF/jspSeguro/correcion/EDITAR_ANOTACION.jsp
Cambiar línea 371:
url = url + '&<%=AWNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ %>=' + document.getElementById('<%=CDocumento.FECHA_RADICACION %>').value;
Por:
url = url + '&<%=AWNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ %>=' + document.getElementById('<%=AWModificarFolio.FOLIO_FECHA_APERTURA %>').value;
Cambiar línea 541:
<input type="hidden" id="<%="CDocumento.FECHA_RADICACION %>" name="<%="CDocumento.FECHA_RADICACION %>" value="<%="session.getAttribute(CDocumento.FECHA_RADICACION) %>"/>

Por:

```
<input type="hidden" id="<%=_AWModificarFolio.FOLIO_FECHA_APERTURA %>"  
name="<%=_AWModificarFolio.FOLIO_FECHA_APERTURA %>"  
value="<%=_session.getAttribute(_AWModificarFolio.FOLIO_FECHA_APERTURA) %>"/>  
JSP
```

WEB-INF/jspSeguro/correcion/AGREGAR\_ANOTACION.jsp

Cambiar línea 167:

```
url = url + '&<%=_WNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ %>=' +  
document.getElementById('<%=_CDocumento.FECHA_RADICACION %>').value;
```

Por:

```
url = url + '&<%=_WNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ %>=' +  
document.getElementById('<%=_AWModificarFolio.FOLIO_FECHA_APERTURA %>').value;
```

Cambiar línea 223:

```
<input type="hidden" id="<%=_CDocumento.FECHA_RADICACION %>"  
name="<%=_CDocumento.FECHA_RADICACION %>"  
value="<%=_session.getAttribute(CDocumento.FECHA_RADICACION) %>"/>
```

Por:

```
<input type="hidden" id="<%=_AWModificarFolio.FOLIO_FECHA_APERTURA %>"  
name="<%=_AWModificarFolio.FOLIO_FECHA_APERTURA %>"  
value="<%=_session.getAttribute(_AWModificarFolio.FOLIO_FECHA_APERTURA) %>"/>
```

JSP

WEB-INF/jspSeguro/correcion/CORREC\_CREAR\_ANOTACIONCANCELACION.jsp

Cambiar línea 122:

```
url = url + '&<%=_WNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ %>=' +  
document.getElementById('<%=_CDocumento.FECHA_RADICACION %>').value;
```

Por:

```
url = url + '&<%=_WNaturalezaJuridica.FOLIO_FECHA_APERTURA_NJ %>=' +  
document.getElementById('<%=_AWModificarFolio.FOLIO_FECHA_APERTURA %>').value;
```

Cambiar línea 405:

```
<input type="hidden" id="<%=_CDocumento.FECHA_RADICACION %>"  
name="<%=_CDocumento.FECHA_RADICACION %>"  
value="<%=_session.getAttribute(CDocumento.FECHA_RADICACION) %>"/>
```

Por:

```
<input type="hidden" id="<%=_AWModificarFolio.FOLIO_FECHA_APERTURA %>"  
name="<%=_AWModificarFolio.FOLIO_FECHA_APERTURA %>"  
value="<%=_session.getAttribute(_AWModificarFolio.FOLIO_FECHA_APERTURA) %>"/>
```

Clase
gov.sir.core.web.acciones.correccion.AWModificarFolio.java
Cambiar línea 6132:
String fechaAperturaFolio = (String)session.getAttribute(CDocumento.FECHA_RADICACION);
Por:
String fechaAperturaFolio = (String)session.getAttribute(AWModificarFolio.FOLIO_FECHA_APERTURA);
Cambiar línea 10489:
String fechaAperturaFolio = (String)session.getAttribute(CDocumento.FECHA_RADICACION);
Por:
String fechaAperturaFolio = (String)session.getAttribute(AWModificarFolio.FOLIO_FECHA_APERTURA);
<b>DESPLIEGUE – 2.7.5</b> <b>30/03/2012</b>
<b>002 CERTIFICADOS REIMPRESOS EN CORRECCIONES (INFORMATIVO)</b>
Certificados reimpressos en correcciones. SIR-275 En la radicación de la solicitud de corrección no hay ningún campo que permita vincular el o los certificados que deben ser reimpressos una vez efectuada la corrección.  Es necesario que se creen controles desde la radicación de la solicitud que permitan controlar la reimpresión de certificados en el proceso de correcciones, con la posibilidad de que el responsable de correcciones en la fase de revisión y análisis pueda vincular el certificado asociado en caso de haberse omitido en la radicación de la solicitud y que al momento de reimprimirse el certificado se pueda modificar la matrícula inmobiliaria contenida en el turno de certificado anexo en aquellos casos en que la corrección requiera abrir una nueva matrícula.
SEGUIMIENTO ADA: Especificar requerimiento, indicando si es reimprimir un turno de certificado existente o crear un nuevo turno de certificado asociado al turno de corrección.
OBSERVACIONES SNR: No considero que el aplicativo deba permitir la reimpresión de folios nuevos (pudo haberse generado una devolución, si hubo omisión en la calificación), el aplicativo tiene controles para permitir la impresión de certificados asociados, no obstante es recomendable la solicitud.  Se aclara esta observación: En folio magnético se encuentra la funcionalidad en correcciones que si se adjunta un certificado de tradición original a la solicitud de correcciones que haya sido expedido dentro del término de meses, al corregir la anotación o campo del folio, se le reimprime el certificado

con la información ya corregida con el turno con que fue expedido el certificado.

Actualmente en SIR está permitiendo reimprimir un certificado con la información corregida sin solicitar un turno lo imprime con el turno de corrección, lo cual no es correcto, debe efectuarse el ajuste conforme esta la solicitud del SIR 275.

#### **Analisis/Desarrollo**

Este requerimiento busca evitar la impresión repetida de certificados de tradición y libertad, con posterioridad a una corrección del folio; incluyendo la verificación del pago. Este caso fue entregado por el anterior operador SIN RESOLVER. En el análisis que se ha realizado, se han realizado varias especificaciones a la solicitud. La última especificación se hizo en febrero 9 de 2011, en conversación con la Doctora Yamile Moreno. Este requerimiento implica modificación al modelo de datos, ya que se debe construir como mínimo dos tablas más que manejen esta relación (que no existe en el modelo actual), y se complemente la tarea en el código de aplicación con la validación del pago. Este tema de verificación de pagos, corresponde a una de las tareas globales de mejora de aplicación, que se han expuesto a SNR; dado que el modelo actual NO fue diseñado para atender la individualidad de la relación pago-producto.

**Solución:** Crear dos tablas nuevas que permitan romper la relación cíclica entre matriculas padre e hijo, en el proceso de creación automática. La otra tabla debe guardar información que relacione el turno con la liquidación y pagos de la matricula generadora.

**Impacto:** En base de datos se crean como mínimo dos tablas nuevas, que pueden ser pobladas con información histórica; o simplemente dar inicio al poblado con el despliegue de aplicación correspondiente. Estos objetos generan nuevas secuencias, índices y relaciones de integridad con el modelo actual. En Aplicación, se modifican las clases que intervienen en el módulo de correcciones y la interface debe acomodarse en lógica de negocio y funcional a este cambio.

**Costo:** Aproximado 4 meses, afectación funcional aproximado del 40% de aplicación. Ya se está trabajando en el control por sesión del número de impresiones.

### **005 ERROR EN LAS FECHAS DE CERTIFICADO DE FOLIO (Informativo)**

Turno:2010-300-1-22194

Matricula(s):300-195018

Motivo Inconsistencia:

José Luis García nos informa que al generar el certificado del folio 300-195018 dicho certificado sale con las fechas incorrectas en las anotaciones 1, 2, 3 y 4.

anotación 1 29/7/1992 y 9/7/1992 cuando en realidad son 30/7/1992 y 10/7/1992

anotación 2 22/9/1992 y 6/9/1992 cuando en realidad son 23/9/1992 y 7/9/1992

anotación 3 22/12/1992 y 15/12/1992 cuando en realidad son 23/12/1992 y 16/12/1992

anotación 4 22/12/1992 y 15/12/1992 cuando en realidad son 23/12/1992 y 16/12/1992

Grupo Bucaramanga

#### **Analisis/Desarrollo**

Este requerimiento busca solucionar un problema que se presenta en la impresión de certificados en impresoras de red. Este error se relaciona con el diseño general del modelo de impresión SIR, que utiliza componente cliente de java en las maquinas locales. Este problema, así como el de las impresiones incompletas de folios extensos, se está evaluando en el marco de un nuevo diseño de impresión orientado a Colas de Mensajería. Hasta el momento NO se tiene una solución a este

problema.

**Solución:** Individualizar el error, que se sabe actualmente que esta generado por la forma en que se distribuye la impresión en red. En este caso se ha establecido que un conjunto amplio de problemas de impresión (certificados que se imprimen parcialmente, firmas que salen con poca estética, necesidad de reimprimir, folios muy extensos, devoluciones de turnos con alto volumen de matrículas asociadas, y otras), pueden ser corregidos haciendo uso de nuevas herramientas de desarrollo (framework) y modelamiento (arquitectura java para JMS); lo cual implica la redefinición, análisis de reglas de negocio, y aplicación de nuevos diseños dentro del esquema J2EE.

**Impacto:** El impacto es de aplicación y compromete el modelo de impresión de certificados.

**Costo:** Se estima el cambio en no menos de 6 meses con una afectación del 40% de los objetos de aplicación.

Después de probar en los ambientes de producción 192.168.202.22:7778/SNR y calidad 192.168.202.9:8100/SNR con el círculo pruebas no se replicó el inconveniente presentado al usuario de Bucaramanga.

**Observaciones:**

La mesa de ayuda hará seguimiento al caso, se da como cerrado este requerimiento ya que no se replicó en las pruebas.

**Despliegue 002\_6\_2**

En las pruebas realizadas para este despliegue se replicó el error, Se radico turno de certificado inmediato en el ambiente de producción del folio 999-240 y se observa que en la anotación número dos en el imprimible se refleja que la fecha de radicación es 27-07-1992, al consultar el folio por la opción ver detalles de folio, para la anotación dos se observa como fecha real 28/07/1992, la prueba se realizó con una impresora de red. Se realizó la misma prueba con una impresora PDF y no se observa la inconsistencia. Por favor tener en cuenta en el análisis.

## 011 ERROR EN IMPRESION DE FORMULARIO DE CALIFICACION

Requerimiento Glpi No 25457

Turno: 2011-300-6-264

Matricula(s): 300-280521, 300-343878 hasta 300-344145.

Motivo Inconsistencia: No es posible obtener la impresión del formulario de calificación.

Se comunica el funcionario José Luis García e informa que se está intentando reimprimir el formulario de calificación del turno 2011-300-6-264 pero no es posible obtener dicha impresión. Favor generar la impresión de dicho formulario de calificación en pdf y adjuntarlo para enviarlo a la oficina y pueda ser impreso.

Grupo Bucaramanga

**Analisis/Solución**

Se realizan pruebas en el ambiente de Local apuntando a la base de datos SURIC, con 300 folios segregados, obteniendo resultados satisfactorios, de igual manera se realizaron pruebas en Producción con en circulo de PRUEBAS con 300 folios segregados y también se obtuvieron resultados satisfactorios, se realizaron pruebas en el ambiente de Calidad y no se pudo imprimir el Formulario de calificación, luego se inicio el servicio de mi localhost con el mismo sir.ear de Calidad y allí se

obtuvieron resultados satisfactorios. se le asigna el caso a la Ing. Angelica Herreño para pruebas en el ambiente de Calidad.

1.Las pruebas en ambiente de calidad no fueron exitosas. Pruebas:

Servidor: <http://192.168.202.9:8100/SNR> [^]

Turno: 2011-300-6-35955

Usuario: HERCILIA.MORALES

Rol: Calificador

2.Se realizan pruebas en el ambiente configurado por el ingeniero Stanlee Santana con Sistema Gestión Documental y no fueron exitosas. Pruebas:

Servidor: <http://192.168.150.102:7778/SNR> [^]

Turno: 2011-300-6-35955

Usuario: HERCILIA.MORALES

Rol: Calificador

3.Se realizan pruebas en el ambiente sin el Sistema Gestión Documental configurado por el ingeniero Stanlee Santana y fueron exitosas. Pruebas:

Servidor: <http://192.168.150.102:7778/SNR> [^]

Turno: 2011-300-6-35955

Usuario: HERCILIA.MORALES

Rol: Calificador

#### Análisis

Después de las pruebas realizadas por la ingeniera Angélica Herreño, se procede a realizar pruebas con una copia generada del ambiente de calidad el día 20 de diciembre 2010.

1.Se realizan pruebas en el ambiente con el Sistema Gestión Documental configurado por el ingeniero Stanlee Santana, las cuales fueron exitosas.

Servidor: <http://192.168.150.102:8100/SNR> [^]

Turno: 2011-300-6-35955

Usuario: HERCILIA.MORALES

Rol: Calificador

2.Se procede a configurar el ambiente que fue exitoso para que pueda trabajar en el servidor de calidad.

Servidor: <http://192.168.202.9:8100/SNR> [^]

Turno: 2011-300-6-35955

Usuario: HERCILIA.MORALES

Rol: Calificador

Las pruebas en dicho servidor fueron exitosas. se determina inconsistencias en los archivos de ambiente que venia corriendo con anterioridad.

#### Análisis

Después de realizar las pruebas por el Ing. Javier Fajardo del día 13/02/2012 en el ambiente de calidad para el aplicativo SIR con los siguientes parámetros iniciales:

Turno: 2012-300-6-4894

Folio Inicial: 300-1546

Numero Segregaciones: 305 folios

Servidor: 192.168.202.9

Puerto: 8100

usuario: LADYS.PALACIO

Se determino que el aplicativo genero la impresión del formulario de calificación temporal y el formulario de calificación definitivo.

Se procede a limpiar el contenido de las carpetas tmp y work localizadas en la siguiente ruta:  
/jboss/jboss-4.0.4.GA/server/default/

Se reinicia el aplicativo y se realizan las pruebas pertinentes por los Ingenieros Javier Fajardo y Angélica Herreño.

Las pruebas en dicho servidor fueron exitosas. Se determina inconsistencias en los archivos temporales del ambiente de calidad.

### 013 REPORTE 91\_INFORME DE INGRESOS DEL IMPUESTO DE REGISTRO

#### Requerimiento Glpi No 26485

Motivo Inconsistencia: funcionaria informa que el "Reporte 91: Informe de ingresos del impuesto de registro", no trae los valores correctos con respecto al "Reporte 51: Resumen diario de ingresos y egresos". Se verifican los reportes enviados por fax de la Orip y se observa que el total de ingresos por registro del reporte "Reporte 91: Informe de ingresos del impuesto de registro" no corresponde al que se observa en el reporte 51 "Reporte 51: Resumen diario de ingresos y egresos".

Se generan reportes en Mesa de ayuda y se observan los reportes con los datos correctos, favor verificar inconsistencia presentada con los datos que genera el "Reporte 91: Informe de ingresos del impuesto de registro" al finalizar el día.

Grupo Aguachica

#### **Analisis/Solución**

Se realizan pruebas pero no se puede replicar el error reportado por el usuario.

Igualmente se realiza análisis al código SP\_REPORT\_91.sql y se encontró una condición que no se tenía en el script versionado. Se opta por suprimir esta condición.

### 014 REPORTE 09 DIARIO RADICADOR DE DOCUMENTOS

#### **REQUERIMIENTO: REPORTE 09 DIARIO RADICADOR DE DOCUMENTOS DEL DIA**

Mediante oficio de marzo 16 de 2011 la Orip de Pereira informa que en Auditoria realizada por la Contraloría General de la Nación, se hicieron observaciones orientadas a mejorar la calidad de los reportes del Aplicativo SIR , entre estas la siguiente "Que en el reporte 09 Diario Radicador de Documentos del Día, en los números de turnos falta el año, círculo y número del proceso"

Por lo expuesto es necesario que la información solicitada se refleje en el reporte mencionado.

#### **Analisis/Desarrollo**

Se modifica el reporte en la columna turno para que muestre el valor completo.

Archivos Adjuntos.  
[SP\_Reporte\_09.sql]  
[reporte-09.jrxml]  
[reporte-09.jasper]

## 018 CORRECCION OFICINA DE ORIGEN INTERNACIONAL

**CORRECCION OFICINA DE ORIGEN INTERNACIONAL:** Se requiere que en el proceso de correcciones al seleccionar el permiso “ANOTACIONES-DOCUMENTO, en el Rol AUXILIAR DE CORRECCIONES en la fase de ejecutar Corrección, en las anotaciones contenidas en los folios de matrícula inmobiliaria y asociadas a un turno de corrección, el aplicativo debe habilitar la opción para modificar, corregir y/o agregar la información del campo de oficina de origen internacional.

### Analisis/Desarrollo

Se crearon jsp nuevos en SIR:

- WEB-INF.jspSeguro.correccion.HELPER\_OFICINAS.jsp.

Se modificaron los siguientes archivos en SIR:

- WEB-INF.jspSeguro.correccion.AGREGAR\_ANOTACION.jsp.
- WEB-INF.jspSeguro.correccion.EDITAR\_ANOTACION.jsp.
- sir.core.web.acciones.correccion.AWModificarFolio.java.
- sir.core.web.helpers.correccion.region.model.FolioAnotacionEditBindManager.java.
- sir.forseti.dao.impl.jdogenie.JDOGenieFolioDAO.java.

#### ARCHIVO JSP:

WEB-INF.jspSeguro.correccion.AGREGAR\_ANOTACION.jsp.

#### Código Modificado Línea 420 - 423:

```
<!-- EL HELPER DE OFICINA EMPIEZA ACA -->
<jsp:include page="../correccion/HELPER_OFICINAS.jsp" flush="true" />
<!-- EL HELPER DE OFICINA TERMINA ACA -->
<%--
```

#### Código Modificado Línea 430:

```
--%>
```

#### ARCHIVO JSP:

WEB-INF.jspSeguro.correccion.EDITAR\_ANOTACION.jsp.

#### Código Modificado Línea 1233 – 1236:

```
<!-- EL HELPER DE OFICINA EMPIEZA ACA -->
<jsp:include page="../correccion/HELPER_OFICINAS.jsp" flush="true" />
<!-- EL HELPER DE OFICINA TERMINA ACA -->
<!--<tr>
```

#### Código Modificado Línea 1239:

```
<%--
```

#### Código Modificado Línea 1247:

```
--%>
```

#### Código Modificado Línea 1251:

```
</tr>-->
```

#### ARCHIVO JAVA:

sir.core.web.acciones.correccion.AWModificarFolio.java.

#### Código Inserado Línea 531 - 540.

```
/*
 * @author : Ellery David Robles Gómez.
 * @casoMantis : 08375.
 * @actaRequerimiento : 261 - Corrección Oficina de Origen Internacional.
 * @change : Se agrega la variable "ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL"
 * para identificarla en la sesión.
 */
/** Esta constante identifica el campo donde se introduce la oficina internacional del documento en la anotación*/
public static final String ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL =
"ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL";
```

#### Código Inserido Línea 5832 - 5839.

```
/*
 * @author : Ellery David Robles Gómez.
 * @casoMantis : 08375.
```

```

* @actaRequerimiento : 261 - Correccion Oficina de Origen Internacional.
* @change      : Se agrega la variable "oficinaInternacional"
*                  para capturar su valor de la respuesta.
*/
String oficinaInternacional = request.getParameter(AWModificarFolio.ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL);

```

#### Código Inserado Linea 6200 - 6207.

```

/**
 * @author      : Ellery David Robles Gómez.
 * @casoMantis   : 08375.
 * @actaRequerimiento : 261 - Correccion Oficina de Origen Internacional.
 * @change      : Se setea el valor de "oficinaInternacional"
 *                  a bindManager array que contiene los campos de la anotacion:documento.
 */
bindManager.getBinderById( "field:folio:anotacion[i]:documento/oficinaInternacional" ).setValue( oficinaInternacional );

```

#### Código Inserado Linea 6238.

```
, bindManager.getBinderById( "field:folio:anotacion[i]:documento/oficinaInternacional" ).getValue()
```

#### Código Modificado Linea 6387 - 6388.

```
//validatorManagerItem_fieldId = "field:folio:anotacion[i]:documento/oficinaOrigen/nombre";
validatorManagerItem_fieldId = "field:folio:anotacion[i]:documento/oficinaInternacional";
```

#### Código Inserado Linea 7344 - 7346.

```
if(t2_folio_anotacion.getDocumento() == null){
    t2_folio_anotacion.setDocumento(t0_folio_anotacion.getDocumento());
}
```

#### Código Inserado Linea 7376 - 7385.

```
if(request.getParameter(WebKeys.TIPO_OFICINA_I_N).equals(WebKeys.TIPO_OFICINA_INTERNACIONAL)){
    Documento documento = (Documento)t2_folio_anotacion.getDocumento();

    documento.setOficinaInternacional(request.getParameter(AWModificarFolio.ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL));
    documento.setOficinaOrigen(null);
    t2_folio_anotacion.setDocumento(documento);
}
```

```
if(request.getParameter(WebKeys.TIPO_OFICINA_I_N).equals(WebKeys.TIPO_OFICINA_NACIONAL)){
    Documento documento = (Documento)t2_folio_anotacion.getDocumento();
    documento.setOficinaInternacional(null);
```

#### Código Modificado Linea 7398 - 7403.

```
documento.setOficinaOrigen(oficinaTemp);
t2_folio_anotacion.setDocumento(documento);
}
}catch (Exception e){
    exception.addError(e.getMessage());
}
```

#### Código Modificado Linea 10375 - 10377.

```
// if ((valorDepartamento.length() <= 0)) {
//     exception.addError("Debe seleccionar un departamento válido para la oficina de procedencia del documento en la
anotación");
// }
```

#### Código Modificado Linea 10394 - 10396.

```
// if ((valorMunicipio.length() <= 0)) {
//     exception.addError("Debe seleccionar un municipio válido para la oficina de procedencia del documento en la
anotación");
// }
```

#### Código Modificado Linea 10405 - 10407.

```
// if ((valorVereda.length() <= 0)) {
//     exception.addError("Debe seleccionar una vereda válida para la oficina de procedencia del documento en la
anotación");
// }
```

#### Código Insertado Linea 10412.

```
String oficinaInternacional = request.getParameter(AWModificarFolio.ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL);
```

#### Código Insertado Linea 10414 - 10426.

```
if(request.getParameter(WebKeys.TIPO_OFICINA_I_N).equals(WebKeys.TIPO_OFICINA_NACIONAL)){
    if ((valorDepartamento.length() <= 0)) {
        exception.addError("Debe seleccionar un departamento válido para la oficina de procedencia del documento en la anotación");
    }

    if ((valorMunicipio.length() <= 0)) {
        exception.addError("Debe seleccionar un municipio válido para la oficina de procedencia del documento en la
anotación");
    }
}
```

```

        if ((valorVereda.length() <= 0)) {
            exception.addError("Debe seleccionar una vereda válida para la oficina de procedencia del documento en la anotación");
        }
    }

```

**Código Insertado Linea 10433 - 10437.**

```

}else{
    if ((oficinaInternacional == null) || (oficinaInternacional.length() <= 0)) {
        exception.addError("Debe indicar una oficina internacional para la procedencia del documento en la anotación");
    }
}

```

**Código Insertado Linea 10631.**

```
if(request.getParameter(WebKeys.TIPO_OFICINA_I_N).equals(WebKeys.TIPO_OFICINA_NACIONAL)){
```

**Código Modificado Linea 10654.**

```
// documento.setTipoDocumento(tipoDoc);
```

**Código Insertado Linea 10656 - 10658.**

```

}else{
    documento.setOficinaInternacional(oficinaInternacional);
}

```

**Código Insertado Linea 10660.**

```
documento.setTipoDocumento(tipoDoc);
```

**Código Insertado Linea 12124 - 12126.**

```

if (request.getParameter(AWModificarFolio.ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL) != null) {
    session.setAttribute(AWModificarFolio.ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL,
request.getParameter(AWModificarFolio.ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL));
}

```

**Código Insertado Linea 12286 - 12288.**

```

if (folio.getDocumento().getOficinaInternacional() != null) {
    String oficinaInternacional = folio.getDocumento().getOficinaInternacional();
    session.setAttribute(AWModificarFolio.ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL,
oficinaInternacional);
}

```

**Código Insertado Linea 12290.**

```
}
```

**Código Insertado Linea 12588 - 12595.**

```

if (request.getParameter(WebKeys.TIPO_OFICINA_I_N) != null) {
    session.setAttribute(WebKeys.TIPO_OFICINA_I_N, request.getParameter(WebKeys.TIPO_OFICINA_I_N));
}

if (request.getParameter(AWModificarFolio.ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL) != null) {
    session.setAttribute(AWModificarFolio.ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL,
request.getParameter(AWModificarFolio.ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL));
}

```

**Código Insertado Linea 12849 - 12852.**

```

if (documento.getOficinaInternacional() != null) {
    session.setAttribute(AWModificarFolio.ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL,
documento.getOficinaInternacional());
}

```

**Código Insertado Linea 13164 - 13165.**

```
session.removeAttribute(WebKeys.TIPO_OFICINA_I_N);
session.removeAttribute(AWModificarFolio.ANOTACION_OFICINA_DOCUMENTO_OFICINA_INTERNACIONAL);
```

**Código Insertado Linea 14666.**

```
, ProcessManager.StaticEvaluator.evaluate(localProcessManager, "field:folio:anotacion[i]:documento/oficinaInternacional" )
```

**Código Insertado Linea 14704.**

```
, new BasicStringComparator(true,true)
```

**Código Insertado Linea 14735.**

```
, bindManager.getBinderById("field:folio:anotacion[i]:documento/oficinaInternacional" ).getPath()
```

**ARCHIVO JAVA:**

```
sir.core.web.helpers.correccion.region.modelFolioAnotacionEditBindManager.java
```

**Código Inserado Linea 29.**

```
, new Binder("field:folio:anotacion[i]:documento/oficinaInternacional" , "documento/oficinaInternacional" )
```

**ARCHIVO JAVA:**

```
sir.forseti.dao.impl.jdogenie.JDOGenieFolioDAO.java.
```

**Código Modificado Linea 13123.**

```
if(anotacion.getDocumento().getOficinaOrigen() != null){
```

**Código Insertado Linea 13137.**

```
}
```

**Código Insertado Linea 13652.**

```
if ((datos.getOficinaInternacional() == null) || (datos.getOficinaInternacional().length() <= 0)) {
```

**Código Modificado Linea 13654 - 13660.**

```
throw new DAOException("El documento debe tener oficina origen");
```

```

}else{
    OficinaOrigenEnhancedPk ofid = new OficinaOrigenEnhancedPk();
    ofid.idOficinaOrigen = datos.getOficinaOrigen().getIdOficinaOrigen();

    OficinaOrigenEnhanced oficina = zonaRegistralDAO.getOficinaOrigen(ofid,pm);
    datos.setOficinaOrigen(oficina);
}

```

**Código Insertado Linea 13661.**

```

}
Código Modificado Linea 13669 - 13680.
// OficinaOrigenEnhancedPk ofid = new OficinaOrigenEnhancedPk();
// ofid.idOficinaOrigen = datos.getOficinaOrigen()
// .getIdOficinaOrigen();
//
// OficinaOrigenEnhanced oficina = zonaRegistralDAO.getOficinaOrigen(ofid,
// pm);
//
// if (oficina == null) {
//     throw new DAOException(
//         "La oficina origen no existe. IdOficina: " +
//         ofid.idOficinaOrigen);
// }

```

**Código Modificado Linea 13694.**

```
// datos.setOficinaOrigen(oficina);
```

**Código Insertado Linea 29210 - 29211.**

```
// toUpdate.setOficinaInternacional(null);
// toUpdate.setOficinaInternacional(fromDoc.getOficinaInternacional());
```

**Código Insertado Linea 29214 - 29222.**

```

if(fromDoc.getCirculo()!= null){
    toUpdate.setCirculo(fromDoc.getCirculo());
}

if (fromDoc.getOficinaInternacional() != null) {
    toUpdate.setOficinaInternacional(fromDoc.getOficinaInternacional());
    toUpdate.setOficinaOrigen(null);
}

```

## 020 REQUERIMIENTO OPCION PANT ADM SOLICITUD DE CORRECCION DE CALIFICACION

### PANTALLA ADMINISTRATIVA OPCION “SOLICITUD DE CORRECCION DE CALIFICACION”

En atención al oficio 2602011EE00944 de fecha 8 de abril de 2011, suscrito por la Doctora Zoraida Arce Cartagena , Registradora Principal de Instrumentos Públicos de Cúcuta, en el cual hace la siguiente solicitud: “Me permito comunicarle que en el Sistema de Información Registral SIR, los turnos de registro que están en la fase de Firma de Registro ó en la fase de Mesa de Control, el sistema SIR permite que estos turnos puedan ser retornados por los calificadores a través de sus pantallas administrativas del SIR para realizar una corrección de calificación, estos turnos quedan trasladados y desaparecen de la fase de mesa de control o firma de registro sin necesidad de ser solicitados al despacho para la devolución del turno al calificador. Ejemplo de ello, se puede observar en el turno 2010-260-6-18543, turno que estando en la fase de mesa de control fue retorna dos veces por el calificador ó el turno 2011-260-6-352 que estando en la fase de Mesa de Control y luego en Firma y Sello fue retorna por el calificador a través de su pantallas administrativas del SIR para la fase de aprobar corrección de calificación.

Considero que el sistema SIR debe ofrecer seguridad en los procesos y no debería presentarse esta situación permisiva, ya que cuando los turnos los refleje el sistema en la fase de Firma de Registro ó Mesa de Control, éstos deben ser solicitados al despacho directamente por el calificador que requiere la corrección y trasladados únicamente por parte del responsable de la estación y no ser trasladados por los calificadores a través del SIR sin previo protocolo de permiso, ya que lo correcto es que el despacho, en el caso específico de Firma de Registro, devuelva el turno por pantallas administrativas e incluya en la nota informativa el motivo de la solicitud”

Por lo anterior y con el fin de establecer control y seguridad en la devolución de turnos de registro para corrección de calificación la SNR solicita que la opción de “Solicitud de Corrección de Calificación” del Bloque “Proceso Registro de documentos” Pantallas Administrativas se inhabilite y no permita realizar devolución de turnos para corrección de calificación a los usuarios calificadores, esta funcionalidad debe realizarse únicamente por el botón “CALIFICACION” que se encuentra en la fase de firma del Rol Registrador, Coordinador Jurídico y Mesa de Control.

#### **Analisis/Desarrollo**

Script\_base de datos:

```
delete from sir_op_rol_pantalla where id_pantalla_administrativa = 105
delete from sir_op_pantalla_administrativa where id_pantalla_administrativa = 105
```

#### **021 ERROR EN ANOTACIONES - CIUDADANOS**

##### **Requerimiento Glpi No 27812**

Motivo Inconsistencia: Se recibe correo con el siguiente reporte: "En el proceso de calificación se está presentando la siguientes falla en el sir: cuando se adicionan las personas intervinientes, y se desea eliminar alguna de las personas agregas para su correspondiente arreglo, el sistema ELIMINA TODAS LAS PERSONAS QUE SE HAN AGREGADO, cuando solo debería eliminar la que se le dijo".

Se realizan pruebas en el aplicativo SIR, se crea una anotación y se agregan varios ciudadanos pero al eliminar uno de ellos se eliminan automáticamente todos los demás.

Favor verificar porque el sistema elimina todo el listado de ciudadanos agregados al eliminar uno de ellos, se anexan imágenes.

Grupo Bucaramanga

#### **Analisis/Desarrollo**

➤ Creación De Clases:

- NINGUNA (el mismo formato de modificación clases o jsp)

Se modificaron los siguientes archivos:

CLASE
SIR-APP\src\gov\sir\core\web\acciones\registro\AWCalificacion.java
Método:
private Evento eliminarCiudadano(HttpServletRequest request) throws AccionWebException

Se adicionan las siguientes líneas al método.

```
/*
 * Fecha: 16-08-2011
 * Autor: Guillermo Cabrera
 * Se valida que no existan ciudadanos en el formulario para poder removerlos de la sesión
 * ya que se estaban removiendo todos los ciudadanos cuando se eliminaba solo uno desde el formulario.
 * Mantis 8411 0008411: Acta - Requerimiento No 021_151 - Error en anotaciones - ciudadanos
 */
/request.getSession().setAttribute(WebKeys.LISTA_CIUDADANOS_ANOTACION, ciudadanos);
if (ciudadanos==null || ciudadanos.isEmpty())
{
    request.getSession().removeAttribute(WebKeys.LISTA_CIUDADANOS_ANOTACION);
}
else
{
    request.getSession().setAttribute(WebKeys.LISTA_CIUDADANOS_ANOTACION, ciudadanos);
}
```

#### **ARCHIVO JAVA:**

SIR-APP\src\gov\sir\forseti\dao\impl\jdogenie\JDOGenieFolioDAO.java
---

<b>Código Modificado Línea 13125 - 13132:</b>
---

```

    /**
     * @author: Ellery David Robles Gómez.
     * @change: Condición que verifica si se encuentra lleno el documento de la anotacion.
     * Mantis : 08411.
     * Acta : Requerimiento No 021_151 - Error en anotaciones - ciudadanos.
     */
    if(anotacion.getDocumento() != null){
        if(anotacion.getDocumento().getOficinaOrigen() != null){
Código insertado Linea 13146 - 13147:
    }
}

```

## 024 ERROR EN ANOTACIÓN DE MEDIDA CAUTELAR

### Requerimiento GLPI No.27950

Turno:2011-290-6-9622

Motivo Inconsistencia:

Se comunica la usuaria Mónica Robledo e informa que con el turno 2011-290-6-9622 se está calificando los folios:

290-28999, 290-48168, 290-48177, 290-48178, 290-51739, 290-51747, 290-80609, 290-96561, 290-103926, 290-117956, 290-117984, 290-117986.

Se está creando la anotación de medida cautelar con naturaleza jurídica EMBARGO CON JURISDICCION COACTIVA en cada uno de los folios pero al ingresar por la opción de gravámenes no me aparece en un folio y en otro si como es el caso del folio 290-48168 que si aparece con la anotación 18 que aún está en temporal y en el folio 290-103926 no aparece y está creada con la anotación 16 que también está en temporal.

Tal como lo muestran las imágenes anexas, se requiere verificar este inconveniente.

Grupo Pereira.

### Analisis/Desarrollo

Al revisar el caso se encuentra bug de la aplicación SIR al momento de mostrar las medidas cautelares del folio en la pantalla de calificación. Se certifica que los datos se encuentran temporales y al hacerlos definitivos estos se podrán observar desde esta pantalla.

Para la resolución de este caso se debe verificar:

1. Que la consulta que se realiza para mostrar estos datos contenga lo que se requiere. Si este es el problema se debe modificar la consulta.
2. Si el problema no es la consulta se tendrá que revisar al momento de mostrar en pantalla el por qué no se está haciendo y corregirlo.

- sir.forseti.interfaz. ForsetiServiceInterface.java
- sir.core.negocio.acciones.registro. ANCcalificacion.java
- sir.forseti.dao.impl.jdogenie. JDOGenieFolioDAO.java
- sir.forseti. ForsetiService.java
- sir.forseti.ejb. ForsetiServiceEJBBean.java
- sir.forseti.dao. FolioDAO.java

### ARCHIVO JAVA:

sir.forseti.interfaz. ForsetiServiceInterface.java

### Código Insertado Linea 1504 - 1513:

```

    /**
     * @author : Ellery David Robles G.
     * @casoMantis : 08469.
     * @actaReq : 024_151 - Error en anotación de medida cautelar.
     * @change : Retorna las anotaciones definitivas y temporales de un folio segun el parametro naturalezajurudica.
     * @param   : oid.

```

```

* @param   : naturalezaJuridica.
* @throws  : Throwable.
*/
public List getAnotacionesFolioNJ(FolioPk oid, String naturalezaJuridica) throws Throwable;

```

**ARCHIVO JAVA:**

sir.core.negocio.acciones.registro.ANCalificacion.java

**Código Modificado Línea 2100:**

```
//long numeroMedidasCautelares = forseti.getCountAnotacionesFolio(id,
CCriterio.POR_GRUPO_NAT_JURIDICA,CGrupoNaturalezaJuridica.MEDIDA_CAUTELAR);
```

**Código Modificado Línea 2110:**

```
List medidasCautelares = forseti.getAnotacionesFolioNJ(id, CGrupoNaturalezaJuridica.MEDIDA_CAUTELAR);
```

**ARCHIVO JAVA:**

sir.forseti.dao.impl.jdogenie.JDOGenieFolioDAO.java

**Código Inserado Línea 1440 - 1536.**

```

* @author : Ellery David Robles G.
* @casoMantis : 08469.
* @actaReq : 024_151 - Error en anotación de medida cautelar.
* @change : Retorna las anotaciones definitivas y temporales de un folio segun el parametro naturaleza juridica.
* @param   : oid.
* @param   : naturalezaJuridica.
* @throws  : DAOException.
*/
@Override
public List getAnotacionesFolioNJ(FolioPk oid, String naturalezaJuridica) throws DAOException {
    FolioEnhanced fol = null;
    AnotacionEnhanced anotacionEnh = null;
    PersistenceManager pm = AdministradorPM.getPM();
    List listaAnotaciones = new ArrayList();
    List listaTMPAnotaciones = new ArrayList();

    try {
        fol = this.getFolioByID(new FolioEnhancedPk(oid), pm);
        if (fol == null) {
            throw new DAOException("El folio no existe. IdMatricula: " + oid.idMatricula);
        }
        List rta2 = this.getAnotacionesRealesFolio(fol, pm);
        //List rta2 = fol.getAnotaciones();
        for (Iterator itr = rta2.iterator(); itr.hasNext();) {
            anotacionEnh = (AnotacionEnhanced) itr.next();
            this.makeTransientAnotacion(anotacionEnh, pm, new ArrayList());
            listaTMPAnotaciones.add(anotacionEnh);
        }
        for (Iterator itr = fol.getAnotacionesTmps().iterator();
             itr.hasNext(); ) {
            this.makeTransientAnotacionTMP((AnotacionTMP) itr.next(),
                pm, new ArrayList());
        }
        pm.makeTransient(fol);
        // Se agregan las anotaciones temporales y definitivas a la lista
        // En caso de que sea una actualización de una anotación definitiva
        // se modifica
        fol.setAnotaciones(listaTMPAnotaciones);
        List anotacionesTMP = fol.getAnotacionesTmps();
        Folio rta = (Folio) fol.toTransferObject();
        if(anotacionesTMP != null){
            AnotacionTMP anota;
            List cache;
            Anotacion anotaAux;

            for (Iterator it = anotacionesTMP.iterator(); it.hasNext(); ) {
                anota = (AnotacionTMP) it.next();
                cache = new ArrayList();
                anotaAux = (Anotacion) anota.getDefinitivo(cache).toTransferObject();

                int indice;
                Anotacion anToUpdate;
                indice = rta.getAnotaciones().indexOf(anotaAux);
                if (indice != -1) {
                    //La anotación temporal representa una actualización a la anotación definitiva
                    anToUpdate = (Anotacion) rta.getAnotaciones().get(indice);
                    anToUpdate.setTemporal(true);

                    if (anotaAux.getNaturalezaJuridica() != null) {

```

```

        anToUpdate.setNaturalezaJuridica(anotaAux.getNaturalezaJuridica());
    }
} else{
    anotaAux.setTemporal(true);
    rta.addAnotacione(anotaAux);
}
}

//Pasar las anotaciones a una lista
for(Iterator iter = rta.getAnotaciones().iterator() ; iter.hasNext();){
    Anotacion anotacion = (Anotacion)iter.next();

    if(anotacion.getNaturalezaJuridica().getGrupoNaturalezaJuridica().getIdGrupoNatJuridica().substring(0,2).equals(naturalezaJuridica.substring(0,2))){
        listaAnotaciones.add(AnotacionEnhanced.enhance(anotacion));
    }
}
} catch (JDOObjectNotFoundException e) {
    Log.getInstance().error(JDOGenieFolioDAO.class, e.getMessage());
    throw new DAOException(e.getMessage(), e);
} catch (JDOException e) {
    Log.getInstance().error(JDOGenieFolioDAO.class, e.getMessage());
    throw new DAOException(e.getMessage(), e);
} finally {
    pm.close();
}
listaAnotaciones = TransferUtils.makeTransferAll(listaAnotaciones);
return listaAnotaciones;
}
/**

```

#### ARCHIVO JAVA:

sir.forseti.ForsetiService.java

#### Código Inserado Línea 2815 - 2835.

```

* @author : Ellery David Robles G.
* @casoMantis : 08469.
* @actaReq : 024_151 - Error en anotación de medida cautelar.
* @change : Retorna las anotaciones definitivas y temporales de un folio segun el parametro naturaleza juridica.
* @param   : oid.
* @param   : naturalezaJuridica.
* @throws  : ForsetiException.
*/
@Override
public List getAnotacionesFolioNJ(FolioPk oid, String naturalezaJuridica) throws ForsetiException {
    try {
        return forseti.getFolioDAO().getAnotacionesFolioNJ(oid, naturalezaJuridica);
    }
    catch (DAOException e) {
        ForsetiException fe = new ForsetiException("Fallo en el servicio de obtención de anotaciones", e);
        Log.getInstance().error(ForsetiService.class, fe.getMessage(), e);
        throw fe;
    }
}
/**

```

#### ARCHIVO JAVA:

sir.forseti.ejb.ForsetiServiceEJBBean.java

#### Código Insertado Línea 918 - 923.

```

* @see gov.sir.forseti.interfaz.ForsetiServiceInterface#getAnotacionesFolio(gov.sir.core.negocio.modelo.Folio.FolioPk, java.lang.String)
* @ejb:interface-method view-type="both"
*/
public List getAnotacionesFolioNJ(FolioPk oid, String naturalezaJuridica) throws ForsetiException {
    return forseti.getAnotacionesFolioNJ(oid, naturalezaJuridica);
}
/**
```

#### ARCHIVO JAVA:

sir.forseti.dao.FolioDAO.java

#### Código Insertado Línea 1191 - 1200.

```

/**
* @author : Ellery David Robles G.
* @casoMantis : 08469.
* @actaReq : 024_151 - Error en anotación de medida cautelar.
* @change : Retorna las anotaciones definitivas y temporales de un folio segun el parametro naturaleza juridica.
* @param   : oid.
* @param   : naturalezaJuridica.
* @throws  : DAOException.
*/

```

```
 */
public List getAnotacionesFolioNJ(FolioPk oid, String naturalezaJuridica) throws DAOException;
```

## 025 ERROR AL DESASOCiar EL FOLIO

### Requerimiento GLPI No.28075

Turno: 2011-260-6-13688

Matricula(s): 260-121216, 260-196058.

Motivo Inconsistencia: El sistema no permite desasociar la matrícula 260-121216 en la fase CONFRONTACION\_CORRECTIVA Estación Asignada: X-CONFRONTADOR.260.

La oficina envía E-mail informando error que notifica el sistema al momento de desasociar la matrícula 260-121216 del turno de registro 2011-260-6-13688 en la fase CONFRONTACION\_CORRECTIVA Estación Asignada: X-CONFRONTADOR.260: No se puede desasociar el folio 260-121216 porque tiene modificaciones temporales. Este procedimiento se realizó también en la fase CALIFICACIÓN dando primero deshacer cambios y posteriormente desasociar folio pero el error continúa. Favor verificar el inconveniente que se está presentando para que en la oficina pueda continuar con el trámite del mismo.

Grupo: Cúcuta

### Analisis/Solución

El sistema debe permitir desasociar folios de turnos que no tienen datos temporales creados con estos turnos así el folio tenga datos temporales con otros turnos asociados.

**Analisis:** Al momento de replicar el caso haciendo un paso a paso en el código fuente del aplicativo existe una validación realizada por el método hasDatosTemporales del archivo JDOGenieFolioDAO.java este método verifica si el folio asociado al turno que desea desasociar posee datos temporales, se encontró que esta verificación está mal planteada, lo correcto es verificar si los datos temporales del folio asociado al turno no fueron creados por el turno que desea desasociar el folio, si esto es correcto, si se puede desasociar el folio, de lo contrario, emitir un mensaje que indique, que existen datos temporales en el folio.

**Solución:** Se implementa un nuevo método hasDatosTemporales2 que a diferencia del anterior este recibe un tercer parámetro el cual es el turno que está trabajando el folio y una vez allí se verifica si los datos temporales del folio no fueron generados por el turno que está tratando el folio, si esto es correcto, si se puede desasociar el folio.

### ARCHIVO JAVA:

Sir.forseti.dao.impl.jdogenie. JDOGenieFolioDAO.java

**Código Modificado Línea 31231 - 31235:**

```
// @author : Ellery David Robles G.
// @casoMantis : 08523.
if(this.hasDatosTemporales2(turno, folio, pm)){
```

**Código Modificado Línea 31386 - 31416:**

```
/**
 * @author : Ellery David Robles G.
 * @casoMantis : 08523.
 * @actaReq : 025_151 - Error al Desasociar el Folio.
 * @change : Indica si el folio tiene información temporal.
 * @param : turno.
 *          * @param : folio.
 *          * @param : pm.
 *          * @return : rta.
 */
private boolean hasDatosTemporales2(TurnoEnhanced turno, FolioEnhanced folio, PersistenceManager pm){
boolean rta = false;
//Si el folio NO es definitivo y es creado por el turno la respuesta es afirmativa
if(!folio.isDefinitivo()){
    rta = true;
}
//Se revisa si tiene anotaciones temporales
if(!rta){
    if(!folio.getAnotacionesTMPs().isEmpty()){
        List anotaList = folio.getAnotacionesTMPs();
        Iterator anotaItem = anotaList.iterator();
        if(anotaItem.hasNext()){
            AnotacionTMP anotaTMP = (AnotacionTMP)anotaItem.next();
            if(turno.getIdWorkflow().equals(anotaTMP.getNumRadicacion())){
                rta = true;
            }
        }
    }
}
return rta;
}
```

**026 PROCESO CERTIFICADOS ERROR AL ENVIAR AMPLIACION TRADICION****Requerimiento GLPI No.28103**

Turno: 2011-320-1-6254

Matricula(s):320-17458

Motivo Inconsistencia: Al momento de enviar el turno 2011-320-1-6254 a la fase de Ampliación de tradición el sistema no lo avanza a esta fase y lo toma como si fuera de antiguo sistema. El tipo de solicitud de este turno 2011-320-1-6254 es AMPLIACION\_TRADICION.

Por favor dejar el turno en la fase de AMPLIACION\_TRADICION en la estación LUCILA.ACEVEDO, puesto que al radicar el turno su solicitud es AMPLIACION\_TRADICION, para que la funcionaria encargada pueda trabajar el turno.

Grupo: San Vicente de Chucuri.

**Analisis/Solución**

- sir.core.negocio.modelo.constants.CFase.java
- sir.core.negocio.modelo.constants.CRol.java
- sir.hermes.dao.impl.jdogenie.JDOTurnosNuevosDAO.java

**ARCHIVO JAVA:**

sir.core.negocio.modelo.constants.CFase.java
--

**Código Insertado Línea 239 - 246:**

```
/**
 * @author : Ellery David Robles Gómez.
```

```
* @casoMantis      : 08539.  
* @actaRequerimiento : 151. Proceso Certificados Error al enviar Ampliacion Tradicion.  
* @change          : Identificador de la fase Ampliacion Tradicion proceso certificados.  
*/  
public static final String AMPLIACION_TRADICION = "AMPLIACION_TRADICION";
```

## **ARCHIVO JAVA:**

sir.core.negocio.modelo.constantes.CRol.java

## Código Insertado Linea 73 - 81:

```
/**  
 * @author : Ellery David Robles Gómez.  
 * @casoMantis : 08539.  
 * @actaRequerimiento : 151. Proceso Certificados Error al enviar Ampliacion Tradicion.  
 * @change : Se crea la variable "SIR_ROL_AMPLIACION_TRADICION" para identificar los  
 * tipos de certificados enviados a la fase "AMPLIACION_TRADICION".  
 */  
public static final String SIR_ROL_AMPLIACION_TRADICION = "SIR_ROL_AMPLIACION_TRADICION";
```

## **ARCHIVO JAVA:**

sir.hermod.dao.impl.jdogenie.JDOTurnosNuevosDAO.java

## **Código Inserido Linea 77.**

```
import gov.sir.core.negocio.modelo.constantes.CROL;
```

**Código Inserido Linea 783 - 797.**

```

    /**
     * @author : Ellery David Robles Gómez.
     * @casoMantis : 08539.
     * @actaRequerimiento : 151. Proceso Certificados Error al enviar Ampliacion Tradicion.
     * @change : Se crea if que identifique si el tipo de certificado es
"AMPLIACION_TRADICION", si su fase es "CER_SOLICITUD"
     *           y si tiene al menos un folio asociado, entonces que avace a
AMPLIACION_TRADICION, si estas condiciones no se cumplen
     *           entonces que avence a ANTIGUO_SISTEMA, para que le creen o asignen una
matricula, para luego avanzar a AMPLIACION_TRADICION.
     */
    if(turno.getIdProceso() == 1){
        LiquidacionTurnoCertificado ltc = (LiquidacionTurnoCertificado)
turno.getSolicitud().getLiquidaciones().get(0);
        if((turno.getSolicitud().getSolicitudFolios().size() >= 1) &&
(ltc.getTipoCertificado().getNombre().equals(CFase.AMPLIACION_TRADICION)) &&
(turno.getIdFase().equals(CFase.CER_SOLICITUD))){
            rta.put(Processor.FASE_DESTINO, CFase.CER_AMPLIACION_TRADICION);
            rta.put(Processor.ROL_DESTINO, CROL.SIR_ROL_AMPLIACION_TRADICION);
        }
    }
}

```

**Código Inserido Linea 799 - 808.**

```

} else {
    if((turno.getIdProceso() == 1) &&
        (ltc.getTipoCertificado().getNombre().equals(CFase.AMPLIACION_TRADICION)) &&
        (turno.getIdFase().equals(CFase.ANT_REVISION))){
        int thTam = turno.getHistorials().size();
        TurnoHistoria th = (TurnoHistoria) turno.getHistorials().get(thTam - 1);
        if(th.getFaseAnterior().equals(CFase.ANT_HOJA_RUTA)){
            rta.put(Processor.FASE_DESTINO, CFase.CER_AMPLIACION_TRADICION);
            rta.put(Processor.ROL_DESTINO, CRol.SIR_ROL_AMPLIACION_TRADICION);
        } else {

```

## **Código Inserido Linea 811 - 820.**

```
        }
    } else {
        rta.put(Processor.FASE_DESTINO, faseDestino);
        rta.put(Processor.ROL_DESTINO, idRol);
    }
}
} else {
    rta.put(Processor.FASE_DESTINO, faseDestino);
    rta.put(Processor.ROL_DESTINO, idRol);
}
```

sir.hermod.dao.impl.jdogenie.JDOTurnosNuevosDAO.java

cambiar las líneas 795 - 796:

```
rta.put(Processor.FASE_DESTINO, CFase.CER_AMPLIACION_TRADICION);
```

```
rta.put(Processor.ROL_DESTINO, CRol.SIR_ROL_AMPLIACION_TRADICION);
```

por:

```
rta.put(Processor.FASE_DESTINO, faseDestino);
rta.put(Processor.ROL_DESTINO, idRol);
```

## 029 ERROR AL AGREGAR NATURALEZA AL FOLIO

### REQUERIMIENTO GLPI No. 28224

Turno:2011-300-3-877

Matricula(s):300-283297

Motivo Inconsistencia:

El usuario José Luis García nos informa que al tratar agregar la naturaleza jurídica al folio 300-283297 con el turno de correcciones 2011-300-3-877 en la fase de ejecutar corrección simple aparece error en la página y no es posible agregarla.

Al tratar agregar la naturaleza jurídica en la anotación 06 del folio 300-283297 con el turno de correcciones 2011-300-3-877 en la fase de ejecutar corrección simple aparece error en la página tal como lo muestra la imagen adjunta y no es posible agregarla, el procedimiento se ha intentado en varios equipos y aparece el mismo error, el usuario que trabaja el turno es JOSE.GARCÍA, se requiere verificar y corregir el inconveniente para poder seguir con el trámite del turno.

Grupo: Bucaramanga.

### Analisi/Desarrollo

Este problema tiene un impacto Medio, debido a que se afecta la funcionalidad en el proceso de correcciones cuando se requiere cambiar naturaleza jurídica de anotaciones canceladoras. Esto evita que el usuario pueda hacer una corrección específica.

Se solicita a mesa de ayuda requerir al usuario el borrado de temporales y cookies del equipo de la Orip; desde donde se trabaja el turno y volver a intentar la operación. El error se vuelve a presentar. Se solicita la revisión del incidente desde el aplicativo por parte del equipo de desarrollo. El usuario que trabaja el turno está documentado por mesa de ayuda.

SIR-APP\web\WEB-INF\jspSeguro\correccion\CORRECC\_FOLIOEDIT\_ANOTACIONCANCELADORA.jsp

### JSP

SIR-APP\web\WEB-INF\jspSeguro\correccion\CORRECC\_FOLIOEDIT\_ANOTACIONCANCELADORA.jsp

### FUNCIÓN JS

```
function juridica(nombre,valor,dimensiones)
```

```

Se modifica la función de java script function juridica(nombre,valor,dimensiones)
function juridica(nombre,valor,dimensiones)
{
    /**
     * Fecha: 18-08-2011
     * Autor: Guillermo Cabrera
     * Propósito: Al editar una anotación canceladora, no estaba cargando el popup para seleccionar
     * la naturaleza jurídica.
     * Se modifica la función ya que se estaba generando mal la url al adicionarle la fecha del documento
     * que para este caso no se necesita.
     * Se setea en la sesión el atributo cancelar en true "session.setAttribute("cancelar", "true")" para que solo
     * se carguen las naturalezas jurídicas de cancelación.
     * Mantis: 0008593: Acta - Requerimiento No 029_151 - Error al agregar naturaleza al folio.
    */
    <%session.setAttribute("cancelar", "true"); %>
document.getElementById('natjuridica_id').value=valor+"_ID";
document.getElementById('natjuridica_nom').value=valor+"_NOM";
/*var fechaRad = document.getElementById('<%= AWModificarFolio.ANOTACION_FECHA_DOCUMENTO %>').value;
var url = nombre + '&<%= AWModificarFolio.ANOTACION_FECHA_RADICACION %>=' + fechaRad;*/
popup=window.open(nombre,valor,dimensiones);
popup.focus();
}

```

## 030 ERROR AL MODIFICAR CIUDADANOS EN TEMPORAL Y LOS CAMBIOS NO SE HACEN DEFINITIVOS

### REQUERIMIENTO GLPI No. 28293

Turno: 2011-148-6-1817

Matricula(s):148-14014

Motivo Inconsistencia: Se comunica la funcionaria Sixta Álvarez informando que al Agregar el usuario con cc 26044447 CRISTIAN.MORALES esta está relacionada con el turno de correcciones 2011-148- 3-169 el cual ya está desanotado.

Por favor verificar y corregir porque al agregar al ciudadano al turno de registro le está generando este error de que la cédula está relacionada con el turno 2011-148-3-169 si al revisar se encuentra finalizado, al verificar las salvedades encontramos que realizó una corrección del mismo ciudadano en la anotación 3 y 6.

Grupo: Sahagún

### En el despliegue 002\_6\_2 DESPLIEGUE se trabajó el Acta de Requerimiento No. 136\_ Anotaciones Temporales Folios

Anotaciones Temporales. Al realizar correcciones sobre ciudadanos en las anotaciones de los folios, los turnos de correcciones avanzan correctamente e imprimen correctamente la información modificada, pero al ingresar otro turno ya sea de registro o correcciones el sistema muestra temporal la anotación o anotaciones donde se hizo edición o corrección de ciudadano y es necesario que por medio de la mesa de ayuda se eliminen estos datos temporales que quedan en los folios.

Por favor escalar a desarrollo para corregir este tema relacionado con los datos temporales en las anotaciones ya que estos datos deben quedar definitivos en su totalidad con el turno de correcciones cuando realiza su flujo normal y no se debe mostrar ningún temporal en los folios.

Por favor crear acta de requerimiento a desarrollo para su solución general. Acta de referencia No 078 de la contratación 374.

### Analisis/Desarrollo

Con un turno de corrección se hacen los cambios al ciudadano, el turno finaliza bien y se imprime el

formulario temporal. Los cambios no se hacen definitivos y aparecen ciudadanos en temporales rezagados.

Se realizó un proceso en línea con el usuario y no se pudo replicar el error, las incidencias

reportadas con este mismo error son: 28441, 26412, 24141, 23854, 22193, 21042, 17975, 17745, 17047, 8680. No se pudo replicar el caso.

**Análisis:**

Al momento de hacer permanentes los cambios de un ciudadano que interviene en una anotación, SIR inicialmente remueve el objeto CiudadanoAnotacionEnhanced de memoria, luego de la base de datos, realiza un flush y por último, vuelve a agregar el CiudadanoAnotacionEnhanced con los cambios ingresados por el usuario.

En el último ciclo de la lista de las anotaciones de ciudadano temporales, los cambios realizados sobre el ciudadano no son enviados a la base de datos, debido a que el VersantPersistenceManager.flush() se encuentra antes de que el ciudadano sea agregado a la base de datos con los cambios dados por el usuario en el proceso de corrección de anotación de ciudadano.

Debido a lo explicado anteriormente, al momento de abstraer de la base de datos los registros temporales de los ciudadanos por folio; el último ciudadano temporal no será regresado por la base de datos ya que al no realizar flush al final de este ciclo, el proceso de agregar el ciudadano con los cambios solo se verá reflejado en el momento de realizar PersistentManager.commit().

Por las razones dadas, en ciertas circunstancias quedaban datos temporales que posteriormente no permitían realizar procesos de registro o corrección sobre esa anotación o ciudadano específico.

Para replicar este error se deben dar las siguientes condiciones:

- En la anotación deben haber 2 ciudadanos. Uno propietario y uno no-propietario.
- La anotación debe ser de mutación (probado en compraventa).
- Se editan todos los ciudadanos de una anotación. Estos ciudadanos deben tener como tipo de documento cédula.
- Se debe agregar un ciudadano no-proprietario("DE") con tipo de documento secuencia.

**CLASE:**

**gov.sir.forseti.dao.impl.jdogenie.JDOGenieFolioDAO**

**Inserción de Código Línea 16433:**

```
/**  
 * Desarrollador: Diana Lora  
 * Mantis 0008623: Acta - Requerimiento No 030_151 - Error al Modificar  
 * Ciudadanos en Temporal y los cambios no se hacen definitivos.  
 */  
pm2.flush();
```

**032 ACTUALIZACIÓN NORMATIVIDAD DE REGISTRO RESOLUCIÓN 5333 DE 2011  
(Informativo)**

**Actualización Normatividad de Registro Resolución 5333 de 2011.**

Se solicita se actualice la tabla de naturaleza jurídica, con el fin de que en el sistema SIR la descripción del código de Naturaleza Jurídica 0902 sea la especificada en la resolución 5333 de 2011,"CORRECCIÓN DEL TITULO RESPECTO A LOS LINDEROS Y AREA (Artículo 49 del

Decreto 2148 de 1983), se deben tener en cuenta las posibles implicaciones en la calificación de documentos y correcciones.

De otra parte se debe guardar el histórico de la información que existe registrada con ese código hasta la fecha de la actualización

Grupo: SNR

Reespecificación recibida por correo por parte de los funcionarios Yamile Moreno y Dora Pachón el día 13 de Julio del 2011.

**De acuerdo a lo verificado en la Base de datos se establece que 2.248 anotaciones tienen el id\_naturaleza\_juridica el código '0902' sin información en el campo NTCN\_especificación,, por lo**

cual se solicita que antes de realizar la actualización requerida, se actualicen estas anotaciones incluyendo en el campo de especificación, la descripción actual del código de naturaleza jurídica 0902 que corresponde a ACTUALIZACION AREA, y posteriormente realizar la actualización en la tabla básica de naturalezas jurídicas.

Así mismo, validar que en la expedición de certificados, formularios de calificación, formularios de correcciones, entre otros, la especificación de las anotaciones no se vea alterada por la actualización de este código jurídico.

**Validaciones a realizar:** Verificar que todas las anotaciones que estén inscritas con el código 0902, tengan información en el campo ntcn\_especificación.

#### Analisis/Desarrollo

Se adjuntó documento SOLICITUD+DE+REQUERIMIENTOnatujuris.doc con la descripción y análisis de impacto del requerimiento solicitado. Se envía correo electrónico al Ingeniero Enrique Venegas informándole el estado del requerimiento el día 12 de Julio del 2011.

#### Análisis:

Se realizó un análisis para determinar el impacto de este requerimiento, a continuación se describe la funcionalidad actual del aplicativo SIR.

Al realizar la modificación de la descripción en la tabla **sir\_ne\_naturaleza\_juridica** el texto de la descripción de la anotación (**sir\_ne\_anotacion**) no se afecta, ya que en esta tabla se guarda el código de la naturaleza y se guarda la descripción (texto) de la especificación (NTCN\_ESPECIFICACION). Es de aclarar que esta especificación en muchas ocasiones no corresponde al id de la naturaleza jurídica ya sea porque son folios que han sido migrados con esa información. Ejemplo el folio 146- 7971 anotaciones 7 y 8.

ID_MATRICULA	ID_ANOTACION	ID_NATURALEZA_JURIDICA	NTCN_ESPECIFICACION
146-7941	7	0902	ACTUALIZAR AREA 336 M2 DE ACUEDO CON LA RESL 234 70075 DEL 10.10.02 I.G.A.C
146-7941	8	0902	COMPROVANTIA PARCIAL (315M2).

La especificación solo cambia si se realizan correcciones sobre la anotación, cambiando el id de la naturaleza jurídica por la aplicación SIR.

Al imprimir certificados, se imprime el valor que se encuentra en la columna ntcn\_especificacion si esta es no es nula, pero al ser nula si trae la información de la tabla actualizada (**sir\_ne\_naturaleza\_juridica**). Lo que se puede presentar para confusiones ya que hay muchos registros con el código 0902 que tienen su especificación con valor NULL (ver Excel adjunto, anotaciones con id naturaleza jurídica 0902).

En nuestro ambiente de pruebas (192.168.202.9:8100/SNR/inicio.view) se actualizó la naturaleza jurídica 0902 con el texto suministrado **CORRECCIÓN DEL TITULO RESPECTO A LOS LINDEROS Y AREA (Artículo 49 del Decreto 2148 de 1983)** ahí se pueden realizar pruebas y validar el impacto del cambio solicitado.

Para finalizar, en el requerimiento se dice lo siguiente “**De otra parte se debe guardar el histórico de la información que existe registrada con ese código hasta la fecha de la actualización**”.

Se Adjunta correo enviado el día de ayer 14-07-2011. Se está a la espera de las pruebas y aprobación del cambio para su implementación en el ambiente productivo.

----- Mensaje original -----

Asunto: Re: Reenviar: RE: SOLICITUD DE REQUERIMIENTOnatujurisv1

Fecha: Thu, 14 Jul 2011 09:13:22 -0500

De: Guillermo Cabrera <[guillermo.cabrera@intek.com.co](mailto:guillermo.cabrera@intek.com.co)>

Para: [javier.rocha@intek.com.co](mailto:javier.rocha@intek.com.co)

CC: Fernando Padilla Velez <[Fernando.padilla@intek.com.co](mailto:Fernando.padilla@intek.com.co)>, Jorge Enrique Venegas Arango <[enrique.venegas@intek.com.co](mailto:enrique.venegas@intek.com.co)>, 'Jorge Arturo Ríos Sánchez' <[jorge.rios@intek.com.co](mailto:jorge.rios@intek.com.co)>

Buenos Días Javier,

Ya se realizó el punto 1 en el ambiente de pruebas (<http://192.168.202.9:8100/SNR/inicio.view>, [^] Base de datos 192.168.202.25 (suric)) por favor realizar los puntos 2, 3 y 4.

Se ejecutó el script SCRIPT\_MANTIS\_8771.sql de actualización.

Igualmente se modificó la descripción de la naturaleza jurídica 0902 por CORRECCIÓN DEL TITULO RESPECTO A LOS LINDEROS Y AREA. (Artículo 49 del Decreto 2148 de 1983).

Adjunto correo de Jorge Ríos donde informa a la SNR la actividad realizada.

----- Mensaje original -----

Asunto: Actualizacion Codigo de Naturaleza Juridica 0902

Fecha: Thu, 4 Aug 2011 16:10:40 -0500

De: Jorge Arturo Ríos Sánchez <[jorge.rios@intek.com.co](mailto:jorge.rios@intek.com.co)>

Para: <[luis.romero@supernotariado.gov.co](mailto:luis.romero@supernotariado.gov.co)>, <[edgar.coronado@intek.com.co](mailto:edgar.coronado@intek.com.co)>, <[edgar.hoyos@intek.com.co](mailto:edgar.hoyos@intek.com.co)>, <[nazly.valencia@co.pwc.com](mailto:nazly.valencia@co.pwc.com)>, "Soporte SIR SNR" <[soportesir@supernotariado.gov.co](mailto:soportesir@supernotariado.gov.co)>, "Guillermo Cabrera" <[guillermo.cabrera@intek.com.co](mailto:guillermo.cabrera@intek.com.co)>, <[yamile.moreno@supernotariado.gov.co](mailto:yamile.moreno@supernotariado.gov.co)>, "Dany Liliana Agon Reyes" <[dany.agon@globalbis.com](mailto:dany.agon@globalbis.com)>, <[ELBA.CORREDOR@supernotariado.gov.co](mailto:ELBA.CORREDOR@supernotariado.gov.co)>, <[andres.cuadros@co.pwc.com](mailto:andres.cuadros@co.pwc.com)>, "fercho" <[fernando.padilla@intek.com.co](mailto:fernando.padilla@intek.com.co)>, <[nestor.sanchez@supernotariado.gov.co](mailto:nestor.sanchez@supernotariado.gov.co)>

CC: <[jorge.rios@intek.com.co](mailto:jorge.rios@intek.com.co)>

Buenas Tardes

Dando cumplimiento al mandato de la SNR sobre el desarrollo para la actualización de la naturaleza Jurídica del código 0902 se ejecutaron los scripts de actualización.

Se validó con el área funcional y se actualizó la descripción de la naturaleza jurídica 0902, quedando en producción así:

CORRECCIÓN DEL TITULO RESPECTO A LOS LINDEROS Y AREA. (Artículo 49 del Decreto 2148 de 1983).

Se ejecutó SCRIPT MANTIS 8771.sql.

#### 034 ERROR PROCESO CORRECIONES - CAMBIO INFORMACION ANOTACION

En el proceso de registro de documentos, con el usuario SUSANAS. RIVERA, se incluyó el folio 368-9292 y 368-48026, el sistema generó el turno de documento 2010-368-6-1519 y certificado asociado 2010-368-1-10440 y 2010-368-1-10441. El turno es avanzado a confrontación al usuario ROMAN.USECHA y avanzado a la fase REG\_REPARTO, usuario MILDRED. BOCANEGRA, el sistema le asigna el turno al usuario. Para el folio 368-9292, se crea la anotación dieciocho (18), para el folio 368-48026, se crea la anotación tres (3), con la misma información en los dos folios. El turno es avanzado hasta finalizarlo.

Desde el proceso de correcciones, con el usuario SUSANAS.RIVERA, se incluyó los folios 368-9292 y 368-48026, el sistema generó el turno de documento 2010-368-3-115. Desde la fase REVISION\_ANALISIS, se asignan permisos de ciudadano y número de documento. El turno es avanzado a CORRECCION\_SIMPLE, usuario ROMAN. USECHE, Para el 368-9292, se corrige la anotación diez y seis (16), y para el folio 368-48026 en la anotación uno y dos (1 y 2) y se corrige el nombre Jose Hernesto por José Pablo. Se toma nuevamente el folio 368-9292 y se realiza corrección sobre la anotación 18, turno de radicación 2010-368-6-1519 escritura 4 del 02/08/2010, notaría 19 de Purificación y es corregido por escritura 15 del 02/08/2010, Notaría diecinueve de Purificación. El folio 368-48026 no se realiza modificación sobre anotaciones. Se revisa el "FORMULARIO DE CORRECCIÓN" y se visualiza que para el folio 368-9292 se realizaron corrección sobre la anotación diecisésis y dieciocho (16 y 18); en el folio 368-48026 informa que los datos modificados son sobre la anotación uno y dos (1 y 2). El turno es finalizado satisfactoriamente.

Al ingresar por Pantalla Administrativa, se ingresó a los folios 368-9292 y 368-48026 con el fin de verificar que la información corregida con el turno 2010-368-3-115 haya quedado en definitivo y se observó que la información corregida en el folio 368-9292 anotación 18, fue modificada en el folio

**Analisis/Desarrollo**

- Se modificaron los siguientes archivos en SIR:
  - WEB-INF.jspSeguro WEB-INF.jspSeguro.correccion.EDITAR\_ANOTACION.jsp.
  - sir.core.web.acciones.correccion.AWModificarFolio.java.
  - sir.core.negocio.modelo.Folio.java.
  - sir.core.negocio.acciones.correccion.ANCorreccion.java.
  - sir.core.web.navegacion.correccion. ControlNavegacionModificacionFolio.java.
  - sir.core.eventos.correccion. EvnCorreccion.java.
  - sir.core.eventos.correccion. EvnRespCorreccion.java.
  - sir.forseti.dao.impl.jdogenie. JDOGenieFolioDAO.java.
  
- Se modificaron clases en GeneralSIR:
  - co.com.iridium.generalSIR.dao.ValidacionesSIRDALO.java.
  - co.com.iridium.generalSIR.negocio.validaciones.ValidacionesSIR.java.

**ARCHIVO JSP:****WEB-INF.jspSeguro.correccion.EDITAR\_ANOTACION.jsp.****Código Modificado Línea 522 - 524:**

```
String consultaAnotacionesResp = (String)session.getAttribute(AWModificarFolio.CONULTA_ANOTACIONES_RESP);
String existenAnotacionesAsociadas = null;
int existenAnotacionesAsociadasInt = 1;
```

**Código Insertado Línea 526 - 530:**

```
<form action="modificacion.do" name="RELOADDATAFORM" id="RELOADDATAFORM" method="post">
  <input type="hidden" name="RELOAD_DATA" id="RELOAD_DATA" value="" />
  <input type="hidden" name="SOURCE_PAGE" id="SOURCE_PAGE" value="<%param_SourcePage%>" />
  <input type="hidden" name="<%=WebKeys.ACCTION%>" id="<%=WebKeys.ACCTION%>" value="<%=
gov.sir.core.web.acciones.correccion.AWModificarFolio.FOLIOEDIT_RELOADBYUPDATECIUDADANO_EDITANOTACIONNORMAL_PAGERENDERING_PROCESS_ACTION %>" />
</form>
```

**Código Modificado Línea 527 - 530:**

```
<form action="modificacion.do" name="ANOTACIONES_ASOCIADAS" id="ANOTACIONES_ASOCIADAS">
  <input type="hidden" name="<%=AWModificarFolio.ANOTACION_ID_ANOTACION%>" value="<%idAnotacion%>" />
  <input type="hidden" name="<%=AWModificarFolio.FOLIO_NUM_FOLIO%>" value="<%idMatricula%>" />
  <input type="hidden" name="<%=WebKeys.ACCTION%>" id="<%=WebKeys.ACCTION%>" value="<%=
gov.sir.core.web.acciones.correccion.AWModificarFolio.ANOTACIONES_ASOCIADAS%>" />
```

**Código Modificado Línea 532 - 538:**

```
<%if(consultaAnotacionesResp == null){%
  <script type="text/javascript">document.ANOTACIONES_ASOCIADAS.submit();</script>
<%} else {
  existenAnotacionesAsociadas =
(String)session.getAttribute(AWModificarFolio.EXISTEN_ANOTACIONES_ASOCIADAS);
  existenAnotacionesAsociadasInt = Integer.parseInt(existenAnotacionesAsociadas);
}>
<form action="modificacion.do" name="ANOTACION" id="ANOTACION" method="post">
```

**Código Insertado Línea 547:**

```
<%if(existenAnotacionesAsociadasInt > 1){%>
```

**Código Modificado Línea 551 - 552:**

```
<td background="<%request.getContextPath()%>/jsp/images/tabla_gral_bgn002.gif">
  <table border="0" cellpadding="0" cellspacing="0">
```

**Código Modificado Línea 554:**

```
<td background="<%request.getContextPath()%>/jsp/images/tabla_gral_bgn001.gif"
class="titulotbgral">Advertencia al momento de corregir anotaciones</td>
```

## Código Insertado Línea 556:

```
<td width="20"></td>
```

## Código Modificado Línea 558 - 559:

```
</td>  
</table>
```

## Código Insertado Linea 566 - 567:

```
<td>
  <table border="0" cellpadding="0" cellspacing="0" width="100%">
```

## Código Modificado Línea 569 - 617:

```

<td width="12"></td>
<td width="12">&nbsp;</td>
</tr>
<tr>
<td>&nbsp;</td>
<td background="<%request.getContextPath()%>/jsp/images/tabla_gral_bgn003.gif">&nbsp;</td>
<td class="tdtablaanexa02">
```

**Código Insertado Linea 1938:**  
`</form>`

**ARCHIVO JAVA:**

`sir.core.web.acciones.correccion.AWModificarFolio.java.`

**Código Insertado Línea 738 - 752:**

```

/** ANOTACIONES ASOCIADAS Caso Mantis: 7275 Autor: Ellery Robles Gómez*/
public static final String ANOTACIONES_ASOCIADAS = "ANOTACIONES_ASOCIADAS";

/** Esta constante identifica el numero del folio*/
public static final String FOLIO_NUM_FOLIO = "FOLIO_NUM_FOLIO";

/** Esta constante identifica si hay anotaciones asociadas*/
public static final String EXISTEN_ANOTACIONES_ASOCIADAS = "EXISTEN_ANOTACIONES_ASOCIADAS";

/** Esta constante identifica si hay anotaciones asociadas*/
public static final String CONSULTA_ANOTACIONES_RESP = "CONSULTA_ANOTACIONES_RESP";

/** Esta constante identifica el tipo de cambio a la anotacion*/
public static final String TIPO_CAMBIO_ANOTACION = "TIPO_CAMBIO_ANOTACION";
```

**Código Modificado Línea 1083 - 1085:**

```

}else if(ANOTACIONES_ASOCIADAS.equals(accion)){
    return anotacionesAsociadas(request);
} else {
```

**Código Insertado Línea 5644:**

```
String tipoCambio = null;
```

**Código Insertado Línea 7336 - 7343:**

```

tipoCambio = request.getParameter(AWModificarFolio.TIPO_CAMBIO_ANOTACION);
if(tipoCambio != null){
    if (tipoCambio.equals("INDIVIDUAL")){
        t2_folio.setTipoCambio(true);
    }else if (tipoCambio.equals("MULTIPLE")){
        t2_folio.setTipoCambio(false);
    }
}
```

**Código Insertado Línea 7680 - 7681:**

```

session.removeAttribute(AWModificarFolio.CONSULTA_ANOTACIONES_RESP);
session.removeAttribute(AWModificarFolio.EXISTEN_ANOTACIONES_ASOCIADAS);
```

**Código Insertado Línea 13403 - 13405:**

```

}else if
(respuesta.getTipoEvento().equals(EvnRespCorreccion.ANOTACIONES_ASOCIADAS_OK)) {
    session.setAttribute(AWModificarFolio.CONSULTA_ANOTACIONES_RESP, "1");
    session.setAttribute(AWModificarFolio.EXISTEN_ANOTACIONES_ASOCIADAS,
    respuesta.getPayload());
```

**Código Insertado Línea 15567 - 15588:**

```

/**
 * Este método se encarga de tomar los valores del <code>HttpServletRequest</code> para
 * generar un evento de consulta de usuarios por círculo
 * @param request
 * @return evento <code>EvnAdministracionFenrir</code> con la información del círculo.
 * @throws AccionWebException
 * Caso Mantis: 7275 Autor: Ellery Robles Gómez*/
private EvnCorreccion anotacionesAsociadas(HttpServletRequest request)
throws AccionWebException {
```

```

HttpSession session = request.getSession();
ValidacionParametrosException exception = new ValidacionParametrosException ();

String idAnotacion = request.getParameter(AWModificarFolio.ANOTACION_ID_ANOTACION);
String idFolio = request.getParameter(AWModificarFolio.FOLIO_NUM_FOLIO);

//Usuario usuario = (Usuario) session.getAttribute(SMARTKeys.USUARIO_EN_SESION);
EvnCorreccion evento = new EvnCorreccion(idFolio, idAnotacion, EvnCorreccion.ANOTACIONES_ASOCIADAS);
session.setAttribute(AWModificarFolio.CONSULTA_ANOTACIONES_RESP, "1");
session.setAttribute(AWModificarFolio.EXISTEN_ANOTACIONES_ASOCIADAS,
"EXISTEN_ANOTACIONES_ASOCIADAS");
return evento;
}

```

**ARCHIVO JAVA:**

sir.core.negocio.modelo.Folio.java.

**Código Insertado Línea 37:**

```
private boolean tipoCambio;
```

**Código Insertado Línea 493 - 495:**

```
public boolean getTipoCambio() {
    return tipoCambio;
}
```

**Código Insertado Línea 497 - 499:**

```
public void setTipoCambio(boolean tc) {
    this.tipoCambio = tc;
}
```

**ARCHIVO JAVA:**

sir.core.negocio.acciones.correccion.ANCorreccion.java.

**Código Insertado Línea 3:**

```
import co.com.iridium.generalsIR.comun.GeneralSIRException;
import co.com.iridium.generalSIR.negocio.validaciones.ValidacionesSIR;
```

**Código Insertado Línea 145 - 146:**

```
import java.util.logging.Level;
import java.util.logging.Logger;
```

**Código Insertado Línea 385 - 387:**

```
else if( EvnCorreccion.ANOTACIONES_ASOCIADAS.equals( local_TipoEvento ) ) {
    return anotacionesAsociadas( evento );
}
```

**Código Insertado Línea 7569 - 7574:**

```
/**
 * @param evento
 * @return
 * Caso mantis: 7275 Autor: Ellery Robles Gómez*/
private EventoRespuesta anotacionesAsociadas(EvnCorreccion evento) throws EventoException {
    String existen = null;
```

**Código Modificado Línea 7576 - 7580:**

```
ValidacionesSIR validacion = new ValidacionesSIR();
try {
    existen = validacion.NumeroAnotacionAsociadas(evento.getIdFolio(), evento.getIdAnotacion());
} catch (GeneralSIRException ex) {
    Logger.getLogger(ANCorreccion.class.getName()).log(Level.SEVERE, null, ex);
```

**Código Insertado Línea 7582 - 7586:**

```
EvnRespCorreccion evRespuesta =
    new EvnRespCorreccion(existen, EvnRespCorreccion.ANOTACIONES_ASOCIADAS_OK);
return evRespuesta;
}
```

**ARCHIVO JAVA:**

sir.core.web.navegacion.correccion.ControlNavegacionModificacionFolio.java.

**Código Insertado Línea 211:**

```
public static final String ANOTACIONES_ASOCIADAS_OK = "ANOTACIONES_ASOCIADAS_OK";
```

**Código Insertado Línea 591 - 592:**

```
}else if(AWModificarFolio.ANOTACIONES_ASOCIADAS.equals(accion)){
    return ANOTACIONES_ASOCIADAS_OK;
```

**ARCHIVO JAVA:**

sir.core.eventos.correccion.EvnCorreccion.java.

**Código Insertado Línea 591 - 592:**

```
public String idAnotacion;
```

```
public String idFolio;
```

**Código Insertado Línea 360 - 366:**

```
* Constante que identifica el Evento para encontrar el numero de anotaciones asociadas a otros folio
* Autor : Ellery D. Robles Gomez
* Caso Mantis : 07275
*/
public final static String ANOTACIONES_ASOCIADAS = "ANOTACIONES_ASOCIADAS";
```

```
/**
```

**Código Insertado Línea 368 - 379:**

```
* @param idAnotacion
* @param idFolio
* @param tipoAccion
*/
public EvnCorreccion(String idFolio, String idAnotacion, String tipoAccion) {
    super(null, tipoAccion);
    this.idAnotacion = idAnotacion;
    this.idFolio = idFolio;
}
/**
```

```
* @param usuario
```

**Código Insertado Línea 549 - 564:**

```
public void setIdAnotacion(String idAnotacion) {
    this.idAnotacion = idAnotacion;
}

public String getIdAnotacion() {
    return idAnotacion;
}

public void setIdFolio(String idFolio) {
    this.idFolio = idFolio;
}

public String getIdFolio() {
    return idFolio;
}
```

**ARCHIVO JAVA:**

sir.core.eventos.correccion.EvnRespCorreccion.java.

**Código Insertado Línea 174 - 178:**

```
/**
 * Constante que identifica que se desea notificar al Ciudadano sobre su solicitud de corrección.
*/
public final static String ANOTACIONES_ASOCIADAS_OK = "ANOTACIONES_ASOCIADAS_OK";
```

**ARCHIVO JAVA:**

sir.forseti.dao.impl.jdogenie.JDOGenieFolioDAO.java.

**Código Insertado Línea 245:**

```
private boolean tipoCambio;
```

**Código Insertado Línea 11371:**

```
tipoCambio = datos.getTipoCambio();/* Caso Mantis: 7275 Autor: Ellery Robles Funcion: Permite identificar si el
cambio de la anotacion es de tipo individual o multiple*/
```

**Código Insertado Línea 28958 - 28972:**

```

if(tipoCambio){
    DocumentoTMP docTMP = new DocumentoTMP(aux);
    //Hallando secuencia del documento definitivo

    docTMP.setIdDocumentoTmp(String.valueOf(this.getSecuencial(CDocumento.TABLE_NAME, null)));
        this.setDocumentoTMPToAnotacion(anotacion, docTMP, pm);
        pm.makePersistent(docTMP);
        this.actualizarDocumentoTMP(docTMP, doc, pm);
    }else{
        this.actualizarDocumentoTMP(auxTMP, doc, pm);
        anotacion.setDocumentoTMP(auxTMP);
    }
} else {
    if(tipoCambio){
        DocumentoTMP docTMP = new DocumentoTMP(aux);
        //Hallando secuencia del documento definitivo

        docTMP.setIdDocumentoTmp(String.valueOf(this.getSecuencial(CDocumento.TABLE_NAME, null)));
            this.setDocumentoTMPToAnotacion(anotacion, docTMP, pm);
            pm.makePersistent(docTMP);
            this.actualizarDocumentoTMP(docTMP, doc, pm);
        }else{
            DocumentoTMP docTMP = new DocumentoTMP(aux);
            this.setDocumentoTMPToAnotacion(anotacion, docTMP, pm);
            pm.makePersistent(docTMP);
            this.actualizarDocumentoTMP(docTMP, doc, pm);
        }
}

```

#### ARCHIVO JAVA:

src.org.auriga.smart.web.xml

#### Código Insertado Línea 1993 - 1997:

```

<!-- Caso mantis:7275 Autor: Ellery Robles Gómez -->
<navegacion>
    <resultado>ANOTACIONES_ASOCIADAS_OK</resultado>
    <url-redireccion>editar.anotacion.view</url-redireccion>
</navegacion>

```

#### ARCHIVO JAVA:

co.com.iridium.generalSIR.dao.ValidacionesSIRDAO.java.

#### Código Insertado Línea 221 - 250:

```

/**
 * Metodo que retorna el numero de anotaciones asociadas a un folio.
 * @param String idMatricula.
 * @return String idAnotacion.
 * @author Ellery Davi Robles Gómez.
 * Caso Mantis : 07275.
 */
public String numeroAnotacionesAsociadas(String idMatricula, String idAnotacion) throws SQLException {
    PreparedStatement ps = null;
    ResultSet rs = null;
    StringBuilder sb = new StringBuilder();
    String numeroAnotAsoc = "0";
    try {
        sb.append("SELECT COUNT(9) AS NUM_ANOT_ASOC FROM SIR_NE_ANOTACION WHERE ID_DOCUMENTO =");
        (SELECT id_documento FROM sir_ne_anotacion WHERE id_matricula = ? AND id_anotacion = ?)");
        ps = conexionBaseDatos.prepareStatement(sb.toString());
        ps.setString(1, idMatricula);
        ps.setString(2, idAnotacion);
        rs = ps.executeQuery();
        if (rs.next()) {
            numeroAnotAsoc = rs.getString("NUM_ANOT_ASOC");
        }
        return numeroAnotAsoc;
    } catch (SQLException sqle) {
        throw sqle;
    } finally{
        cerrarConexion(null,ps,null);
    }
}

```

**ARCHIVO JAVA:**

co.com.iridium.generalSIR.negocio.validaciones.ValidacionesSIR.java.

**Código Insertado Línea 139 - 155:**

```
/**
 * Metodo que consulta si una anotaciones asociada a otras anotaciones de otros folios.
 * @param String IdMatricula, IdAnotacion.
 * @return String
 * @author Ellery Davi Robles Gómez
 * Caso Mantis : 07275
 */
public String NumeroAnotacionAsociadas(String idMatricula, String idAnotacion) throws GeneralSIREException{
    ValidacionesSIRDAO validacionesSIRDAO = new ValidacionesSIRDAO();
    try{
        return validacionesSIRDAO.numeroAnotacionesAsociadas(idMatricula, idAnotacion);
    } catch (SQLException ex){
        throw new GeneralSIREException("Ocurrio un error al tratar "
            + "de consultar el folio " + idMatricula.toString() + ", si posee mas de una anotacion asociada a otros
folios." + ex.getMessage(),ex);
    }
}
```

Se realizaron pruebas con Omaira Triana de Mesa de Ayuda, obteniendo resultado positivo acerca de la corrección del caso.

....-- MODIFICACIONES AL LA DOCUMENTACIÓN --:....

**ARCHIVO JSP:**

WEB-INF.jspSeguro.correccion.EDITAR\_ANOTACION.jsp.

**Código Insertado Línea 553 - 559:**

```
<xRegionTemplate:DisplayPoint
    regionId="htRgn_AnotacionesCambioDocumento"
    regionName="anotacion - tipo de cambio"
    regionDescription="tipo de modificacion en la anotacion"
    debugEnabled="false"
    displayExtraMessage="null,true,false"

regionManager="<%=(gov.sir.core.web.helpers.correccion.region.model.RegionManager)pageLocalRegionManager%>">
```

**Código Insertado Línea 602:**

&lt;/xRegionTemplate:DisplayPoint&gt;

**ARCHIVO JAVA:**

sir.core.web.helpers.correccion.region.model.EditFoliosRegionManager.java

**Código Insertado Línea 40:**

, new Region.Binder( ANOTACION\_DOCUMENTO\_ID, "htRgn\_AnotacionesCambioDocumento" )

....-- 2. MODIFICACIONES AL LA DOCUMENTACIÓN --:....

**ARCHIVO JSP:**

WEB-INF.jspSeguro.correccion.EDITAR\_ANOTACION.jsp.

**Código Modificado Línea 553:**

&lt;%--&lt;xRegionTemplate:DisplayPoint

**Código Modificado Línea 602 - 609:**

```
</xRegionTemplate:DisplayPoint>--%>
<tr>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
    <td><input type="hidden" name="<%=_AWModificarFolio.TIPO_CAMBIO_ANOTACION%>" value
="INDIVIDUAL"/></td>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
</tr>
```

### 041 PROBLEMA\_FOLIO\_COLUMNA.FLIO\_LAST\_ID\_ANOTACIÓN (Informativo)

Se han detectado folios (1791) donde la información de la columna flio\_last\_id\_anotacion no coincide con el número de anotaciones definitivas que tiene actualmente.

Ejemplo el folio 162-30689.

Se realiza la consulta sobre la tabla SIR\_NE\_FOLIO y se observa que los id de anotación temporal y definitivo se encuentra en 3.

```
SELECT f.flfo_last_id_anotacion , f.flfo_last_id_anotacion_tmp  
FROM SIR.SIR_NE_FOLIO F  
WHERE F.ID_MATRICULA = '162-30689'
```

Resultado de la Consulta

FLFO_LAST_ID_ANOTACION	FLFO_LAST_ID_ANOTACION_TMP
1	3

Se realiza la consulta para determinar el id máximo en anotaciones definitivas.

```
SELECT MAX(TO_NUMBER(an.id_anotacion))  
FROM SIR.SIR_NE_ANOTACION AN  
WHERE ID_MATRICULA = '162-30689'
```

Resultado de la Consulta

MAX(TO_NUMBER(AN.ID_ANOTACION))
4

Al realizar un detalle de las información de las anotaciones, se observa que hay una inconsistencia con respecto al orden, ya que la anotación con id 4 tiene orden 5.

```
SELECT an.id_anotacion, an.id_matricula, an.ntcn_orden  
FROM SIR.SIR_NE_ANOTACION AN  
WHERE ID_MATRICULA = '162-30689'
```

Resultado de la Consulta

ID_ANOTACION	ID_MATRICULA	NTCN_ORDEN
1	162-30689	01
3	162-30689	3
2	162-30689	2
4	162-30689	5

Se crea el turno de registro 2011-162-6-1701, con el cual se crea la anotación temporal con la siguiente información.

```

SELECT ant.id_anotacion_tmp, ant.id_matricula, ant.nttm_orden
FROM SIR.SIR_NE_ANOTACION_TMP ANT
WHERE ID_MATRICULA = '162-30689'

```

Resultado de la Consulta		
ID_ANOTACION_TMP	ID_MATRICULA	NTTM_ORDEN
1 4	162-30689	6

Al avanzar el turno de la fase FIRMA DE REGISTRO, se consulta la información definitiva y se observa que la anotación con id\_anotacion 4 quedó registrada con orden 6. Al verificar toda la información de la anotación 4 esta se sobrescribió con la anotación del turno 2011-162-6-1701

```

SELECT an.id_anotacion, an.id_matricula, an.ntcn_orden
FROM SIR.SIR_NE_ANOTACION AN
WHERE ID_MATRICULA = '162-30689'

```

ID_ANOTACION	ID_MATRICULA	NTCN_ORDEN
1 1	162-30689	01
2 3	162-30689	3
3 2	162-30689	2
4 4	162-30689	6

Crear un procedimiento almacenado a nivel de base de datos para actualizar los 1791 folios inconsistentes. Se debe corregir la columna flio\_last\_id\_anotacion de la tabla SIR\_NE\_FOLIO para que quede consistente de acuerdo al número de anotaciones (SIR\_NE\_ANOTACION).

#### Analisis/Desarrollo

Adjunto el archivo SP\_UPDATE\_LAST\_ANOTA.sql que contiene el código fuente del procedimiento almacenado SP\_UPDATE\_LAST\_ANOTA. Con este procedimiento se actualizan los folios que ya se encuentran inconsistentes, por lo tanto no es la solución definitiva al incidente.

Este caso esta relacionado con el caso mantis 5306, que era el que daba origen a la inconsistencia.

El script SP\_UPDATE\_LAST\_ANOTA.sql se ejecutó para solucionar los casos que ya se encontraban con la inconsistencia.

Se ejecuto script SP\_UPDATE\_LAST\_ANOTA.sql el día 19 de Septiembre del 2011 actualizando 1879 registros.

#### 042 ERROR AL SEGREGAR Y HEREDAR ANOTACIÓN

Turno: 2011-140-6-8935

Matricula(s):140-5628

Motivo Inconsistencia: Al realizar el procedimiento de segregación sobre el folio 140-5628 y heredar la

anotación 1263 "DEMANDA EN PROCESO DE PERTENENCIA" el sistema hereda una anotación incorrecta con naturaleza "COMPRAVENTA PARCIAL" en el folio nuevo.

Favor verificar y corregir inconsistencia presentada en el folio al intentar segregar y heredar la anotación 1263 DEMANDA EN PROCESO DE PERTENENCIA, el usuario con el cual se está trabajando el turno es LINA.MERCADO.

### Analisis/Desarrollo

Se presentan inconsistencias al momento de heredar anotaciones en el proceso de segregación porque cuando es seleccionada una anotación para heredar la cual tiene diferentes valores en su id\_anotacion y el orden ver (sir\_ne\_anotacion campos id\_anotacion y ntcn\_orden) la anotación que es heredada no corresponde con la seleccionada para el proceso.

### Ejemplo

En el folio 140-5628 al seleccionar la anotación para heredar 1263 se hereda por error la anotación 1260. se verifica en la tabla sir\_ne\_anotacion campos id\_matricula = 140-5628, id\_anotacion = 1263 y ntcn\_orden = 1260

### ANÁLISIS:

El error se presenta porque las anotaciones se muestran por número de orden (**ntcn\_orden**), al seleccionar la anotación 1260 para hacer heredada, esta anotación corresponde al id 1263 (**id\_anotacion**).

ASISTENTE SEGREGACIÓN PASO 2 - SELECCIONAR ANOTACIONES A HEREDAR									
Anotaciones del Folio									
Anotaciones									
Id	# radicación	Documento			Naturaleza jurídica			Valor	Acciones
		Tipo	Número	Fecha	Id	Naturaleza			
1251	2006-140-6-8719	V	D	ESCRITURA	2437	26/09/2006	0913	DECLARACION PARTE RESTANTE	<input type="checkbox"/>
1252	2006-140-6-8719	V	D	ESCRITURA	2437	26/09/2006	0913	DECLARACION PARTE RESTANTE	<input type="checkbox"/>
1253	2006-140-6-8762	V	D	ESCRITURA	2455	27/09/2006	0126	COMPRAVENTA PARCIAL	10,000,000 <input type="checkbox"/>
1254	2006-140-6-8762	V	D	ESCRITURA	2455	27/09/2006	0913	DECLARACION PARTE RESTANTE	<input type="checkbox"/>
1255	2006-140-6-8781	V	D	ESCRITURA	2454	27/09/2006	0126	COMPRAVENTA PARCIAL	767,000 <input type="checkbox"/>
1256	2006-140-6-8781	V	D	ESCRITURA	2454	27/09/2006	0913	DECLARACION PARTE RESTANTE	<input type="checkbox"/>
1257	2006-140-6-8784	V	D	ESCRITURA	463	21/03/2006	0126	COMPRAVENTA PARCIAL	10,000 <input type="checkbox"/>
1258	2006-140-6-8784	V	D	ESCRITURA	463	21/03/2006	0913	DECLARACION PARTE RESTANTE	<input type="checkbox"/>
1259	2006-140-6-8975	V	D	ESCRITURA	2453	27/09/2006	0126	COMPRAVENTA PARCIAL	10,000,000 <input checked="" type="checkbox"/>
1260	2006-140-6-8975	V	D	ESCRITURA	2453	27/09/2006	0913	DECLARACION PARTE RESTANTE	<input checked="" type="checkbox"/>

Se verifica la información correspondiente a la matrícula 140-5628 id\_anotacion 1263, le corresponde el orden 1260.

Hoja de Trabajo Generador de Consultas

```

1 select ID_MATRICULA, ID_ANOTACION, ntcn_orden
2 from sir.sir_ne_anotacion
3 where id_matricula = '140-5628'
4 and id_anotacion = '1263';

```

Resultado de la Consulta

Todas las Filas Recuperadas: 1 en 0,007 segundos

ID_MATRICULA	ID_ANOTACION	NTCN_ORDEN
140-5628	1263	1260

Al realizar el paso 2 Seleccionar anotaciones a heredar, en la tabla **SIR\_OP\_WEB\_ANOTA\_HEREDA** se registra la anotación con el id\_anotacion = 1263

Columnas Datos Restricciones Permisos Estadísticas Disparadores Dependencias Detalles Particiones Índices SQL

Ordenar... Filtrar: ID\_SOLICITUD = '2011-140-6-8185'

ID_ANOTACION	ID_MATRICULA	ID_WEB_SEGENG	ID_SOLICITUD	WBNH_FECHA_CREACION
1263	140-5628	1	2011-140-6-8185	22/09/11 00:00:00

En el procedimiento de segregación SP\_SG al realizar la búsqueda para copiar la anotación a heredar la realizan por el orden, cuando lo correcto debe ser

Por id ya que fue la información que se registró.

## CODIGO SP\_SG INSERTAR ANOTACION A HEREDAR

```

39  /*
40  Cursor y tablas para las anotaciones a heredar
41  */
42  CURSOR C_ANOTACIONES_HEREDAR(P_ID_SOLICITUD VARCHAR2, P_ID_WEB_SEGENG VARCHAR2)
43  IS SELECT ID_ANOTACION, ID_MATRICULA
44  FROM SIR_OP_WEB_ANOTA_HEREDA
45  WHERE ID_WEB_SEGENG = P_ID_WEB_SEGENG
46  AND ID_SOLICITUD = P_ID[SOLICITUD]
47  ORDER BY TO_NUMBER(ID_ANOTACION);
48
49  TYPE T3 IS TABLE OF C_ANOTACIONES_HEREDAR%ROWTYPE INDEX BY BINARY_INTEGER;
50  T_ANOTACION_HEREDAR T3;

873      INSERT INTO SIR_NE_ANOTACION_TMP
874      (ID_ANOTACION_TMP, ID_MATRICULA, ID_NATURALEZA_JURIDICA,
875       NTTM_FECHA_REGISTRO, NTTM_VALOR, ID_DOCUMENTO, ID_TIPO_ANOTACION,
876       NTTM_RADICACION, NTTM_ESPECIFICACION,
877       NTTM_ORDER, NTTM_LAST_ID_SALVEDAD,
878       ID_ESTADO_AN, NTTM_LAST_ID_CANCELACION_TMP, NTTM_ORDER_LPAD,
879       ID_USUARIO, NTTM_HEREDADO, NTTM_ID_WORKFLOW, NTTM_FECHA_RADICACION,
880       NTTM_COMENTARIO)
881      (SELECT V_CONTADOR_ID_ANOTACION, V_ID_MATRICULA_HIJA,
882       ID_NATURALEZA_JURIDICA, SYSDATE, NTCN_VALOR, ID_DOCUMENTO,
883       4, NTCN_RADICACION, NTCN_ESPECIFICACION,V_CONTADOR_ID_ANOTACION,0,
884       'V',0,V_ORDER_LPAD, V_ID_USUARIO, 1, P_ID_WORKFLOW, NTCN_FECHA_RADICACION,
885       V_COMENTARIO_ANOTA_HEREDA
886      FROM SIR_NE_ANOTACION
887      WHERE ID_MATRICULA = V_ID_MATRICULA_PADRE
888      AND NTCN_ORDER = T_ANOTACION_HEREDAR(I).ID_ANOTACION);
~~

```

Se debe modificar el paquete SP\_SG, para realizar la búsqueda por ID\_ANOTACION y no

NTCN\_ORDEN como se está realizando actualmente.

## SOLUCIÓN

Se realiza la modificación solicitada en el Analisis\_Acta\_042\_151.docx.

## SCRIPTS:

SP\_SG.sql

## 058 CERTIFICADOS DE OTROS CÍRCULOS\_MA

Bucaramanga agosto 26 de 2011

Señores

Mesa de Ayuda

SIR-SNR

Me permito adjuntar el listado de los turnos de certificados que se encuentran sin finalizar y que corresponden a otros Círculos de Registro de los años 2009, 2010 y 2011.

Lo anterior con el fin de que todos los procesos se finalicen correctamente.

Cordialmente,

Lorenzo Silva Pedraza  
Coordinador Operativo

Oficina de Registro e Instrumentos Publicos  
Carrera 20 # 34 - 01  
Bucaramanga, Colombia

----- Original Message -----

From: yamile.moreno@supernotariado.gov.co  
To: Lorenzo Silva Pedraza  
Cc: Luis Emilio Romero Mogollon ; Imelda Muñoz Mancipe ; Dora Esperanza Pachon Rodriguez  
Sent: Monday, August 29, 2011 11:49 AM  
Subject: Re: Certificados de otros circulos

Ingeniero  
Buenos días

La SNR se encuentra proyectando un requerimiento para la solución de la inconsistencia reportada en este correo ,como es descargar los turnos de certificados nacionales, , que actualmente se encuentran listados en los roles de entrega certificado en Círculos Registrales a los cuales no corresponde el turno y los turnos que se encuentran en estado entrega, de los aplicativos de certificados en Línea y Supercades, así mismo se busca la solución para que estos turnos el sistema los finalice.

En el curso de esta semana se enviara a ETB el requerimiento para que ingrese a cola de desarrollo.

Cordialmente,

EMMA YAMILE MORENO MELO

Funcionaria Dirección de Registro

----- Original Message -----

De: Lorenzo Silva Pedraza [mailto:[lorenzo.silva@supernotariado.gov.co](mailto:lorenzo.silva@supernotariado.gov.co)]

Enviado el: martes, 08 de noviembre de 2011 05:47 p.m.

Para: Soporte SIR SNR

CC: Despacho Superintendencia de Notariado y Registro; Luis Emilio Romero Mogollon; Dora Esperanza Pachon Rodriguez; Imelda Muñoz Mancipe; Elba Lucia Corredor; Yamile Moreno Melo

Asunto: Fw: Certificados de otros círculos

Recordando nuevamente lo del asunto.

Existe un total de 1407 turnos

Gracias

Cordialmente,

Lorenzo Silva Pedraza

Coordinador Operativo

#### ESPECIFICACION DEL REQUERIMIENTO ENVIADO POR LA SNR

**JUSTIFICACION:** En el aplicativo SIR Las fases de entrega de certificados se encuentran saturadas de turnos de certificados nacionales generando bloqueo y lentitud para desplegar la lista de turnos de certificados para entrega, para dar solución a esta incidencia se requieren que los turnos de certificados nacionales sean avanzados a la fase de entregado .

1)La SNR solicita que los turnos de certificados Nacionales con fecha de radicación anterior a dos días que salga este desarrollo a producción y que se encuentren en la fase de entrega de certificados, se deben avanzar a la fase entregado, esto en todos los círculos que tengan turnos de certificados nacionales en la fase entrega .Esta labor debe realizarla ETB en forma masiva .

2)El sistema debe permitir anular turnos de certificados Nacionales desde la fecha de radicación máximo hasta el día siguiente de la fecha de radicación. Superado este tiempo el sistema debe mostrar un mensaje restrictivo informando que “se venció el tiempo establecido para anular turnos de certificados nacionales“

#### Análisis/Desarrollo

En el rol ENTREGA CERTIFICADOS al consultar los turnos de certificados pendientes para entrega carga tanto los de orden nacional como los pertenecientes al círculo.

En la pantalla administrativa OTROS->Entrega Masiva de Certificados, cuando se busca los turnos por una fecha determinada por el usuario, este proceso solo carga los turnos de certificados que pertenecen al círculo y no incluye los creados en la oficina pero de orden nacional.

El problema de lentitud en la carga que se presenta en el rol ENTREGA CERTIFICADOS es debido al gran volumen de turnos de certificados de orden nacional que no han sido entregados (represados).

Proponemos: modificar la pantalla administrativa OTROS->Entrega Masiva de Certificados para que se puedan entregar también los turnos de certificados de orden nacional futuros, y además realizar un procedimiento para entregar los turnos represados.

Solicitamos: se realice la revisión del proceso y determinen la viabilidad del requerimiento.

De: Angélica María Herreño

Fecha: 10/01/2012 03:27:26 p.m.

Para: Yamile Moreno Melo; Fernando Padilla Velez; Imelda Muñoz Mancipe; Dora Esperanza Pachón Rodríguez

Asunto: Solicitud información req 058 - Certificados de otros círculos

Buenos tardes,

En conversación telefónica con la funcionaria Yamile Moreno el día 10 de enero en las horas de la mañana se expone inquietud con respecto al requerimiento 058 - Certificados de otros círculos

### **REQUERIMIENTO**

**JUSTIFICACION:** En el aplicativo SIR Las fases de entrega de certificados se encuentran saturadas de turnos de certificados nacionales generando bloqueo y lentitud para desplegar la lista de turnos de certificados para entrega, para dar solución a esta incidencia se requieren que los turnos de certificados nacionales sean avanzados a la fase de entregado .

1) La SNR solicita que los turnos de certificados Nacionales con fecha de radicación anterior a dos días que salga este desarrollo a producción y que se encuentren en la fase de entrega de certificados, se deben avanzar a la fase entregado, esto en todos los círculos que tengan turnos de certificados nacionales en la fase entrega. Esta labor debe realizarla ETB en forma masiva.

2) El sistema debe permitir anular turnos de certificados Nacionales desde la fecha de radicación máximo hasta el día siguiente de la fecha de radicación. Superado este tiempo el sistema debe mostrar un mensaje restrictivo informando que “ se venció el tiempo establecido para anular turnos de certificados nacionales ”

### **OBSERVACION ETB**

Con respecto a la justificación se entiende que el proceso de la radicación debe ser modificado: El turno de certificado nacional quede como finalizado después de su radicación, pero al quedar como finalizado el punto uno debe aplicarse a la finalización de todos los turnos de certificado nacional sin validar que su fecha de radicación sea inferior a dos días de la puesta en producción de este desarrollo y así mismo el punto dos no cumpliría con la validación que existe actualmente con respecto a los turnos finalizados.

Otro de los casos que se da han entender es que los turnos de certificados nacionales sigan el mismo proceso y solo se finalicen los que se encuentren en el momento en que se ponga en producción el desarrollo validando que estos tengan 2 días de radicación. En conclusión se seguirían quedando los turnos en la fase de entrega Certificados.

Quedamos atentos a sus observaciones.

Cordial saludo

Angelica Maria Herreño Gomez

Ingeniera de Sistemas

De: [yamile.moreno@supernotariado.gov.co](mailto:yamile.moreno@supernotariado.gov.co) ([yamile.moreno@supernotariado.gov.co](mailto:yamile.moreno@supernotariado.gov.co))

Enviado: jueves, 19 de enero de 2012 12:12:36 p.m.

Para: Angelica Maria Herreno Gomez ([anggie35@hotmail.com](mailto:anggie35@hotmail.com));[angelicamaria.h@intek.com.co](mailto:angelicamaria.h@intek.com.co)

CC: [fernando.padilla@supernotariado.gov.co](mailto:fernando.padilla@supernotariado.gov.co); [dora.pachon@supernotariado.gov.co](mailto:dora.pachon@supernotariado.gov.co)

ingeniera  
buenos días

En atención al hilo de este correo , y de acuerdo a la conversación con usted el día 18 de enero ,se aclara que la entrega de turnos de certificados nacionales debe realizarla el usuario entrega de certificados de la Orip que lo expidió.

Cordialmente,

EMMA YAMILE MORENO MELO  
Funcionario Dirección Desarrollo Registral

#### MODIFICACION SIR-APP

- Script Bases de Datos:
  - Ninguno
- Creacion De Imagenes:
  - Ninguno

Se modificaron los siguientes archivos:

<b>CLASE:</b>
<b>gov.sir.core.negocio.acciones.administracion.AnTrasladoTurno.java</b>
<b>Método:</b>
<b>private EvnRespTrasladoTurno anularTurno(EvnTrasladoTurno evento) throws EventoException</b>

#### Adición Línea 1321:

```
/*
 * @author: Diana Lora
 * @change: Mantis 0010028: Acta - Requerimiento No 058_151 - Certificados
de otros circulos.
 */
if(turno.isNacional()){
    Calendar siguienteFechaInicio = Calendar.getInstance();
    siguienteFechaInicio.setTime(turno.getFechaInicio());
    siguienteFechaInicio.add(Calendar.DAY_OF_MONTH, 1);

    Calendar tempFecha = Calendar.getInstance();
    Calendar hoy = new GregorianCalendar(tempFecha.get(Calendar.YEAR),
        tempFecha.get(Calendar.MONTH),
        tempFecha.get(Calendar.DAY_OF_MONTH));
    if(hoy.getTime().after(siguienteFechaInicio.getTime())){
        throw new HermodException("Se venció el tiempo establecido para anular
turnos de certificados nacionales");
    }
}
```

Para que los turnos de certificados Nacionales con fecha de radicación anterior a dos días que salga este desarrollo a producción y que se encuentren en la fase de entrega de certificados.

## 061 Ajustes Módulo Reparto Notarial - Resolución 10137 de noviembre de 2011\_RN

Conforme a lo dispuesto en la Resolución No.10137 de noviembre de 2011, por la cual se establece un procedimiento para el trámite del reparto notarial, se requiere efectuar los siguientes ajustes en el módulo de reparto notarial, se requiere efectuar los siguientes ajustes al módulo de reparto notarial en las ORIP'S que se operan con el aplicativos SIR.

Las ORIP's sobre las cuales se debe realizar ese ajuste son:

DEPARTAMENTO	OFICINAS DE REGISTRO DE I.I.P.P
BOLIVAR	CARTAGENA
BOYACÁ	MONIQUIRÁ
BOYACÁ	TUNJA
CALDAS	MANIZALES
CAQUETA	FLORENCIA
CASANARE	YOPAL
CAUCA	POPAYAN
CESAR	VALLEDUPAR
CHOCÓ	QUIBDÓ
CORDOBA	MONTERÍA
HUILA	NEIVA
MAGDALENA	SANTA MARTA
META	VILLAVICENCIO
NARIÑO	PASTO
NORTE DE SANTANDER	CUCUTA
QUINDIO	ARMENIA
RISARALDA	PEREIRA
SANTANDER	BUCARAMANGA
SANTANDER	MÁLAGA
SUCRE	SINCELEJO
TOLIMA	IBAGUÉ
TOLIMA	ESPINAL
VALLE	SEVILLA
CAUCA	BUENAVENTURA

1. Radicación de minutas: El sistema debe permitir radicar las minutas todos los días dentro del respectivo horario laboral de cada una de las Oficinas de Registro, entre las 8:00 am a las 12:00 m.
2. Reparto de minutas: El sistema debe permitir ejecutar el reparto de minutas todos los días dentro de la jornada laboral de cada ORIP, a las 2:30 pm y no puede superar la hora de cierre de la jornada laboral.

### Analisis/Desarrollo

- Sir\_core.jar
- sir.core.web.acciones.comun. AWTurno.java
- sir.core.negocio.acciones.comun.ANTurno.java
- generalSIR.negocio.validaciones.ValidacionesSIR.java
- Script Bases de Datos:
  - HORARIO\_NOTARIAL.sql

- SIR\_OP\_HORARIO\_NOTARIAL.sql

**ARCHIVO JAVA:**

sir.core.web.acciones.comun.AWTurno.java.

**Código Insertado Línea 484:**

```
Circulo circulo = (Circulo) request.getSession().getAttribute(WebKeys.CIRCULO);
```

**Código Modificado Línea 498:**

```
return new EvnTurno(usuario, AWTurno.FASE_INICIAL, null, proceso, estacion, rol, circulo);
```

**ARCHIVO JAVA:**

sir.core.negocio.acciones.comun.ANTurno.java.

**Código Insertado Línea 3 - 4:**

```
import co.com.iridium.generalSIR.comun.GeneralSIRException;
import co.com.iridium.generalSIR.negocio.validaciones.ValidacionesSIR;
```

**Código Insertado Línea 20:**

```
import gov.sir.core.negocio.modelo.InicioProcesos;
```

**Código Insertado Línea 53 - 54:**

```
import java.util.logging.Level;
import java.util.logging.Logger;
```

**Código Insertado Línea 336 - 349:**

```
String tipoRepartoNotarial = "";
try {
    InicioProcesos inicioProcesos =
        hermod.obtenerFaseInicial(CProceso.PROCESO_REPARTO_NOTARIAL_MINUTAS);
    if (inicioProcesos.getIdFase().equals("REP_REPARTO")) {
        tipoRepartoNotarial = "REP_REPARTO_NOTARIAL";
    }
} catch (Throwable ex) {
    Logger.getLogger(ANTurno.class.getName()).log(Level.SEVERE, null, ex);
}
ValidacionesSIR validacion = new ValidacionesSIR();
try {
    if("true".equals(validacion.HorarioNotarial(circulo.getIdCirculo(), tipoRepartoNotarial))){
        throw new ListarNoEfectuadoException("No se puede hacer reparto notarial en este horario");
    }
}
```

**Código Insertado Línea 351 - 353:**

```
} catch (GeneralSIRException ex) {
    Logger.getLogger(ANTurno.class.getName()).log(Level.SEVERE, null, ex);
}
```

**Código Insertado Línea 393 - 702:**

```
String tipoRepartoNotarial = "";
Circulo circulo = evento.getCirculo();
try {
    InicioProcesos inicioProcesos =
        hermod.obtenerFaseInicial(CProceso.PROCESO_REPARTO_NOTARIAL_MINUTAS);
    if (inicioProcesos.getIdFase().equals("REP_REPARTO")) {
        tipoRepartoNotarial = "REP_REPARTO_MINUTAS";
    }
} catch (Throwable ex) {
    Logger.getLogger(ANTurno.class.getName()).log(Level.SEVERE, null, ex);
}
```

**Código Insertado Línea 732 - 739:**

```
ValidacionesSIR validacion = new ValidacionesSIR();
try {
    if("true".equals(validacion.HorarioNotarial(circulo.getIdCirculo(), tipoRepartoNotarial))){
        throw new ListarNoEfectuadoException("No se puede hacer reparto notarial en este horario");
    }
} catch (GeneralSIRException ex) {
    Logger.getLogger(ANTurno.class.getName()).log(Level.SEVERE, null, ex);
}
```

**ARCHIVO JAVA:**

generalSIR.negocio.validaciones.ValidacionesSIR.java.

**Código Insertado Línea 156 - 172:**

```
/*
 * Metodo que consulta si el horario es habil para radicar minutias en reparto notarial.
 * @param String idCirculo.
 * @return String
```

```

 * @author Ellery David Robles Gómez
 * Caso Mantis : 010258
 */
public String HorarioNotarial(String idCirculo, String tipoRepartoNotarial) throws GeneralSIREException{
    ValidacionesSIRDAO validacionesSIRDAO = new ValidacionesSIRDAO();
    try{
        return validacionesSIRDAO.HorarioNoHabil_RadicarRepartoNotarial(idCirculo, tipoRepartoNotarial);
    } catch (SQLException ex){
        throw new GeneralSIREException("Ocurrio un error al tratar "
            + " de consultar el horario de Reporto Notarial para el circulo " + idCirculo.toString() + "." +
            ex.getMessage(),ex);
    }
}

```

### ARCHIVO JAVA:

generalSIR.dao.ValidacionesSIRDAO.java

#### Código Modificado Línea 330 - 352:

```

 /**
 * Metodo que retorna el horario de reparto notarial.
 * @param String idCirculo.
 * @return String horarioNoHabil.
 * @author Ellery David Robles Gómez.
 * Caso Mantis : 07275.
 */
public String HorarioNoHabil_RadicarRepartoNotarial(String idCirculo, String tipoRepartoNotarial) throws SQLException
{
    String fechaSistema = "";
    String diaSistema = "";
    String horarioNoHabil = "false";

    PreparedStatement ps = null;
    ResultSet rs = null;
    StringBuilder sb = new StringBuilder();
    sb.append("SELECT TO_CHAR(SYSDATE,'DD-MM-YYYY HH24:MI') AS FECHASISTEMA, TO_CHAR(SYSDATE,'DAY') AS DIASISTEMA FROM DUAL");
    ps = conexionBaseDatos.prepareStatement(sb.toString());
    rs = ps.executeQuery();

    if (rs.next()) {
        fechaSistema = rs.getString("FECHASISTEMA");
        diaSistema = rs.getString("DIASISTEMA").trim();
    }
}

```

#### Código Modificado Línea 354 - 384:

```

 /**
 * @author : Ellery David Robles G.
 * @casoMantis : 08523.
 * @actaReq : 025_151 - Error al Desasociar el Folio.
 * @change : Indica si el folio tiene información temporal.
 * @param : turno.
 *          * @param : folio.
 *          * @param : pm.
 *          * @return : rta.
 */
private boolean hasDatosTemporales2(TurnoEnhanced turno, FolioEnhanced folio, PersistenceManager pm){
    boolean rta = false;
    //Si el folio NO es definitivo y es creado por el turno la respuesta es afirmativa
    if(!folio.isDefinitivo()){
        rta = true;
    }
    //Se revisa si tiene anotaciones temporales
    if(!rta){
        if(!folio.getAnotacionesTMPs().isEmpty()){
            List anotaList = folio.getAnotacionesTMPs();
            Iterator anotaItem = anotaList.iterator();
            if(anotaItem.hasNext()){
                AnotacionTMP anotaTMP = (AnotacionTMP)anotaItem.next();
                if(turno.getIdWorkflow().equals(anotaTMP.getNumRadicacion())){
                    rta = true;
                }
            }
        }
    }
}

```

```
    }
    return rta;
}
```

**Cambios realizados de acuerdo a la solicitud hecha en la nota.**

ARCHIVO JAVA:

```
sir.core.web.acciones.comun.AWTurno.java.
```

**Código Insertado Línea 484:**

```
Circulo circulo = (Circulo) request.getSession().getAttribute(WebKeys.CIRCULO);
```

**Código Modificado Línea 498:**

```
return new EvnTurno(usuario, AWTurno.FASE_INICIAL, null, proceso, estacion, rol, circulo);
```

ARCHIVO JAVA:

```
sir.core.negocio.acciones.comun.ANTurno.java.
```

**Código Insertado Línea 3 - 4:**

```
import co.com.iridium.generalSIR.comun.GeneralSIRException;
import co.com.iridium.generalSIR.negocio.validaciones.ValidacionesSIR;
```

**Código Insertado Línea 20:**

```
import gov.sir.core.negocio.modelo.InicioProcesos;
```

**Código Insertado Línea 53 - 54:**

```
import java.util.logging.Level;
import java.util.logging.Logger;
```

**Código Insertado Línea 336 - 350:**

```
String tipoRepartoNotarial = "";
if (fase.getId().equals(CFase.REP_REPARTO)) {
    tipoRepartoNotarial = "REP_REPARTO_NOTARIAL";
}
if ("REP_REPARTO_NOTARIAL".equals(tipoRepartoNotarial)) {
    ValidacionesSIR validacion = new ValidacionesSIR();
    try {
        if("true".equals(validacion.HorarioNotarial(circulo.getIdCirculo(), tipoRepartoNotarial))){
            throw new ListarNoEfectuadoException("No se puede hacer reparto notarial en este horario");
        }
    } catch (GeneralSIRException ex) {
        Logger.getLogger(ANTurno.class.getName()).log(Level.SEVERE, null, ex);
    }
}
```

**Código Insertado Línea 689 - 698:**

```
String tipoRepartoNotarial = "";
Circulo circulo = evento.getCirculo();
try {
    fase = hermod.getFase(CFase.REP_REPARTO);
    if (fase.getId().equals("REP_REPARTO")) {
        tipoRepartoNotarial = "REP_REPARTO_MINUTAS";
    }
} catch (Throwable ex) {
    Logger.getLogger(ANTurno.class.getName()).log(Level.SEVERE, null, ex);
}
```

**Código Insertado Línea 727 - 736:**

```
if ("REP_REPARTO_MINUTAS".equals(tipoRepartoNotarial)) {
    ValidacionesSIR validacion = new ValidacionesSIR();
    try {
        if("true".equals(validacion.HorarioNotarial(circulo.getIdCirculo(), tipoRepartoNotarial))){
            throw new ListarNoEfectuadoException("No se puede radicar minutos de reparto notarial en este horario");
        }
    } catch (GeneralSIRException ex) {
        Logger.getLogger(ANTurno.class.getName()).log(Level.SEVERE, null, ex);
    }
}
```

ARCHIVO JAVA:

```
generalSIR.negocio.validaciones.ValidacionesSIR.java.
```

**Código Insertado Línea 156 - 172:**

```
/**
 * Método que consulta si el horario es habil para radicar minutos en reparto notarial.
 * @param String idCirculo.
 * @return String
 * @author Ellery David Robles Gómez
```

```

* Caso Mantis : 010258
*/
public String HorarioNotarial(String idCirculo, String tipoRepartoNotarial) throws GeneralSIREException{
    ValidacionesSIRDAO validacionesSIRDAO = new ValidacionesSIRDAO();
    try{
        return validacionesSIRDAO.HorarioNoHabil_RadicarRepartoNotarial(idCirculo, tipoRepartoNotarial);
    } catch (SQLException ex){
        throw new GeneralSIREException("Ocurrio un error al tratar "
            + " de consultar el horario de Reporto Notarial para el circulo " + idCirculo.toString() + "." +
            ex.getMessage(),ex);
    }
}

```

**ARCHIVO JAVA:**  
**generalsIR.dao.ValidacionesSIRDAO.java**

**Código Modificado Línea 330 - 484:**

```

/**
 * Metodo que retorna el horario de reparto notarial.
 * @param String idCirculo.
 * @return String horarioNoHabil.
 * @author Ellery David Robles Gómez.
 * Caso Mantis : 07275.
 */
public String HorarioNoHabil_RadicarRepartoNotarial(String idCirculo, String tipoRepartoNotarial) throws SQLException
{
    String fechaSistema = "";
    String diaSistema = "";
    String horarioNoHabil = "false";

    PreparedStatement ps = null;
    ResultSet rs = null;
    StringBuilder sb = new StringBuilder();
    sb.append("SELECT TO_CHAR(SYSDATE,'DD-MM-YYYY HH24:MI') AS FECHASISTEMA, TO_CHAR(SYSDATE,'DAY') AS DIASISTEMA FROM DUAL");
    ps = conexionBaseDatos.prepareStatement(sb.toString());
    rs = ps.executeQuery();

    if (rs.next()) {
        fechaSistema = rs.getString("FECHASISTEMA");
        diaSistema = rs.getString("DIASISTEMA").trim();
    }
    rs.close();
    ps.close();

    if (!fechaSistema.equals("")) {
        /* *** hallar fecha Actual de sistema *** */
        String[] fechaActual = fechaSistema.split(" ");
        String[] horaSis = fechaActual[1].split(":");
        String[] m_hora_inicio = null;
        String[] m_hora_fin = null;
        String[] t_hora_inicio = null;
        String[] t_hora_fin = null;

        ps = null;
        rs = null;
        StringBuilder sb2 = new StringBuilder();
        sb2.append("SELECT M_HORA_INICIO, M_HORA_FIN, T_HORA_INICIO, T_HORA_FIN "
            + "FROM SIR.SIR_OP_HORARIO_NOTARIAL "
            + "WHERE ID_CIRCULO = ? AND HRREP_DIA = ? AND TIPO_REP_NOTARIAL = ?");
        ps = conexionBaseDatos.prepareStatement(sb2.toString());
        ps.setString(1, idCirculo);
        ps.setString(2, diaSistema);
        ps.setString(3, tipoRepartoNotarial);
        rs = ps.executeQuery();
        if (rs.next()) {
            /* Validando Radicaciones por la Mañana */
            if (rs.getString("M_HORA_INICIO") != null && rs.getString("M_HORA_FIN") != null &&
                rs.getString("T_HORA_INICIO") == null && rs.getString("T_HORA_FIN") == null) {
                m_hora_inicio = rs.getString("M_HORA_INICIO").split(":");
                m_hora_fin = rs.getString("M_HORA_FIN").split(":");
            }
        }
    }
}

```

```

        if (Integer.parseInt(horaSis[0]) == Integer.parseInt(m_hora_inicio[0])) {
            if (Integer.parseInt(horaSis[1]) < Integer.parseInt(m_hora_inicio[1])) {
                horarioNoHabil = "true";
            }
        }
        if (Integer.parseInt(horaSis[0]) == Integer.parseInt(m_hora_fin[0])) {
            if (Integer.parseInt(horaSis[1]) > Integer.parseInt(m_hora_fin[1])) {
                horarioNoHabil = "true";
            }
        }
        if (Integer.parseInt(horaSis[0]) < Integer.parseInt(m_hora_inicio[0])) {
            horarioNoHabil = "true";
        }
        if (Integer.parseInt(horaSis[0]) > Integer.parseInt(m_hora_fin[0])) {
            horarioNoHabil = "true";
        }
    }

    /* Validando Radicaciones por la Tarde */
    if (rs.getString("T_HORA_INICIO") != null && rs.getString("T_HORA_FIN") != null &&
    rs.getString("M_HORA_INICIO") == null && rs.getString("M_HORA_FIN") == null) {
        t_hora_inicio = rs.getString("T_HORA_INICIO").split(":");
        t_hora_fin = rs.getString("T_HORA_FIN").split ":";

        if (Integer.parseInt(horaSis[0]) == Integer.parseInt(t_hora_inicio[0])) {
            if (Integer.parseInt(horaSis[1]) < Integer.parseInt(t_hora_inicio[1])) {
                horarioNoHabil = "true";
            }
        }

        if (Integer.parseInt(horaSis[0]) == Integer.parseInt(t_hora_fin[0])) {
            if (Integer.parseInt(horaSis[1]) > Integer.parseInt(t_hora_fin[1])) {
                horarioNoHabil = "true";
            }
        }

        if (Integer.parseInt(horaSis[0]) < Integer.parseInt(t_hora_inicio[0])) {
            horarioNoHabil = "true";
        }

        if (Integer.parseInt(horaSis[0]) > Integer.parseInt(t_hora_fin[0])) {
            horarioNoHabil = "true";
        }
    }

    /* Validando Radicaciones por la Mañana y la Tarde */
    if (rs.getString("M_HORA_INICIO") != null && rs.getString("M_HORA_FIN") != null &&
    rs.getString("T_HORA_INICIO") != null && rs.getString("T_HORA_FIN") != null) {
        m_hora_inicio = rs.getString("M_HORA_INICIO").split(":");
        m_hora_fin = rs.getString("M_HORA_FIN").split(":");
        t_hora_inicio = rs.getString("T_HORA_INICIO").split(":");
        t_hora_fin = rs.getString("T_HORA_FIN").split ":";

        /* ----- */
        if (Integer.parseInt(horaSis[0]) == Integer.parseInt(m_hora_inicio[0])) {
            if (Integer.parseInt(horaSis[1]) < Integer.parseInt(m_hora_inicio[1])) {
                horarioNoHabil = "true";
            }
        }

        if (Integer.parseInt(horaSis[0]) == Integer.parseInt(m_hora_fin[0])) {
            if (Integer.parseInt(horaSis[1]) > Integer.parseInt(m_hora_fin[1])) {
                horarioNoHabil = "true";
            }
        }

        /* ----- */
        if (Integer.parseInt(horaSis[0]) == Integer.parseInt(t_hora_inicio[0])) {
    
```

```

        if (Integer.parseInt(horaSis[1]) < Integer.parseInt(t_hora_inicio[1])) {
            horarioNoHabil = "true";
        }
    }
    if (Integer.parseInt(horaSis[0]) == Integer.parseInt(t_hora_fin[0])) {
        if (Integer.parseInt(horaSis[1]) > Integer.parseInt(t_hora_fin[1])) {
            horarioNoHabil = "true";
        }
    }

    /* -----Inicio de la Mañana----- */
    if (Integer.parseInt(horaSis[0]) < Integer.parseInt(m_hora_inicio[0])) {
        horarioNoHabil = "true";
    }

    /* -----Fin de la Tarde----- */
    if (Integer.parseInt(horaSis[0]) > Integer.parseInt(t_hora_fin[0])) {
        horarioNoHabil = "true";
    }

    /* -----Tiempo intermedio Mañana y Tarde----- */
    if ((Integer.parseInt(horaSis[0]) > Integer.parseInt(m_hora_fin[0])) && (Integer.parseInt(horaSis[0]) <
        Integer.parseInt(t_hora_inicio[0]))) {
        horarioNoHabil = "true";
    }
}
rs.close();
ps.close();
}
return horarioNoHabil;
}
}

```

## 062 PRUEBAS DE FUNCIONALIDAD DE CONEXIÓN Y MENSAJERÍA ENTRE SGD Y SIR

De: Luz Edith Florián Sánchez [mailto:luz.florian@supernotariado.gov.co]

Enviado el: martes, 13 de diciembre de 2011 12:19 p.m.

Para: Luis Emilio Romero Mogollón

Asunto: PRUEBAS DE CONEXIÓN Y MENSAJERIA ENTRE SGD y SIR

Ingeniero Luis Emilio

Muy buenos días, de manera atenta me permito recordar que para la implementación de las ORIP con sgd, se deben hacer pruebas de funcionalidad DE CONEXIÓN Y MENSAJERIA ENTRE SGD y SIR de acuerdo al cronograma estas pruebas se deben realizar en el mes de enero y febrero.

De nuevo envío las ORIP a implementar de las cuales hay 12 que funcionan con SIR

SIR	Carmen de Bolívar
SIR	Santa Marta
SIR	Plato
SIR	Pto López
SIR	Sincelejo
SIR	Valledupar
SIR	Tunja
SIR	Villavicencio
SIR	Pereira
SIR	Manizales
FOLIO	Calarca
SIR	Ibagué
SIR	Armenia

	FOLIO	Yarumal	
Luz Edith Florián			
Carmen de Bolívar	062	Villavicencio	230
Tunja	070	Puerto Lopez	234
Santa Marta	080	Armenia	280
Manizales	100	Pereira	290
Valledupar	190	Sincelejo	340
Plato	226	Ibagué	350
Se modifica archivo gov.sir.hermod.properties en la librería mapas.jar			
<b>ANTES</b>			
#Círculos con Mensajería gov.sir.hermod.SGD.círculos=060,300,260			
<b>POR</b>			
#Círculos que envían mensajería gov.sir.hermod.SGD.círculos=060,062,070,080,100,190,226,230,234,260,280,290,300,340,350			
También en el sistema de gestión documental de pruebas se crearon los usuarios para las ORIPS, para la creación de dichos usuarios es necesario realizar la intervención de las siguientes tablas en la base de datos “jboss_9”:			
JMS_ROLES JMS_SUBSCRIPTIONS JMS_USERS			
La base de datos se encuentra en el servidor 192.168.202.9.			
Se crearon los siguientes archivos para poblar las tablas con la información correspondiente a las ORIPS. JMS_ROLES.csv JMS_SUBSCRIPTIONS.csv JMS_USERS.csv			
Después de poblar las tablas se procede a reiniciar el servicio de JBOSS donde se encuentra Gestión Documental.			
<b>063 AJUSTES REPARTO NOTARIAL</b>			
Conforme a lo dispuesto en la Resolución No. 10137 de noviembre de 2011, por la cual se establece un procedimiento para el trámite del Reparto Notarial, se aclaran los ajustes a realizar en el módulo de Reparto Notarial en las ORIP's que operan con el aplicativo SIR.			
<ol style="list-style-type: none"> <li>1. Radicación de minutos: el sistema de permitir radicar las minutos todos los días dentro del respectivo horario laboral de cada una de las oficinas de registro, entre las 8:00 am hasta las 12:00 am.</li> </ol> <p>Esta restricción debe aplicarse solo para la radicación de minutos de reparto ordinario; sin embargo, como el sistema SIR permite seleccionar si la minuta a radicar corresponde al tipo de reparto ORDINARIO o EXTRAORDINARIO, en el caso de superar la hora estipulada para radicar, solo debe habilitar el botón para radicar minutos de reparto EXTRAORDINARIO.</p> <ol style="list-style-type: none"> <li>2. Categorización de las minutos radicadas: Si la solicitud contiene varios actos o negocios jurídicos, el sistema debe permitir marcar o seleccionar el acto o la cuantía que determinará la categorización de la minuta.</li> </ol>			

Es importante, que se verifique si las categorías indicadas en el Artículo 13 de la presente Resolución y sus párrafos, están parametrizadas y actualizadas en el SIR.

3. Recibo Minuta: Se debe ajustar el reporte de recibo de radicación de minutas en los siguientes ítems:
  - Cambiar título “SOLICITUD DE RADICACIÓN DE MINUTAS” por “RADICACIÓN SOLICITUD REPARTO NOTARIAL”.
  - Incluir el número de recibo.
  - En la categoría el valor o cuantía, deber ser el marcado o seleccionado al momento de ingresar los actos o negocios jurídicos que determinará la categorización de la minuta.
  - Incluir por cada acto o contrato relacionado, el valor o cuantía. Si no contiene, dejar en blanco.
4. Reparto de minutas: El sistema debe permitir ejecutar el reparto de minutas todos los días dentro de la jornada laboral de cada ORIP, a las 2:30 pm y no puede superar la hora de cierre de la jornada laboral. Esta restricción debe aplicarse solo para las minutas que hayan sido radicadas para reparto ordinario.

Para el caso de efectuar reparto extraordinario, se debe indicar el motivo o justificación y no aplicar la restricción solicitada.

Este Caso tiene como referencia el acta de requerimiento No 061. Ajustes Módulo Reparto Notarial - Resolución 10137 de noviembre de 2011\_RN.

- Script Bases de Datos:
  - Ninguno
- Creacion De Imagenes:
  - Ninguno
- Se modificaron los siguientes archivos:
  - sir\_core.jar
  - sir\_hermod.jar
  - EnvRepartoNotarial
  - ControlNavegacionRepartoNotarial AWturno.

**CLASE:**

**WEB-INF/jspSeguro/repartonotarial/SOLICITUD\_REPARTO\_NOTARIAL.jsp**

**Método JavaScript:**

**function actualizarCuantia(radio, cuantia, unidades)**

**Adición Línea 102:**

```
/*
 * @author: Diana Lora
 * @change: Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 */
function actualizarCuantia(radio, cuantia, unidades, idAccionNotarial){
  if(radio.checked == true){
    document.getElementById('<%=CMinuta.CUANTIA_SELECCIONADA%>').value = cuantia;
    document.getElementById('<%=CMinuta.UNIDAD_SELECCIONADA%>').value = unidades;
    document.getElementById('<%=CMinuta.ID_ACCION_NOTARIAL_SELECCIONADA%>').value = idAccionNotarial;
  }
}
```

**Adición Línea 250:**

```
<%
  String repartoMinutaOrdinario =
  (String)request.getSession().getAttribute(WebKeys.REPARTO_MINUTAS_EN_HORARIO);
  if(repartoMinutaOrdinario.equals("true")){
  %>
```

**Adición Línea 254:**

```
<%
```

```

}else{
    %>
    <td width="20%" align='right'>Ordinaria</td>
    <%
}
%>

```

**Adición Línea 258:**

```
if(repartoMinutaOrdinario.equals("true")){
```

**Adición Línea 267:**

```

}else{
    request.getSession().setAttribute(CMinuta.TIPO_REPARTO,CMinuta.EXTRAORDINARIO);
}
```

**Adición Línea 272:**

```

if(repartoMinutaOrdinario.equals("true")){
    radioHelper.setValordefecto(CMinuta.ORDINARIO);
    radioHelper.setNombre(CMinuta.TIPO_REPARTO);
        radioHelper.setId(CMinuta.TIPO_REPARTO);
    radioHelper.render(request,out);
}
```

**Adición Línea 418:**

```
String idAccionNotarialSeleccionada = null, cuantiaSeleccionada = null, unidadSeleccionada = null;
```

**Adición Línea 420:**

```

idAccionNotarialSeleccionada = (String)session.getAttribute(CMinuta.ID_ACCION_NOTARIAL_SELECCIONADA);
cuantiaSeleccionada = (String)session.getAttribute(CMinuta.CUANTIA_SELECCIONADA);
unidadSeleccionada = (String)session.getAttribute(CMinuta.UNIDAD_SELECCIONADA);
```

**Adición Línea 427:**

```
<td></td>
```

**Adición Línea 434:**

```

<input type="hidden" id="<%="CMinuta.CUANTIA_SELECCIONADA%">" name="<%="CMinuta.CUANTIA_SELECCIONADA%">">
value="<%=">
    if(cuantiaSeleccionada != null && cuantiaSeleccionada.trim().length() > 0){
        out.print(cuantiaSeleccionada);
    }else if(cuantiaSeleccionada == null){
        out.print("");
    }
    %>"/>
<input type="hidden" id="<%="CMinuta.UNIDAD_SELECCIONADA%">" name="<%="CMinuta.UNIDAD_SELECCIONADA%"> value="<%=">
    if(unidadSeleccionada != null && unidadSeleccionada.trim().length() > 0){
        out.print(unidadSeleccionada);
    }else if(unidadSeleccionada == null){
        out.print("");
    }
    %>"/>
<input type="hidden" id="<%="CMinuta.ID_ACCION_NOTARIAL_SELECCIONADA%">" name="<%="CMinuta.ID_ACCION_NOTARIAL_SELECCIONADA%"> value="<%=">
    if(idAccionNotarialSeleccionada != null && idAccionNotarialSeleccionada.trim().length() > 0){
        out.print(idAccionNotarialSeleccionada);
    }else if(idAccionNotarialSeleccionada == null){
        out.print("");
    }
    %>"/>
```

**Adición Línea 475:**

```

<td>
    <%
        if(idAccionNotarialSeleccionada != null &&
idAccionNotarialSeleccionada.equals(accionNotarialLista.getIdAccionNotarial())){
            %>
            <input type="radio" id="radioCuantia<%=a%>" name="radioCuantia" checked="checked"
onclick="actualizarCuantia(this, '<%=""'+valor%>', '<%=""'+unidades%>', '<%=accionNotarialLista.getIdAccionNotarial()%>')"
value="<%="accionNotarialLista.getIdAccionNotarial()%>"/>
    <%
```

<pre>         }else{         %&gt;         &lt;input type="radio" id="radioCuantia&lt;%=a%&gt;" name="radioCuantia" onclick="actualizarCuantia(this, '&lt;%=""+valor%&gt;', '&lt;%=""+unidades%&gt;', '&lt;%=accionNotarialLista.getIdAccionNotarial()%&gt;')" /&gt;         &lt;%         }         %&gt;     &lt;/td&gt; </pre>	<b>CLASE:</b> <b>gov.sir.core.web.acciones.comun.AWTurno.java</b>
<b>Adición Línea 597:</b>	
<pre>     /**      * @author : Diana Lora      * @change : Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial      */     String repartoMinutas = (String)tabla.get(WebKeys.REPARTO_MINUTAS_EN_HORARIO);     if(repartoMinutas != null){         request.getSession().setAttribute(WebKeys.REPARTO_MINUTAS_EN_HORARIO, repartoMinutas);     } </pre>	<b>CLASE:</b> <b>gov.sir.core.web.WebKeys.java</b>
<b>Adición Línea 1250:</b>	
<pre>     /**      * @author : Diana Lora      * @Change: Nueva constante para controlar el horario del reparto notarial de minutos.      * Caso Mantis: 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial      */     public static final String REPARTO_MINUTAS_EN_HORARIO = "REPARTO_MINUTAS_EN_HORARIO"; </pre>	<b>CLASE:</b> <b>gov.sir.core.negocio.modelo.constantes.CMinuta.java</b>
<b>Adición Línea 109:</b>	
<pre>     /**      * @author: Diana Lora      * @change: Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial      * Categorización de las minutos radicadas: Si la solicitud contiene varios      * actos o negocios jurídicos, el sistema debe permitir marcar o seleccionar el      * acto o la cuantía que determinará la categorización de la minuta.      */     public final static String CUANTIA_SELECCIONADA = "CUANTIA_SELECCIONADA";     public final static String UNIDAD_SELECCIONADA = "UNIDAD_SELECCIONADA";     public final static String ID_ACCION_NOTARIAL_SELECCIONADA = "ID_ACCION_NOTARIAL_SELECCIONADA"; </pre>	<b>CLASE:</b> <b>gov.sir.core.negocio.acciones.comun.ANTurno.java</b>
	<b>Método:</b> <b>private EvnRespTurno buscarNotasInicio(EvnTurno evento) throws EventoException</b>
<b>Original Línea 730:</b>	
<pre> if("true".equals(validacion.HorarioNotarial(circulo.getIdCirculo(), tipoRepartoNotarial))){     throw new ListarNoEfectuadoException("No se puede radicar minutos de reparto notarial en este horario"); } </pre>	
<b>Cambio Línea 730:</b>	
<pre>     /**      * @author : Diana Lora      * @change : Mantis 0010028: Acta - Requerimiento No 058_151 - Certificados de otros circulos      */     if("true".equals(validacion.HorarioNotarial(circulo.getIdCirculo(), tipoRepartoNotarial))){         throw new ListarNoEfectuadoException("No se puede radicar minutos de reparto notarial en este horario");         tabla.put(WebKeys.REPARTO_MINUTAS_EN_HORARIO, "false");     }else{         tabla.put(WebKeys.REPARTO_MINUTAS_EN_HORARIO, "true");     } </pre>	<b>CLASE:</b> <b>gov.sir.core.web.acciones.repartonotarial.AWLiquidacionRepartoNotarial.java</b>
	<b>Método:</b> <b>private EvnRepartoNotarial obtenerCategoria(HttpServletRequest request) throws AccionWebException</b>
<b>Cambio Línea 716:</b>	

```

//          double cuantiaTotal = 0;
//          long unidadesTotal = 0;

Cambio Línea 724:
//          cuantiaTotal = cuantiaTotal + actoMinutaTmp.getValor();
//          unidadesTotal = unidadesTotal + actoMinutaTmp.getUnidades();

Original Línea 730:
/*
 * @author : Diana Lora
 * @change : Mantis 0010028: Acta - Requerimiento No 058_151 - Certificados de otros circulos
 * Se coloca el valor de las unidades y cuantias seleccionada por el usuario.
 */
minuta.setUnidades((long) unidades);
minuta.setValor(cuantia);

Adición Línea 677:
/*
 * @author : Diana Lora
 * @change : Mantis 0010028: Acta - Requerimiento No 058_151 - Certificados de otros circulos
 */
double unidades = (Double)request.getSession().getAttribute(CMinuta.UNIDAD_SELECCIONADA);
double cuantia = (Double)request.getSession().getAttribute(CMinuta.CUANTIA_SELECCIONADA);

```

**Método:**

<b>private EvnRepartoNotarial validarInformacion(HttpServletRequest request) throws AccionWebException</b>
--

```

Cambio Línea 782:
//          String cuantia = request.getParameter(CMinuta.CUANTIA);
//          String cuantiaOriginal = request.getParameter(CMinuta.CUANTIA_SELECCIONADA);
//          String cuantia = cuantiaOriginal.replace(".", "");
//          String unidades = request.getParameter(CMinuta.UNIDADES);
//          String unidades = request.getParameter(CMinuta.UNIDAD_SELECCIONADA);

Adición Línea 854:
request.getSession().setAttribute(CMinuta.UNIDAD_SELECCIONADA, unidades);
Adición Línea 859:
else{    request.getSession().setAttribute(CMinuta.UNIDAD_SELECCIONADA,
"0");
}

Cambio Línea 865:
exception.addError("Debe seleccionar el valor de la cuantía");

Cambio Línea 870:
request.getSession().setAttribute(CMinuta.CUANTIA_SELECCIONADA, Double.parseDouble(cuantia));

Cambio Línea 875:
if(!exception.getErrores().isEmpty()){
    throw exception;
}
```

**Método:**

<b>private void preservarInfoLiquidacion(HttpServletRequest request)</b>
--

```

Agregar Línea 895:
sesion.setAttribute(CMinuta.ID_ACCION_NOTARIAL_SELECCIONADA,
request.getParameter(CMinuta.ID_ACCION_NOTARIAL_SELECCIONADA));
    sesion.setAttribute(CMinuta.CUANTIA_SELECCIONADA,
request.getParameter(CMinuta.CUANTIA_SELECCIONADA));
    sesion.setAttribute(CMinuta.UNIDAD_SELECCIONADA, request.getParameter(CMinuta.UNIDAD_SELECCIONADA));
```

**CLASE:**

<b>gov.sir.hermiod.dao.impl.jdogenie.JDORreportosDAO.java</b>
---

**Método:**

<b>public Categoria getCategoríaClasificacionMinuta(Minuta minuta) throws DAOException</b>
--

**Cambio Línea 644:**

```

//          double valorMaximo = 0;
//          double unidadesMaximo = 0;
```

**Cambio Línea 679:**

```

//          valorMaximo = valorMaximo + minutaAccionNotarial.getValor();
//          unidadesMaximo = unidadesMaximo + minutaAccionNotarial.getUnidades();
```

**Adición Línea 683:**

	<pre>        double valorMaximo = minuta.getValor();         double unidadesMaximo = minuta.getUnidades();</pre>
<b>RECIBO MINUTAS:</b>	
	<b>CLASE:</b>
	<b>gov.sir.core.negocio.acciones.repartonotarial.ANRepartoNotarial.java</b>
	<b>Método:</b>
	<b>private EvnRespRepartoNotarial crearSolicitud(EvnRepartoNotarial evento) throws EventoException</b>
<b>Cambio Línea 393:</b>	
	<pre>        long reciboMinutaRepartoNotarial = hermod.getSecuencialRepartoNotarial(turno.getIdCirculo(), turno.getAnio());         ImprimibleRecibo impRec = new ImprimibleRecibo(pago, circulo, fechaImpresion, CTipoImprimible.RECIBO,         reciboMinutaRepartoNotarial);</pre>
	<b>CLASE:</b>
	<b>gov.sir.core.negocio.modelo.imprimibles.ImprimibleRecibo.java</b>
<b>Adición Línea 166:</b>	
	<pre>private List PagoAsociado = null; /*  * @author : Diana Lora  * @change : Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial  * Incluir el número de recibo.  */ private long reciboMinutaRepartoNotarial = -1;  public long getReciboMinutaRepartoNotarial() {     return reciboMinutaRepartoNotarial; } public void setReciboMinutaRepartoNotarial(long reciboMinutaRepartoNotarial) {     this.reciboMinutaRepartoNotarial = reciboMinutaRepartoNotarial; } public List getPagoAsociado() {     return PagoAsociado; }  public void setPagoAsociado(List PagoAsociado) {     this.PagoAsociado = PagoAsociado; }</pre>
<b>Nuevo Contructor:</b>	
/**	
	<pre>* @author: Diana Lora * @change: 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial */ public ImprimibleRecibo(Pago pago, Circulo circulo, String fechaImpresion, String tipoImprimible, long reciboMinutaRepartoNotarial) {     super(tipoImprimible);     // prevenir, cuando se edite y llame a otro constructor     AppletLogger loggerImpl = AppletLoggerImpl1.getAppletLogger();     if (null == this.logger) {         this.logger = loggerImpl;     }     setTransferObject(pago);     this.pago = pago; this.reciboMinutaRepartoNotarial = reciboMinutaRepartoNotarial; */ * @author : Julio Alcázar Rivas * @change : asigna al documento la solicitud al imprimible * Caso Mantis : 000941 * List liquidaciones = pago.getLiquidacion().getSolicitud().getLiquidaciones(); if(!liquidaciones.isEmpty()){     Liquidacion liquidacion = (Liquidacion) liquidaciones.get(liquidaciones.size() - 1);     if (pago.getLiquidacion().getSolicitud() instanceof SolicitudCertificadoMasivo         &amp;&amp; (((LiquidacionTurnoCertificadoMasivo)liquidacion).getTipoTarifa().equals(CTipoTarifa.EXENTO))){         this.documento = ((SolicitudCertificadoMasivo) pago.getLiquidacion().getSolicitud()).getDocumento();     } } this.circulo = circulo; this.fechaImpresion = fechaImpresion;</pre>
	//if( pago != null ) {

```

Solicitud tempSolicitud = pago.getLiquidacion().getSolicitud();
Turno tempSolicitudTurno = tempSolicitud.getTurno();
if (tempSolicitudTurno != null) {
    this.solicitud = tempSolicitud.getTurno().getSolicitud();
}
// }

if (this.solicitud == null) {
    this.solicitud = pago.getLiquidacion().getSolicitud();
}
//no pintar el margen en los recibos
this.setImprimirMargen(false);
}

Método:
private void imprimirCuerpoReciboRepartoNotarial()

Cambio Línea 3446:
/*
 * @author: Diana Lora
 * @change: 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 */
this.imprimirLinea(ImprimibleConstantes.PLANO, "CONTRATO ("+(a+1)+") : " + an.getNombre() + " - CUANTÍA O VALOR : " +
NumberFormat.getInstance().format(man.getValor()));
Método:
protected void makeNewPage()

Agregar Línea 1819:
/*
 * @author: Diana Lora
 * @change: 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 * Incluir el número de recibo.
 */
this.imprimirLinea(ImprimibleConstantes.TITULO_GRANDE,
(int)(ImprimibleConstantes.MARGEN_IZQ * 10.0d),
""+reciboMinutaRepartoNotarial, true);
Método:
protected String getTitulo()

Cambio Línea 1906:
/*
 * @author: Diana Lora
 * @change: 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 * Cambiar título por "RADICACIÓN SOLICITUD REPARTO NOTARIAL".
 */
titulo = "RADICACIÓN SOLICITUD REPARTO NOTARIAL";
CLASE:
gov.sir.hermode.HermodeException.java

Cambio Línea 1899:
/*
 * @author: Diana Lora
 * @Change: Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 */
public static final String SECUENCIAL_REPARTO_NOTARIAL_NO_OBTENIDO = "No fue posible obtener el secuencial de
recibo correspondiente al proceso de reparto notarial.";
CLASE:
gov.sir.hermode.HermodeService.java
Nuevo Método:
public long getSecuencialRepartoNotarial(String circuloId, String year) throws HermodeException

Línea 7838:
/*
 * @author : Diana Lora
 * @change : Obtiene la secuencia de los recibos de reparto notarial, de acuerdo
 * a los parametros recibidos.
 * @Caso Mantis : 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 * @param year El año en el que se va a expedir el recibo de reparto notarial.
 * @return El secuencial requerido.
 * @throws HermodeException
 */
public long getSecuencialRepartoNotarial(String circuloId, String year) throws HermodeException {

```

```

try {
    return factory.getProcesosDAO().getSecuencialRepartoNotarial(circuloId,year);
} catch (DAOException e) {
    HermodException he = new
    HermodException(HermodException.SECUENCIAL_REPARTO_NOTARIAL_NO_OBTENIDO, e);
    ExceptionPrinter ep = new ExceptionPrinter(he.fillInStackTrace());
    Log.getInstance().error(this.getClass(), he.getMessage(), e);
    throw he;
}
}

CLASE:  

gov.sir.hermode.ejb.HermodServiceEJBBean.java  

Nuevo Método:  

public long getSecuencialRepartoNotarial(String circuloId, String year) throws HermodException  

Línea 2868:  

/*
 * @author : Diana Lora
 * @change : Obtiene la secuencia de los recibos de reparto notarial, de acuerdo
 * a los parametros recibidos.
 * @see gov.sir.forseti.interfaz.ForsetiServiceInterface#getFolioOrigenTraslado(java.lang.String matricula)
 * @ejb:interface-method view-type="both"
 * @Caso Mantis : 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 * @param year El año en el que se va a expedir el recibo de reparto notarial.
 * @return El secuencial requerido.
 * @throws HermodException
 */
public long getSecuencialRepartoNotarial(String circuloId, String year) throws HermodException {
    return hermod.getSecuencialRepartoNotarial(circuloId, year);
}  

INTERFAZ:  

gov.sir.hermode.interfaz.HermodServiceInterface.java  

Nuevo Método:  

public long getSecuencialRepartoNotarial(String circuloId, String year) throws HermodException  

Línea 4640:  

/*
 * @author : Diana Lora
 * @change : Obtiene la secuencia de los recibos de reparto notarial, de acuerdo
 * a los parametros recibidos.
 * @Caso Mantis : 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 * @param year El año en el que se va a expedir el recibo de reparto notarial.
 * @return El secuencial requerido.
 * @throws HermodException
 */
public long getSecuencialRepartoNotarial(String circuloId, String year) throws HermodException;  

INTERFAZ:  

gov.sir.hermode.dao.ProcesosDAO.java  

Nuevo Método:  

public long getSecuencialRepartoNotarial(String circuloId, String year) throws HermodException  

Línea 138:  

/*
 * @author : Diana Lora
 * @change : Obtiene la secuencia de los recibos de reparto notarial, de acuerdo
 * a los parametros recibidos.
 * @Caso Mantis : 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 * @param year El año en el que se va a expedir el recibo de reparto notarial.
 * @return El secuencial requerido.
 * @throws HermodException
 */
public long getSecuencialRepartoNotarial(String circuloId, String year) throws DAOException ;  

CLASE:  

gov.sir.hermode.dao.impl.jdogenie.JDOProcesosDAO.java  

Nuevo Método:  

public long getSecuencialRepartoNotarial(String circuloId, String year) throws HermodException  

/*
 * @author : Diana Lora
 * @change : Obtiene la secuencia de los recibos de reparto notarial, de acuerdo
```

```

* a los parametros recibidos.
* @Caso Mantis : 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
* @param year El año en el que se va a expedir el recibo de reparto notarial.
* @return El secuencial requerido.
* @throws HermodException
*/
public long getSecuencialRepartoNotarial(String circuloId, String year) throws DAOException {
    long rta = -1;
    PersistenceManager pm = null;
    VersantPersistenceManager pm2 = null;

    if (circuloId == null) {
        throw new DAOException("El identificador del círculo es inválido");
    }
    try {
        pm = AdministradorPM.getPM();
        pm.currentTransaction().begin();
        //Se hace el cambio de tipo de bloqueo pesimista para
        //que se bloquee la tabla la cual nos
        //provee el secuencial
        pm2 = (VersantPersistenceManager) pm;
        pm2.setDatastoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
        pm = pm2;

        CirculoProcesoEnhancedPk sid = new CirculoProcesoEnhancedPk();
        sid.anio = year;
        sid.idCirculo = circuloId;
        Long idProcesoString = new Long (CProceso.PROCESO_REPARTO_NOTARIAL_MINUTAS);
        sid.idProceso = idProcesoString.longValue();
        CirculoProcesoEnhanced circuloProceso = this.get CirculoProcesoByID(sid, pm);
        if (circuloProceso == null)
        {
            pm2.set DatastoreTxLocking(VersantPersistenceManager.LOCKING_NONE);
            throw new DAOException ("No se pudo obtener el secuencial para la solicitud de reparto
notarial");
        }
        circuloProceso.setLastIdRepartoNotarial(circuloProceso.getLastIdRepartoNotarial() + 1);
        //Volvemos a setear el tipo de bloqueo pesimista
        //para que no nos bloquee los siguientes registros
        //consultados
        pm2.set DatastoreTxLocking(VersantPersistenceManager.LOCKING_NONE);

        rta = circuloProceso.getLastIdRepartoNotarial();
    }

    //No se encontró el círculo Proceso.
    catch (JDOObjectNotFoundException e) {
        pm2.set DatastoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
        if (pm.currentTransaction().isActive())
        {
            pm.currentTransaction().rollback();
        }
        throw new DAOException("No se pudo obtener el secuencial para el recibo de certificados masivos");
    }
    //Excepción JDO
    catch (JDOException e) {
        pm2.set DatastoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
        if (pm.currentTransaction().isActive())
        {
            pm.currentTransaction().rollback();
        }
        throw new DAOException("No se pudo obtener el secuencial para el recibo de certificados masivos",e);
    }
    catch (DAOException e) {
        pm2.set DatastoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
        if (pm.currentTransaction().isActive())
        {
    }

```

```

                pm.currentTransaction().rollback();
            }
            throw new DAOException("No se pudo obtener el secuencial para el recibo de certificados
masivos",e);
        }
        catch (Exception e) {
            pm2.setDatastoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
            if (pm.currentTransaction().isActive())
            {
                pm.currentTransaction().rollback();
            }
            throw new DAOException("No se pudo obtener el secuencial para el recibo de certificados
masivos",e);
        }
        //Otra Excepción
        catch (Throwable t) {
            pm2.setDatastoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
            if (pm.currentTransaction().isActive())
            {
                pm.currentTransaction().rollback();
            }
            throw new DAOException("No se pudo obtener el secuencial para el recibo de certificados masivos");
        }

        //Cerrar la transacción
        finally {
            //pm2.setDatastoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
            if (pm.currentTransaction().isActive()) {
                pm.currentTransaction().commit();
            }
            pm.close();
        }
        return rta;
    }

```

#### REPARTO MINUTAS:

**CLASE:**

**org.auriga.smart.vistas.xml**

#### Adición Línea 6431:

```

<!--
 * @author : Diana Lora
 * @change : se agrega una nueva vista al proceso de reparto notarial extraordinario
 * Caso Mantis : 0010397
!!-->
<vista>
    <autenticacion>true</autenticacion>
    <nombre>reparto.notarial.nota.informativa.view</nombre>
    <parametro><llave>error</llave><valor>/WEB-
INF/jspSeguro/comun/ERROR_MULTIPLE.jsp</valor></parametro>
    <parametro><llave>head</llave><valor>/WEB-
INF/jspSeguro/comun/HEADER_LOGOUT_MENU.jsp</valor></parametro>
    <parametro><llave>body</llave><valor>/WEB-
INF/jspSeguro/repartonotarial/NOTAS_INFORMATIVAS_REPARTO_NOTARIAL.jsp</valor></parametro>
    <parametro><llave>bottom</llave><valor>/WEB-
INF/jspSeguro/comun/FOOTER_DEFAULT.jsp</valor></parametro>
</vista>

```

**CLASE:**

**org.auriga.smart.web.xml**

#### Adición Línea 470:

```

<!--
 * @author : Diana Lora
 * @change : se agrega una nueva vista al proceso de reparto notarial extraordinario
 * Caso Mantis : 0010397
!!-->
<navegacion>
    <resultado>AGREGAR_NOTA_INFORMATIVA</resultado>
    <url-redireccion>reparto.notarial.nota.informativa.view</url-redireccion>
</navegacion>

```

**Adición Línea 478:**

```

<!--
 * @author : Diana Lora
 * @change : se agrega una nueva vista al proceso de reparto notarial extraordinario
 * Caso Mantis : 0010397
!!-->
<mapping>
    <url-origen>reparto.notarial.nota.informativa.do</url-origen>
    <accion-web>gov.sir.core.web.acciones.repartonotarial.AWRepartoNotarial</accion-web>
    <informacion-navegacion>
        <control>gov.sir.core.web.navegacion.repartonotarial.ControlNavegacionRepartoNotarial</control>
        <navegacion>
            <resultado>AGREGAR_NOTA_INFORMATIVA</resultado>
            <url-redireccion>reparto.notarial.nota.informativa.view</url-redireccion>
        </navegacion>
        <navegacion>
            <resultado>REP_ENTREGA</resultado>
            <url-redireccion>reparto.entrega.view</url-redireccion>
        </navegacion>
        <navegacion>
            <resultado>REPARTO_OK</resultado>
            <url-redireccion>reparto.resultado.view</url-redireccion>
        </navegacion>
        <navegacion>
            <resultado>REPARTO_NOTARIAL_FAILED</resultado>
            <url-redireccion>turnos.view</url-redireccion>
        </navegacion>
    </informacion-navegacion>
</mapping>

```

**CLASE:****gov.sir.core.web.helpers.comun.TurnosHelper.java****Método:****public void drawGUI(HttpServletRequest request, JspWriter out) throws IOException, HelperException****Cambio Línea 162:**

```

out.write("<input type=\"hidden\" id=\"\" + WebKeys.REDIRECCION_DO + \"\" name=\"\" + WebKeys.REDIRECCION_DO + \"\" value=\"reparto.registro.do\" >");
        out.write("<input type=\"hidden\" id=\"\" + WebKeys.accion + \"\" name=\"\" + WebKeys.accion + \"\" value=\"\" + AWReparto.CREAR + \"\" >");
```

**Cambio Línea 165:**

```
out.write("<form id=\"\"+formName+\"\" name=\"\"+formName+\"\" method=\"post\" action=\"wait.view\">");
```

**Cambio Línea 168:**

```

out.write("<input type=\"hidden\" id=\"\" + WebKeys.REDIRECCION_DO + \"\" name=\"\" + WebKeys.REDIRECCION_DO + \"\" value=\"repartoNotarial.do\" >");
```

```
out.write("<input type=\"hidden\" id=\"\"+ACCION_REPARTO+\"\" name=\"\" + WebKeys.accion + \"\" value=\"\" + AWRepartoNotarial.EJECUTAR_REPARTO + \"\" >");
```

**Cambio Línea 172:**

```
out.write("<input type=\"hidden\" id=\"\" + WebKeys.REDIRECCION_DO + \"\" name=\"\" + WebKeys.REDIRECCION_DO + \"\" value=\"restitucionReparto.do\" >");
```

```
out.write("<input type=\"hidden\" id=\"\" + WebKeys.accion + \"\" name=\"\" + WebKeys.accion + \"\" value=\"\" + AWRestitucionReparto.CONSULTAR_TURNOS_ANALISIS_RESTITUCION+ \"\" >");
```

**Adición Línea 293:**

```

out.write("document.getElementById(\"" + WebKeys.REDIRECCION_DO + "\").value = 'reparto.notarial.nota.informativa.do' \n");
out.write("document.getElementById('ACCION_REPARTO').value = \"\" + AWRepartoNotarial.AGREGAR_NOTA_INFORMATIVA + \"\" \n");
out.write("document."+formName+".action = 'reparto.notarial.nota.informativa.do' \n");
out.write("document."+formName+".submit() \n");
```

**CLASE:****gov.sir.core.web.acciones.repartonotarial.AWRepartoNotarial.java****Adición de Import:**

```

import gov.sir.core.eventos.comun.EvnNotas;
import gov.sir.core.negocio.acciones.excepciones.ValidacionParametrosException;
import gov.sir.core.negocio.modelo.Nota;
import gov.sir.core.negocio.modelo.Proceso;
```

```

import gov.sir.core.negocio.modelo.TipoNota;
import gov.sir.core.negocio.modelo.constantes.CNota;
import gov.sir.core.negocio.modelo.constantes.CTipoNota;
import gov.sir.core.web.acciones.excepciones.ValidacionParametrosAgregarNotaException;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Date;
import java.util.Iterator;
import java.util.Vector;

Adición Línea 105:
/*
 * @author: Diana Lora
 * @change: Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 * En caso que se realice un reparto notarial extraordinario, se debe agregar una
 * nota informativa para dicho reparto.
 */
public final static String AGREGAR_NOTA_INFORMATIVA = "AGREGAR_NOTA_INFORMATIVA";
public final static String COLOCAR_DESCRIPCION = "COLOCAR_DESCRIPCION";
public static final String CANCELAR_REPARTO_NOTARIAL = "CANCELAR_REPARTO_NOTARIAL";
public static final String AGREGAR_NOTA = "AGREGAR_NOTA";
public static final String IMPRIMIR_NOTA_INFORMATIVA = "IMPRIMIR_NOTA_INFORMATIVA";

```

<b>Método:</b>	<b>public Evento perform(HttpServletRequest request) throws AccionWebException</b>
----------------	--

**Adición Línea 158:**

```

} else if (accion.equals(COLOCAR_DESCRIPCION)) {
    return buscarDescripcion(request);
} else if (accion.equals(AGREGAR_NOTA_INFORMATIVA)) {
    return guardarTurnosARepartir(request);
} else if (accion.equals(CANCELAR_REPARTO_NOTARIAL)) {
    return limpiarSesionRepartoExtraordinario(request);
} else if (accion.equals(AGREGAR_NOTA)) {
    return agregarNotaInformativa(request);
} else if (accion.equals(IMPRIMIR_NOTA_INFORMATIVA)) {
    try {
        return imprimirNotaInformativa(request);
    } catch (ValidacionParametrosException ex) {
        throw new AccionWebException(ex.getMessage());
    }
}

```

<b>Nuevo Método:</b>	<b>private Evento buscarDescripcion(HttpServletRequest request)</b>
----------------------	---

```

/*
 * Método carga la descripción de la nota informativa para reparto notarial extraordinario.
 * @author: Diana Lora
 * @change: Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 * @param request
 * @return
 */
private Evento buscarDescripcion(HttpServletRequest request) {
    ValidacionParametrosAgregarNotaException exception = new ValidacionParametrosAgregarNotaException();
    String idTipoNota = request.getParameter(CTipoNota.ID_TIPO_NOTA);
    if (idTipoNota == null || idTipoNota.equals(WebKeys.SIN_SELECCIONAR)) {
        exception.addError("Debe escoger un tipo de nota valido");
    }

    HttpSession session = request.getSession();

    Iterator itTiposNotas = ((List) request.getSession().getAttribute(WebKeys.LISTA_TIPOS_NOTAS)).iterator();
    while (itTiposNotas.hasNext()) {
        TipoNota elemento = (TipoNota) itTiposNotas.next();
        if (elemento.getIdTipoNota().equals(idTipoNota)) {
            session.setAttribute(CTipoNota.DESCRIPCION, elemento.getDescripcion());
        }
    }
}

```

```

        session.setAttribute(CTipoNota.ID_TIPO_NOTA, idTipoNota);
        session.setAttribute(CNota.DESCRIPCION, request.getParameter(CNota.DESCRIPCION));
        session.setAttribute(CNota.NOMBRE, elemento.getNombre());
    }
}

return null;
}

Nuevo Método:
private Evento agregarNotaInformativa(HttpServletRequest request) throws AccionWebException

/*
 * Agrega notas informativas para el reparto notarial extraordinario.
 * @author: Diana Lora
 * @change: Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 * @param request
 * @return
 * @throws AccionWebException
 */
private Evento agregarNotaInformativa(HttpServletRequest request) throws AccionWebException {
    HttpSession session = request.getSession();

    ValidacionParametrosAgregarNotaException exception = new ValidacionParametrosAgregarNotaException();
    Turno turno = (Turno) session.getAttribute(WebKeys.TURNO);
    Circulo circulo = (Circulo) session.getAttribute(WebKeys.CIRCULO);
    Proceso proceso = (Proceso) session.getAttribute(WebKeys.PROCESO);
    gov.sir.core.negocio.modelo.Usuario usuario = (gov.sir.core.negocio.modelo.Usuario)
    session.getAttribute(WebKeys.USUARIO);

    //CASO ESPECIAL NO EXISTE TURNO.
    if (turno == null) {
        //1. Se valida que se haya seleccionado un tipo de nota informativa.
        String idTipoNota = request.getParameter(CTipoNota.ID_TIPO_NOTA);
        if (idTipoNota == null || idTipoNota.equals(WebKeys.SIN_SELECCIONAR)) {
            exception.addError("Debe escoger un tipo de nota válido");
        }

        if (exception.getErrores().size() > 0) {
            throw exception;
        }

        //2. Se crea la nota.
        Nota notaInformativa = new Nota();
        Calendar calendar = Calendar.getInstance();

        notaInformativa.setIdNota(idTipoNota);
        notaInformativa.setAnio(String.valueOf(calendar.get(Calendar.YEAR)));
        notaInformativa.setIdCirculo(circulo.getIdCirculo());
        notaInformativa.setIdProceso(proceso.getIdProceso());
        notaInformativa.setIdTurno(null);
        notaInformativa.setTime(calendar.getTime());
        notaInformativa.setUsuario(usuario);

        String descripcion = request.getParameter(CNota.DESCRIPCION);
        if(descripcion!=null){
            descripcion = descripcion.toUpperCase();
        }
        notaInformativa.setDescripcion(descripcion);
        String descripcionTipo = request.getParameter((CTipoNota.DESCRIPCION));
        String nombreTipo = (String) request.getSession().getAttribute(CNota.NOMBRE);

        TipoNota tipoNota = new TipoNota();
        tipoNota.setDescripcion(descripcionTipo);
        tipoNota.setIdTipoNota(idTipoNota);
        tipoNota.setNombre(nombreTipo);
        notaInformativa.setTiponota(tipoNota);

        //3. Se agrega la nota a la lista de notas.
    }
}


```

```

List listaNotas = (List)
request.getSession().getAttribute(WebKeys.LISTA_NOTAS_INFORMATIVAS_SIN_TURNO);

//3.1 Si la lista es vacía se crea y se agrega la nota
if (listaNotas == null) {
    listaNotas = new ArrayList();
    listaNotas.add(notaInformativa);

}

//3.2 Si la lista ya tiene elementos, se agregan
else {
    listaNotas.add(notaInformativa);
}

//4. Se sube la lista a la sesión
session.setAttribute(WebKeys.LISTA_NOTAS_INFORMATIVAS_SIN_TURNO, listaNotas);

//5. Se retorna null para que no se genere ningún evento y se continue
//en la misma jsp.
return null;
}

//CASO EXISTE TURNO
else {

    if (turno == null) {
        exception.addError("El turno que se encontro en la sesion es invalido");
    }
    if (circulo == null) {
        exception.addError("El circulo que se encontro en la sesion es invalido");
    }
    if (proceso == null) {
        exception.addError("El proceso que se encontro en la sesion es invalido");
    }
    if (usuario == null) {
        exception.addError("El usuario que se encontro en la sesion es invalido");
    }

    String idTipoNota = request.getParameter(CTipoNota.ID_TIPO_NOTA);
    if (idTipoNota == null || idTipoNota.equals(WebKeys.SIN_SELECCIONAR)) {
        exception.addError("Debe escoger un tipo de nota valido");
    }
    String descripcion = request.getParameter(CNota.DESCRIPCION);
    if(descripcion!=null){
        descripcion = descripcion.toUpperCase();
    }
    String descripcionTipo = request.getParameter((CTipoNota.DESCRIPCION));

    if (exception.getErrores().size() > 0) {
        throw exception;
    }

    TipoNota tipoNota = new TipoNota();
    tipoNota.setDescripcion(descripcionTipo);

    tipoNota.setIdTipoNota(idTipoNota);

    List tiposNotas = (List) request.getSession().getAttribute(WebKeys.LISTA_TIPOS_NOTAS);
    Iterator itTiposNotas = tiposNotas.iterator();
    String nombre = "";
    while (itTiposNotas.hasNext()) {
        TipoNota elemento = (TipoNota) itTiposNotas.next();
        if (elemento.getIdTipoNota().equals(idTipoNota)) {
            nombre = elemento.getNombre();
        }
    }
}

```

```

        tipoNota.setNombre(nombre);

        Nota nota = new Nota();

        Date date = new Date();
        Calendar calendar = Calendar.getInstance();
        calendar.setTime(date);
        nota.setAnio(String.valueOf(calendar.get(Calendar.YEAR)));
        nota.setDescripcion(descripcion);
        nota.setIdCirculo(circulo.getIdCirculo());
        nota.setIdProceso(proceso.getIdProceso());
        nota.setIdTurno(turno.getIdTurno());
        nota.setTime(calendar.getTime());
        nota.setTiponota(tipoNota);
        nota.setTurno(turno);
        nota.setUser(usuario);
        org.auriga.core.modelo.transferObjects.Usuario usuarioAuriga =
(org.auriga.core.modelo.transferObjects.Usuario) request.getSession().getAttribute(SMARTKeys.USUARIO_EN_SESION);

        turno.addNota(nota);
        session.setAttribute(WebKeys.TURNO, turno);

        String numNotas = (String) session.getAttribute(WebKeys.NUM_NOTAS_FASE);

        if (numNotas == null) {
            numNotas = new String();
            numNotas = "0";
        }
        int i = Integer.valueOf(numNotas).intValue();
        i++;
        numNotas = String.valueOf(i);
        session.setAttribute(WebKeys.NUM_NOTAS_FASE, numNotas);
        return new EvnNotas(usuarioAuriga, nota, turno);
    }
}

```

#### Nuevo Método:

<b>private Evento imprimirNotaInformativa(HttpServletRequest request) throws ValidacionParametrosException</b>
--

```

/**
 * Imprime las notas informativas ingresadas para el reparto notarial extraordinario.
 * @author: Diana Lora
 * @change: Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 * @param request
 * @return
 * @throws ValidacionParametrosException
 */
private Evento imprimirNotaInformativa(HttpServletRequest request) throws ValidacionParametrosException {

    ValidacionParametrosException exception = new ValidacionParametrosException();

    List notas = null;
    List notasImpresion = new Vector();
    Nota nota = null;
    String valPos = request.getParameter(WebKeys.POSICION);
    int pos = 0;

    if(valPos == null){
        exception.addError("La posición de la nota informativa es inválida");
    }
    try{
        pos = Integer.valueOf(valPos).intValue();
        if(pos < 0){
            exception.addError("La posición de la nota informativa es inválida");
        }
    }catch(Exception e){
        exception.addError("La posición de la nota informativa es inválida");
    }
}

```

```

String valCopias = request.getParameter(WebKeys.NUMERO_COPIAS_IMPRESION);
int copias = 1;

if(valCopias != null){
    try{
        copias = Integer.valueOf(valCopias).intValue();
        if(copias < 1){
            exception.addError("El número de copias es inválido");
        }
    }catch(Exception e){
        exception.addError("El número de copias es inválido");
    }
}

if(valCopias == null || valCopias.equals("")){
    exception.addError("Por favor ingrese el numero de copias a imprimir.");
}

//1. Si existe un turno, se recupera la lista de notas informativas asociada
//con el turno.
Turno turno = (Turno)request.getSession().getAttribute(WebKeys.TURNO);
if (turno != null)
{
    notas = ((Turno)request.getSession().getAttribute(WebKeys.TURNO)).getNotas();
}

//2. Si el turno no ha sido creado, se recupera de sesión la lista de notas informativas.
else
{
    List listaNotas = (List)
request.getSession().getAttribute(WebKeys.LISTA_NOTAS_INFORMATIVAS_SIN_TURNO);

    //2.1 Si la lista no es vacía se obtienen las notas informativas.
    if (listaNotas != null)
    {
        notas = listaNotas;
    }
}

//Si se requiere la impresión de notas devolutivas
//if(request.getParameter(WebKeys.NOTA_DEVOLUTIVA)){
List listaAuxNotasDevolutivas = new ArrayList();
List listaAuxNotasInformativas = new ArrayList();
for (int i=0; i<notas.size(); i++){
    Nota notaAux = (Nota) notas.get(i);
    if (notaAux != null){
        TipoNota tipoNota = notaAux.getTiponota();
        if (tipoNota != null){
            if (tipoNota.isDevolutiva()){
                String idTipoNota = notaAux.getTiponota().getIdTipoNota();
                String idTipoNotaPadre="";
                if(!idTipoNota.equals("999") &&
!idTipoNota.substring(1,3).equals("50"))
                    && !idTipoNota.substring(1,3).equals("00")){
                    int separador =
                        if(separador>=5)
                            separador=5;
                        else
                            separador=0;
                    idTipoNotaPadre =
notaAux.getTiponota().getIdTipoNota().substring(0, 1) +Integer.toString(separador)+ "0";
                }
                String complemento =
"("++(idTipoNotaPadre.equals("")?"";idTipoNotaPadre+ " ")+idTipoNota+"")";
                notaAux.getTiponota().setDescripcion(notaAux.getTiponota().getDescripcion()+complemento);
                //end daniel
            }
        }
    }
}
}

```

```

        listaAuxNotasDevolutivas.add(notaAux);
    }
    else
        listaAuxNotasInformativas.add(notaAux);
    }
}
if(request.getParameter(WebKeys.NOTA_DEVOLUTIVA) != null){
    notas = listaAuxNotasDevolutivas;
}
else if(request.getParameter(WebKeys.NOTA_INFORMATIVA) != null)
    notas = listaAuxNotasInformativas;

if(notas == null){
    notas = new Vector();
}

if(pos >= notas.size()){
    exception.addError("La nota informativa que desea ver esta fuera del rango");
}

if (exception.getErrores().size() > 0) {
    throw exception;
}

nota = (Nota) notas.get(pos);
if(turno!=null){
    nota.setTurno(turno);
}

notasImpresion.add(nota);

Circulo circulo = (Circulo)request.getSession().getAttribute(WebKeys.CIRCULO);
org.auriga.core.modelo.transferObjects.Usuario usuarioAuriga = (org.auriga.core.modelo.transferObjects.Usuario)
request.getSession().getAttribute(SMARTKeys.USUARIO_EN_SESION);
gov.sir.core.negocio.modelo.Usuario usuarioSIR = (gov.sir.core.negocio.modelo.Usuario)
request.getSession().getAttribute(WebKeys.USUARIO);
String UID = request.getSession().getId();

EvnNotas evn = new EvnNotas(usuarioAuriga, EvnNotas.IMPRIMIR_NOTA_INFORMATIVA, notasImpresion,
UID, circulo, turno, usuarioSIR);
evn.setNumCopiasImpresion(copias);
return evn;
}

Nuevo Método:
public Evento guardarTurnosARepartir(HttpServletRequest request)

/*
 * Toma los valores del request y los ingresa en la session para se utilizados
 * posteriormente en el reparto notarial extraordinario.
 * @author: Diana Lora
 * @change: Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 * @author: Diana Lora
 * @param request
 * @return
 */
public Evento guardarTurnosARepartir(HttpServletRequest request){

String numActas = request.getParameter(CMinuta.NUMERO_ACTAS);
String numResumenes = request.getParameter(CMinuta.NUMERO_RESUMENES);
String tipoReparto = request.getParameter(CMinuta.TIPO_REPARTO);
String[] idsTurnos = request.getParameterValues(CTurno.ID_TURNO);

HttpSession sesion = request.getSession();

sesion.setAttribute(CMinuta.NUMERO_ACTAS, numActas);
sesion.setAttribute(CMinuta.NUMERO_RESUMENES, numResumenes);
sesion.setAttribute(CMinuta.TIPO_REPARTO, tipoReparto);
```

```
sesion.setAttribute(CTurno.ID_TURNO, idsTurnos);
```

```
    return null;
```

```
}
```

#### Nuevo Método:

##### **public Evento limpiarSesionRepartoExtraordinario(HttpServletRequest request)**

```
/**
```

```
* En caso de cancelar el reparto notarial extraordinario, limpia de sesión  
* los valores seteados.  
* @author: Diana Lora  
* @change: Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial  
* @param request  
* @return  
*/
```

```
public Evento limpiarSesionRepartoExtraordinario(HttpServletRequest request){
```

```
    HttpSession session = request.getSession();
```

```
    session.removeAttribute(CMinuta.NUMERO_ACTAS);  
    session.removeAttribute(CMinuta.NUMERO_RESUMENES);  
    session.removeAttribute(CMinuta.TIPO_REPARTO);  
    session.removeAttribute(CTurno.ID_TURNO);  
    session.removeAttribute(WebKeys.LISTA_NOTAS_INFORMATIVAS_SIN_TURNO);  
    return null;
```

```
}
```

#### Método:

##### **private EvnRepartoNotarial crearReparto(HttpServletRequest request) throws AccionWebException**

#### Método Original:

```
/**
```

```
* Método que se encarga de realizar el proceso de reparto de minutas y asignarlas a las  
* diferentes notarías dependiendo de las condiciones previamente establecidas  
* @param request  
* @return  
* @throws AccionWebException  
*/
```

```
private EvnRepartoNotarial crearReparto(HttpServletRequest request) throws AccionWebException {
```

```
    HttpSession sesion = request.getSession();
```

```
    Estacion e = (Estacion) sesion.getAttribute(WebKeys.ESTACION);
```

```
    Circulo circulo = (Circulo) request.getSession().getAttribute(WebKeys.CIRCULO);
```

```
    Usuario u = (Usuario) sesion.getAttribute(SMARTKeys.USUARIO_EN_SESION);
```

```
    String UID = request.getSession().getId();
```

```
    gov.sir.core.negocio.modelo.Usuario usuarioSIR = (gov.sir.core.negocio.modelo.Usuario)
```

```
request.getSession().getAttribute(WebKeys.USUARIO);
```

```
//NÚMERO DE IMPRESIONES.
```

```
String numActas = request.getParameter(CMinuta.NUMERO_ACTAS);
```

```
String numResumenes = request.getParameter(CMinuta.NUMERO_RESUMENES);
```

```
String tipoReparto = request.getParameter(CMinuta.TIPO_REPARTO);
```

```
String idsTurnos[] = request.getParameterValues(CTurno.ID_TURNO);
```

```
EvnRepartoNotarial evento = new EvnRepartoNotarial(u, usuarioSIR, circulo, e,
```

```
EvnRepartoNotarial.EJECUTAR_REPARTO, numActas, numResumenes, tipoReparto, idsTurnos);
```

```
evento.setUID(UID);
```

```
return evento;
```

```
}
```

#### Refactorización del método:

```
/**
```

```
* Método que se encarga de realizar el proceso de reparto de minutas y asignarlas a las  
* diferentes notarías dependiendo de las condiciones previamente establecidas  
* @param request  
* @return  
* @throws AccionWebException  
*/
```

```
private EvnRepartoNotarial crearReparto(HttpServletRequest request) throws AccionWebException {
```

```
    HttpSession session = request.getSession();
```

```
    Estacion e = (Estacion) session.getAttribute(WebKeys.ESTACION);
```

```
    Circulo circulo = (Circulo) request.getSession().getAttribute(WebKeys.CIRCULO);
```

```

        Usuario u = (Usuario) session.getAttribute(SMARTKeys.USUARIO_EN_SESION);
        // String UID =request.getSession().getId();
        String UID = circulo.getIdCirculo();
        gov.sir.core.negocio.modelo.Usuario usuarioSIR = (gov.sir.core.negocio.modelo.Usuario)
        request.getSession().getAttribute(WebKeys.USUARIO);

        //NÚMERO DE IMPRESIONES.
        String tipoReparto = request.getParameter(CMinuta.TIPO_REPARTO);
        if(tipoReparto != null){
            //REPARTO ORDINARIO
            String numActas = request.getParameter(CMinuta.NUMERO_ACTAS);
            String numResumenes = request.getParameter(CMinuta.NUMERO_RESUMENES);
            String[] idsTurnos = request.getParameterValues(CTurno.ID_TURNO);

            EvnRepartoNotarial evento = new EvnRepartoNotarial(u, usuarioSIR, circulo, e,
            EvnRepartoNotarial.EJECUTAR_REPARTO, numActas, numResumenes, tipoReparto, idsTurnos);
            evento.setUID(UID);
            return evento;
        }else{
            tipoReparto = (String) session.getAttribute(CMinuta.TIPO_REPARTO);
            String numActas = (String) session.getAttribute(CMinuta.NUMERO_ACTAS);
            String numResumenes = (String) session.getAttribute(CMinuta.NUMERO_RESUMENES);
            String[] idsTurnos = (String[]) session.getAttribute(CTurno.ID_TURNO);
            List notasInformativas = (List) session.getAttribute(WebKeys.LISTA_NOTAS_INFORMATIVAS_SIN_TURNO);
            EvnRepartoNotarial evento = new EvnRepartoNotarial(u, usuarioSIR, circulo, e,
            EvnRepartoNotarial.EJECUTAR_REPARTO, numActas, numResumenes, tipoReparto, idsTurnos, notasInformativas);
            evento.setUID(UID);
            return evento;
        }
    }
}

```

#### CLASE:

gov.sir.core.eventos.repartonotarial. EvnRepartoNotarial.java

#### Nuevo Constructor Línea 138:

```

public EvnRepartoNotarial(Usuario usuario, gov.sir.core.negocio.modelo.Usuario usuarioSIR, Circulo circulo, Estacion estacion, String
tipoAcccion, String numeroActas, String numeroResumenes, String tipoReparto, String[] turnosExtraordinarios, List notasInformativas)
{
    super(usuario, tipoAcccion);
    this.circulo = circulo;
    this.numeroActas = numeroActas;
    this.numeroResumenes = numeroResumenes;
    this.usuarioSIR = usuarioSIR;
    this.tipoReparto = tipoReparto;
    this.turnosExtraordinarios = turnosExtraordinarios;
    this.notasInformativas = notasInformativas;
}

```

#### CLASE:

gov.sir.core.web.navegacion.repartonotarial. ControlNavegacionRepartoNotarial.java

#### Nueva Variable Línea 31:

```

/**
 * @author: Diana Lora
 * @change: Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 * En caso que se realice un reparto notarial extraordinario, se debe agregar una
 * nota informativa para dicho reparto.
 */
public final static String AGREGAR_NOTA_INFORMATIVA = "AGREGAR_NOTA_INFORMATIVA";

```

#### Método:

public String procesarNavegacion(HttpServletRequest request) throws ControlNavegacionException

#### Adición Línea 70:

```

else if (accion.equals(AWRepartoNotarial.AGREGAR_NOTA_INFORMATIVA)) {
    return AGREGAR_NOTA_INFORMATIVA;
} else if (accion.equals(AWRepartoNotarial.COLOCAR_DESCRIPCION)) {
    return AGREGAR_NOTA_INFORMATIVA;
} else if (accion.equals(AWRepartoNotarial.AGREGAR_NOTA)) {
    return AGREGAR_NOTA_INFORMATIVA;
} else if (accion.equals(AWRepartoNotarial.IMPRIMIR_NOTA_INFORMATIVA)) {
    return AGREGAR_NOTA_INFORMATIVA;
} else if (accion.equals(AWRepartoNotarial.CANCELAR_REPARTO_NOTARIAL)) {

```

	return REPARTO_NOTARIAL_FAILED;
	<b>CLASE:</b>
	<b>gov.sir.core.negocio.acciones.repartonotarial.ANRepartoNotarial.java</b>
<b>Adición de Import:</b>	import gov.sir.core.negocio.acciones.excepciones.NotaNoAgregadaException;
	<b>Método:</b>
	<b>private EvnRespRepartoNotarial ejecutarReparto(EvnRepartoNotarial evento) throws EventoException</b>
<b>Adición Línea 719:</b>	<pre>     /**      * @author: Diana Lora      * @change: Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial      * Se guardan las notas informativas asociadas al turno extraordinario repartido.      */     if (!evento.getTipoReparto().equals(CMinuta.ORDINARIO)) {          String[] turnosExtraordinarios = evento.getTurnosExtraordinarios();         for (int j = 0; j &lt; turnosExtraordinarios.length; j++) {             // ID_WORKFLOW: 1. ANIO 2. CIRCULO 3. PROCESO 4. CONSECUITIVO             String[] turnoExtraordinario = turnosExtraordinarios[j].split("-");              TurnoPk id = new TurnoPk();             id.anio = turnoExtraordinario[0];             id.idCirculo = turnoExtraordinario[1];             id.idProceso = Integer.parseInt(turnoExtraordinario[2]);             id.idTurno = turnoExtraordinario[3];              List notasInformativas = evento.getNotasInformativas();             if(notasInformativas != null){                 for (int k = 0; k &lt; notasInformativas.size(); k++) {                     Nota nota = (Nota) notasInformativas.get(k);                      try {                         hermod.addNotaToTurno(nota, id);                     } catch (HermodException e) {                         throw new NotaNoAgregadaException("La nota no pudo ser agregada", e);                     } catch (Throwable e) {                         ExceptionPrinter printer = new ExceptionPrinter(e);                         Log.getInstance().error(ANRepartoNotarial.class,"Ha ocurrido una excepcion inesperada " +                             printer.toString());                     }                 }             }         }     } } </pre>
	<b>NUEVA JSP:</b>
	<b>WEB-INF/jspSeguro/repartonotarial/NOTAS_INFORMATIVA_REPARTO_NOTARIAL.jsp</b>
	<b>WEB-INF/jspSeguro/repartonotarial/NOTAS_INFORMATIVA_REPARTO_NOTARIAL.jsp</b>
	<b>CLASE:</b>
	<b>gov.sir.core.web.acciones.repartonotarial.AWLiquidacionRepartoNotarial</b>
en el procedimiento: validarInformacion;	
se debe actualizar la linea 784	
String cuantia = cuantiaOriginal.replace(".", "");	
por	
String cuantia = (cuantiaOriginal.replace(".", "")).replace(",", ""));	
	<b>Clase</b>
	<b>gov.sir.core.eventos.comun.EvnRespTurno.java</b>
Incluir declaración como código nuevo:	
/**	
* @author : Edgar Lora	

```

 * @change : Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 */
private boolean enHorario;

```

Incluir declaración como código nuevo:

```

 /**
 * @author : Edgar Lora
 * @change : Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 */
public boolean isEnHorario() {
    return enHorario;
}
public void setEnHorario(boolean enHorario) {
    this.enHorario = enHorario;
}

```

### Clase

**gov.sir.core.negocio.acciones.comun.ANTurno.java**

### Procedimiento Perform

Línea 331 - 354

Remplazar:

```

try {
    Collections.sort(turnos, new IDidTurnoComparator());
} catch (Exception e) {
    Log.getInstance().error(ANTurno.class,"No se pudieron ordenar los Turnos");
}
String tipoRepartoNotarial = "";
if (fase.getID().equals(CFase.REP_REPARTO)) {
    tipoRepartoNotarial = "REP_REPARTO_NOTARIAL";
}
if ("REP_REPARTO_NOTARIAL".equals(tipoRepartoNotarial)) {
    ValidacionesSIR validacion = new ValidacionesSIR();
    try {
        if("true".equals(validacion.HorarioNotarial(circulo.getIdCirculo(), tipoRepartoNotarial))){
            throw new ListarNoEfectuadoException("No se puede hacer reparto notarial en este
horario");
        }
    } catch (GeneralSIRException ex) {
        Logger.getLogger(ANTurno.class.getName()).log(Level.SEVERE, null, ex);
    }
}
EvnRespTurno eventoResp =new EvnRespTurno(turnos);
eventoResp.setParametros(tabla);
eventoResp.setTurnoSirMig(turnosSirMig);
return eventoResp;

```

Por:

```

/** * @author : Edgar Lora
 * @change : Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
*/
EvnRespTurno eventoResp = new EvnRespTurno(turnos);
try {

```

```

        Collections.sort(turnos, new IDidTurnoComparator());
    } catch (Exception e) {
        Log.getInstance().error(ANTurno.class,"No se pudieron ordenar los Turnos");
    }
    String tipoRepartoNotarial = "";
    if (fase.getID().equals(CFase.REP_REPARTO)) {
        tipoRepartoNotarial = "REP_REPARTO_NOTARIAL";
    }
    if ("REP_REPARTO_NOTARIAL".equals(tipoRepartoNotarial)) {
        ValidacionesSIR validacion = new ValidacionesSIR();
        try {
            if("true".equals(validacion.HorarioNotarial(circulo.getIdCirculo(), tipoRepartoNotarial))){
                eventoResp.setEnHorario(false);
                //throw new ListarNoEfectuadoException("No se puede hacer reparto notarial en este horario");
            }else{
                eventoResp.setEnHorario(true);
            }
        } catch (GeneralSIRException ex) {
            Logger.getLogger(ANTurno.class.getName()).log(Level.SEVERE, null, ex);
        }
    }
    eventoResp.setParametros(tabla);
    eventoResp.setTurnoSirMig(turnosSirMig);
    return eventoResp;

```

**Clase**

**gov.sir.core.web.acciones.comun.AWTurno.java**

**Procedimiento doEnd**

Incluir como código nuevo en la línea 605

```

/**
 * @author : Edgar Lora
 * @change : Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes
Reparto Notarial
 */
if(!evResp.isEnHorario()){

request.getSession().setAttribute(WebKeys.REPARTO_NOTARIAL_EN_HORARIO, "No se puede
hacer reparto notarial en este horario");
}

```

**Clase**

**gov.sir.core.web.WebKeys.java**

Incluir declaración como código nuevo

```

/**
 * @author : Edgar Lora
 * @change : Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
*/
public static final String REPARTO_NOTARIAL_EN_HORARIO =
"REPARTO_NOTARIAL_EN_HORARIO";

```

**Clase**

**gov.sir.core.web.helpers.comun.MenuHelper.java**

### Procedimiento limpiarInfoSession

Incluir como código nuevo en la línea 178

```
!nombre.equals(WebKeys.REPARTO_NOTARIAL_EN_HORARIO) &&
    Clase
        gov.sir.core.web.helpers.comun.TurnosHelper.java
```

Incluir como código nuevo

```
/**
 * @author : Edgar Lora
 * @change : Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 */
private boolean enHorarioRepartoNotarial = true;
```

Incluir como código nuevo

```
/**
 * @author : Edgar Lora
 * @change : Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes Reparto Notarial
 */
public boolean isEnHorarioRepartoNotarial() {
    return enHorarioRepartoNotarial;
}

public void setEnHorarioRepartoNotarial(boolean enHorarioRepartoNotarial) {
    this.enHorarioRepartoNotarial = enHorarioRepartoNotarial;
}
```

### Clase

```
gov.sir.core.web.helpers.comun.TurnosHelper.java
Procedimiento drawGUI
```

Línea 468

Remplazar:

```
out.write("<td align=\"center\" colspan=\"2\"><input name=\"imageField\" id=\"imageField_id\" type=\"image\" src=\"\" + request.getContextPath() + \"/jsp/images/btn_repartir.gif\" width=\"139\" onClick=\"javascript:cambiarTipoRepartoNotarial(\"+CMinuta.ORDINARIO+)\" height=\"21\" border=\"0\"></td>");
```

Por:

```
if(enHorarioRepartoNotarial){
    out.write("<td align=\"center\" colspan=\"2\"><input
name=\"imageField\" id=\"imageField_id\" type=\"image\" src=\"\" + request.getContextPath() +
\"/jsp/images/btn_repartir.gif\" width=\"139\""
onClick=\"javascript:cambiarTipoRepartoNotarial(\"+CMinuta.ORDINARIO+)\" height=\"21\""
border=\"0\"></td>");
}
else{
    out.write("<td align=\"center\" colspan=\"2\"><a
href=\"javascript:alert('No es horario para reparto de las minutas de tipo ordinario.')\"><image src=\"\" +
request.getContextPath() + \"/jsp/images/btn_repartir.gif\" width=\"139\" height=\"21\""
border=\"0\"></a></td>");
}
```

### JSP

```
WEB-INF/jspSeguro/comun/LISTAR_TURNOS.jsp
```

Línea 8

Remplazar:

```
<% TurnosHelper turnosHelper = new TurnosHelper();%>
```

Por:

```
<% TurnosHelper turnosHelper = new TurnosHelper();
String mensaje =
(String)request.getSession().getAttribute(WebKeys.REPARTO_NOTARIAL_EN_HORARIO);
%>
```

Línea 165, incluir código nuevo:

```
if(mensaje == null){
    turnosHelper.setEnHorarioRepartoNotarial(true);
} else{
    turnosHelper.setEnHorarioRepartoNotarial(false);

request.getSession().removeAttribute(WebKeys.REPARTO_NOTARIAL_EN_HORARIO);
}
```

Antes de la línea:

```
turnosHelper.render(request,out);
```

#### Clase

#### gov.sir.core.web acciones.comun.AWTurno.java

En el procedimiento doEnd, en la línea 609 se debe incluir la siguiente línea de código:

```
request.getSession().removeAttribute(WebKeys.REPARTO_NOTARIAL_EN_HORARIO);
```

Como resultado, el bloque de código que estaba así:

```
/***
 * @author : Edgar Lora
 * @change : Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes
Reparto Notarial
*/
if(!evResp.isEnHorario()){
    request.getSession().setAttribute(WebKeys.REPARTO_NOTARIAL_EN_HORARIO, "No se puede hacer reparto notarial en este horario");
}
```

debe quedar así:

```
/***
 * @author : Edgar Lora
 * @change : Mantis 0010397: Acta - Requerimiento No 063_151 - Ajustes
Reparto Notarial
*/
request.getSession().removeAttribute(WebKeys.REPARTO_NOTARIAL_EN_HORARIO);
if(!evResp.isEnHorario()){
    request.getSession().setAttribute(WebKeys.REPARTO_NOTARIAL_EN_HORARIO, "No se puede hacer reparto notarial en este horario");
}
```

<b>Clase</b> <b>gov.sir.core.web.acciones.repartonotarial.AWLiquidacionRepartoNotarial.java</b>
Se modifco Línea 806
Remplazar: exception.addError("El reparto es extraordinario. Debe ingresar una nota informativa.");
por: exception.addError("La radicación de la minuta es extraordinaria. Debe ingresar una nota informativa.");
----- Línea 865
Remplazar: exception.addError("Debe seleccionar el valor de la cuantía");
Por: exception.addError("Debe seleccionar el valor de la Cuantia y/o Unidades");
<b>Clase</b> <b>WEB-INF/jspSeguro/repartonotarial/SOLICITUD_REPARTO_NOTARIAL.jsp</b>
Se modifco
Línea 434 Remplazar: <td></td>
Por: <td class="titulotbcentral" width="10%">Cuantía y/o Unidades</td>
<b>Clase</b> <b>WEB-INF/jspSeguro/comun/LISTAR_TURNOS.jsp</b>
Archivo:
La línea 369 debe ser borrada: session.removeAttribute(WebKeys.REPARTO_NOTARIAL_EN_HORARIO);
<b>Clase</b> <b>gov.sir.core.negocio.acciones.repartonotarial. ANRepartoNotarial.java</b>
import gov.sir.core.negocio.modelo.DocPagoEfectivo; import java.util.Date;
<b>Procedimiento crearSolicitud</b>
Línea nueva 262 Pago pago = null;
Línea 331 – 347 Remplazar:
solicitud = (Solicitud) hermod.crearSolicitud(evento.getSolicitudRepartoNotarial());
if (solicitud == null) { throw new LiquidacionNoEfectuadaException("No existe solicitud asociada"); }
SolicitudRepartoNotarial solicitudRepartoNotarial = (SolicitudRepartoNotarial) solicitud;

```

solicitudRepartoNotarial =
hermod.addMinutaToSolicitudReparto(solicitudRepartoNotarial, evento.getMinuta());

    Usuario usuarioSir = evento.getUsuarioSIR();
    String idEstacion = estacion.getEstacionId();
    try {
        //1. Crear el turno
        turno =
hermod.crearTurnoRepartoNotarial(solicitudRepartoNotarial,idEstacion,usuarioSir);

Por:
/***
 * @autor: Edgar Lora
 * @mantis: 0010397
 * @requerimiento: 063_151
 */
LiquidacionTurnoRepartoNotarial liquidacion = new
LiquidacionTurnoRepartoNotarial();
solicitud = (Solicitud)
hermod.crearSolicitud(evento.getSolicitudRepartoNotarial());

if (solicitud == null) {
    throw new LiquidacionNoEfectuadaException("No existe solicitud
asociada");
}

SolicitudRepartoNotarial solicitudRepartoNotarial = (SolicitudRepartoNotarial)
solicitud;
solicitudRepartoNotarial =
hermod.addMinutaToSolicitudReparto(solicitudRepartoNotarial, evento.getMinuta());

    Usuario usuarioSir = evento.getUsuarioSIR();
    String idEstacion = estacion.getEstacionId();

liquidacion.setSolicitud(solicitud);
liquidacion.setUsuario(usuarioSir);
LiquidacionTurnoRepartoNotarial auxLiquidacion =
(LiquidacionTurnoRepartoNotarial)hermod.liquidar(liquidacion);
DocPagoEfectivo docPago = new DocPagoEfectivo(liquidacion.getValor());
AplicacionPago appEfectivo = new AplicacionPago();
appEfectivo.setIdLiquidacion(liquidacion.getIdLiquidacion());
appEfectivo.setIdSolicitud(solicitud.getIdSolicitud());
appEfectivo.setValorAplicado(liquidacion.getValor());
appEfectivo.setDocumentoPago(docPago);

pago = new Pago(auxLiquidacion, null);
pago.addAplicacionPago(appEfectivo);
pago.setIdLiquidacion(liquidacion.getIdLiquidacion());
pago.setIdSolicitud(solicitud.getIdSolicitud());
pago.setLiquidacion(auxLiquidacion);
pago.setUsuario(usuarioSir);

```

```

auxLiquidacion.setSolicitud(solicitud);

//String numRecibo = ((Liquidacion) liquidaciones.get(liquidaciones.size() - 1)).getPago().getNumRecibo();
EstacionReciboPk estacionReciboID = new EstacionReciboPk();
estacionReciboID.idEstacion = idEstacion;

Circulo circulo = evento.getCirculo();
String fechaImpresion = this.getFechaImpresion();
Calendar calendar = Calendar.getInstance();
calendar.setTime(new Date());
long reciboMinutaRepartoNotarial =
hermod.getSecuencialRepartoNotarial(circulo.getIdCirculo(),
String.valueOf(calendar.get(Calendar.YEAR)));
pago.setNumRecibo(String.valueOf(reciboMinutaRepartoNotarial));

if (validarPago(pago,
hermod.getRangoAceptacionPago(
solicitud.getProceso().getNombre())))) {
pago = hermod.validarPago(pago);
}
try {
//1. Crear el turno
turno = hermod.procesarPago(pago, estacion.getEstacionId(), "imp",
pago.getUsuario(), null, false);
pago.getLiquidacion().getSolicitud().setTurno(turno);

```

Clase

gov.sir.core.negocio.acciones.administracion. ANImprimirFolio.java

Línea 353

Remplazar:

```
ImprimibleRecibo impRec = new ImprimibleRecibo(pago, circulo, fechaImpresion,
CTipoImprimible.RECIBO);
```

Adición del Radiobutton (Cuantia y/o Unidades) en la sección Acciones Notariales, del formulario Radicación de minutias;

Cuantía y/o Unidades	#	Tipo Accion Notarial	Cuantía	Unidades	Con Cuentía
<input checked="" type="checkbox"/>	1	ACLARACION	12,000	0	SI <input checked="" type="radio"/>
<input checked="" type="checkbox"/>	2	HIPOTECA	20,000,000	0	SI <input checked="" type="radio"/>

Total Cuentía: 20012600.0 Total Unidades Inmobiliarias: 0

Con Cuentía:  Tipo de Contrato:  Cuentía:   
 Seleccionar en el radiobutton el acto que categorizara la minuta  
 HIPOTECA Unidades Inmobiliarias:

Agregar

#### 065 MODIFICACIÓN\_CAMPOS\_PORCENTAJE\_SIR\_NE\_FOLIO\_DERIVADO Y SIR\_NE\_FOLIO\_DERIVADO\_TMP (Informativo)

Teniendo en cuenta el caso 32196 de la ORIP de Montería, se solicita actualizar las tablas sir\_ne\_folio\_derivado y sir\_ne\_folio\_derivado\_tmp, para que los campos FLDR\_PORCENTAJE y

FDTM\_PORCENTAJE respectivamente sean actualizados en su tamaño dejándolos como varchar2(50) que es el tamaño que corresponde al campo NTCD:PORCENTAJE de la tabla SIR\_NE\_ANOTACIÓN\_CIUDADANO.

#### **Analisis/Desarrollo**

Se realiza el cambio del tamaño de los campos de porcentaje solicitado por la SNR, por favor indicarle al usuario que puede avanzar el turno 2012-140-6-1089.

#### **DESARROLLO**

Se ejecuta los siguientes scripts en ambiente de producción el 13 de Febrero del 2012.

```
ALTER TABLE SIR_NE_FOLIO_DERIVADO_TMP  
MODIFY FDTM_PORCENTAJE VARCHAR2(50 BYTE);
```

```
ALTER TABLE SIR_NE_FOLIO_DERIVADO  
MODIFY FLDR_PORCENTAJE VARCHAR2(50 BYTE);
```

```
ALTER TABLE SIR_OP_WEB_FOLIO_DERIVADO  
MODIFY WBFD_PORCENTAJE VARCHAR2(50 BYTE);
```

#### **LOG DE EJECUCION**

table SIR\_NE\_FOLIO\_DERIVADO\_TMP alterado.

table SIR\_NE\_FOLIO\_DERIVADO alterado.

table SIR\_OP\_WEB\_FOLIO\_DERIVADO alterado.

#### **PRUEBA**

Turno: 2012-999-6-12

Folio: 999-1089

Segregación: 999-7617  
999-7611

Se consultan los datos de los folios segregados y son correctos.

#### **DESPLIEGUE – 2.7.4**

**10/02/2012**

#### **008 Reporte 24\_Modificacion llamado reporte**

Reporte 24 – Informe mayores valores ingresados del día xx al día yy

Actualmente el jasper se encarga de ejecutar la consulta a varias tablas lo cual genera lentitud en la generación del reporte. La metodología implementada para los reportes es ejecutar un procedimiento y almacenar en una tabla, esta tabla será la que se llame del jasper.

#### **Desarrollo**

- Script Bases de Datos:  
SCRIPT\_MANTIS\_8645.sql
- Script Bases de Datos: Procedimiento  
SP\_REPORTE\_24.sql

#### **CLASE**

SIR-APP\src\gov\sir\core\web\acciones\administracion\ AWReportes.java

//GCABRERA 15/12/2011

//Se agrega parámetro REPORTE\_24\_PARAM\_USUARIOLOG

```

public static final String REPORTE_24_PARAM_USUARIOLOG = "P_USUARIO_LOG";
CLASE
SIR-APP\src\gov\sir\core\web\acciones\administracion\ AWReportes.java
Metodo:
private void do_Reporte24_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {
//GCABRERA 15/12/2011
//Se agrega parámetro REPORTE_24_PARAM_USUARIOLOG
String param_04 = request.getParameter(REPORTE_24_PARAM_USUARIOLOG);

REPORTE_XX_PARAM_CMDKEY, REPORTE_24_PARAM_PLIQUIDACIONFECHAINI,
REPORTE_24_PARAM_PLIQUIDACIONFECHAFIN, REPORTE_24_PARAM_PCIRCULONOMBRE,
REPORTE_24_PARAM_USUARIOLOG
Metodo private void do_Reporte24_SaveState(HttpServletRequest request)

REPORTE_24_PARAM_PLIQUIDACIONFECHAINI, REPORTE_24_PARAM_PLIQUIDACIONFECHAFIN,
REPORTE_24_PARAM_PCIRCULONOMBRE, REPORTE_24_PARAM_USUARIOLOG

Metodo private void do_Reporte24_RemoveState(HttpServletRequest request)
REPORTE_24_PARAM_PLIQUIDACIONFECHAINI, REPORTE_24_PARAM_PLIQUIDACIONFECHAFIN,
REPORTE_24_PARAM_PCIRCULONOMBRE, REPORTE_24_PARAM_USUARIOLOG
CLASE
SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDOResultadosJasperDAO.java
//GCABRERA 15/12/2011
//Reporte 24
//Se obtienen los parámetros para enviar al procedimiento
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_24_PARAM_PLIQUIDACIONFECHAINI);
listaParametros.add(AWReportes.REPORTE_24_PARAM_PLIQUIDACIONFECHAFIN);
listaParametros.add(AWReportes.REPORTE_24_PARAM_PCIRCULONOMBRE);
mapaReportes.put(AWReportes.REPORTE_24_PARAM_CMDKEY, new Object[]{
    "SELECT SIR.SEQ_Reporte_24.NEXTVAL FROM dual",
    "sp_reporte_24(?, ?, ?, ?)", listaParametros,
    "DELETE FROM sir_op_reporte_24 WHERE sec_proceso = "});
JSP
SIR-APP\web\WEB-INF\jsp\Seguro\administracion\ADMINREPORTES_REPORT24_PARAMFORM.jsp
<%-- GCABRERA 15/12/2011
Se agrega parámetro REPORTE_24_PARAM_USUARIOLOG
--%>
<%
Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
%>
<input type="hidden" name="<%=AWReportes.REPORTE_24_PARAM_USUARIOLOG%>"  

id="<%=AWReportes.REPORTE_24_PARAM_USUARIOLOG%>"  

value="<%="usuario.getUsername()%>">
```

## 010 Error turno anulado - Reparto Notarial (Informativo)

Requerimiento GLPI No 24562 - Turno:2010-230-7-255

**Motivo Inconsistencia:** Se recibe correo con el siguiente requerimiento: "Les informo que el Aplicativo en el reparto del hoy, repartió y asignó el turno 2010-230-7-255 que fue anulado, no aparecía en el listado para repartir, pero en el radicador que imprime cuando se realizar el reparto salió asignada a la notaría Tercera y también imprimió la carátula de la minuta y en el reporte 32 Radicador de minutos por reparto no la relaciona. Adjunto los archivos para que verifiquen y miren porque está haciendo esto".

El turno 2010-230-7-255 se anuló el día de ayer por solicitud de la oficina, favor verificar los datos del turno ya que no debió entrar al Reparto notarial. Grupo Villavicencio  
Se relaciona el caso 059 teniendo en cuenta que el turno 2010-230-7-255 se quedó sin fase ni estación, se abre acta nueva No 241, para verificar si el proceso de reparto valida que los turnos a repartir no se encuentren anulados y si esta no existe agregarla.

**NOTA:** La inconsistencia reportada en el caso 059 se presenta aleatoriamente por tal motivo se dio por cerrado el caso. Error presentado por persistencia.

**Desarrollo**

Se realizaron pruebas con acompañamiento de mesa de ayuda y no se replicó el caso, esta incidencia se le atribuye a problemas de conectividad.

**Análisis:**

Se realizaron pruebas radicando turnos de reparto notarial, anulando unos y realizando el reparto notarial con los turnos restantes.

Una de las pruebas fueron con los turnos: 2011-300-7-590 y 2011-300-7-591. El primer turno fue anulado. Se realizó el reparto notarial y solo repartió el turno 2011-300-7-591.

Se revisó la consulta que se hace a la base de datos y esta es correcta. De igual forma se verificó que el algoritmo de reparto notarial hiciera todo correctamente.

La inconsistencia se presentó por la perdida de fase en el turno de reparto notarial 2010-230-7-255.

En el caso 059 del contrato 374, revisado en el despliegue 2.7.1, se analizó el tema de perdida de estaciones y fases:

"En la revisión del código se evidencio que cada uno de los modelos utilizados para realizar el avance de turnos de una a fase a otra, podrían presentar algunos inconvenientes pero que su modificación conllevaría a mejoras de performance y cambios generales de la aplicación. Estos cambios se deberían enfocar a transacciones atómicas (los cambios o bien ocurren todos o ninguno), los cuales deben ser detallados como cambio de alto impacto de aplicación."

**015 REQUERIMIENTO REPORTE 39-vimm**

**REQUERIMIENTO: REPORTE 39 DOCUMENTOS EN CORRECCIONES**

Se requiere que:

- 13) Después de la columna radicación se debe adicionar la columna con titulo "Fecha Radicación"  
En esta columna se debe reflejar la fecha de radicación del turno de corrección.
- 14) Se debe suprimir la columna comentario.
- 15) Agregar la columna con Titulo "Motivo Solicitud" en esta columna se debe mostrar la información del campo Descripción de la pantalla de radicación solicitud de corrección.
- 16) Después de la Columna Fase se debe adicionar la columna con titulo "Usuario"; en esta columna se debe mostrar el nombre del usuario del sistema, que tiene asignado el turno en la fase correspondiente.
- 17) Al final del reporte se mostrara un resumen agrupado por fase y/o enviados a otras dependencias, con su respectivo total de turnos de corrección.
- 18) Debe adicionarse un Item de "Total Turnos de Corrección", que corresponde a la sumatoria de turnos que reporta el sistema en el rango de fecha seleccionado.

SE ANEXA FORMATO DEL REPORTE SOLICITADO

SUPERINTENDENCIA DE NOTARIADO Y REGISTRO  
MINISTERIO DE INTERIOR Y DE JUSTICIA  
REPÚBLICA DE COLOMBIA

OFICINA DE REGISTRO DE II.PP DE BUCARAMANGA CÍRCULO REGISTRAL 300  
INFORME DE DOCUMENTOS EN CORRECCIÓN

DEL: 01-02-2011 AL: 16-03-2011  
 Impreso el 16 de Marzo de 2011 a las 16:06:06  
 Usuario que genera reporte: ROSENDO.ROSAS

## Página 1

Radicación	Fecha-Radica	Solicitante	Motivo-Solicitud	Matrícula	Fase	Usuario
2011-999-3-666	29/01/2011	JOSÉ BACCA MELO	CORREGIR NRO. ESCRITURA 546 POR ESCRITURA 546.	999-666666	REVISIÓN Y ANÁLISIS ROMO.HERRERA	

### TOTALES GENERALES

<b>TOTAL CORRECCIONES EN TRAMITE</b>	<b>51</b>
<b>TOTAL CORRECCIONES DESANOTADAS APROBADAS EN ESTE RANGO DE FECHAS</b>	<b>20</b>
<b>TOTAL CORRECCIONES DESANOTADAS NEGADAS EN ESTE RANGO DE FECHAS</b>	<b>3</b>

## Desarrollo

- Script Bases de Datos:
  - SCRIPT\_MANTIS\_8430.sql
- Script Bases de Datos: Procedimiento
  - SP\_Reporte\_39.sql

Para el punto 4 del requerimiento (4) Despues de la Columna Fase se debe adicionar la columna con título “Usuario”; en esta columna se debe mostrar el nombre del usuario del sistema, que tiene asignado el turno en la fase correspondiente) se obtiene de la estación ya que en fases con estaciones compartidas, no se puede determinar el usuario al que está asignado el turno actualmente. Por esta razón se opto por tomar la información del usuario de la estación asignada en la tabla sir.sir\_op\_turno\_estacion.

Actualmente el jasper se encarga de ejecutar la consulta a varias tablas lo cual genera lentitud en la generación del reporte. La metodología implementada para los reportes es ejecutar un procedimiento y almacenar en una tabla, esta tabla será la que se llame del jasper.

Se realiza el desarrollo para la integración del reporte con el procedimiento creado y se realiza la modificación para el llamado desde SIR.

- Reportes:
  - reporte-39.jrxml.
  - SUBREPORT-E03.jrxml.
- Se modificaron los siguientes archivos:  
 AWReportes.java  
 JDOResportesJasperDAO.java  
 ADMINREPORTES\_REPORT39\_PARAMFORM.jsp

### CLASE

**SIR-APP\src\gov\sir\core\web\acciones\administracion\ AWReportes.java**

```
//GCABRERA 19/12/2011
//Se agrega parámetro REPORTE_39_PARAM_USUARIOLOG
```

```
public static final String REPORTE_39_PARAM_USUARIOLOG = "P_USUARIO_LOG";
```

### CLASE

**SIR-APP\src\gov\sir\core\web\acciones\administracion\ AWReportes.java**

#### Metodo:

```
private void do_Reporte39_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {
```

//GCABRERA 19/12/2011

```
//Se agrega parámetro REPORTE_39_PARAM_USUARIOLOG
String param_04 = request.getParameter(REPORTE_39_PARAM_USUARIOLOG);
```

```
//GCABRERA 19/12/2011
//Se agrega parámetro REPORTE_39_PARAM_USUARIOLOG
REPORTE_39_PARAM_PFECHAINI, REPORTE_39_PARAM_PFECHAFIN,
REPORTE_39_PARAM_PCIRCULONOMBRE, REPORTE_39_PARAM_USUARIOLOG
```

```
//GCABRERA 19/12/2011
//Se agrega parámetro REPORTE_39_PARAM_USUARIOLOG
REPORTE_XX_PARAM_CMDKEY, REPORTE_39_PARAM_PFECHAINI,
REPORTE_39_PARAM_PFECHAFIN, REPORTE_39_PARAM_PCIRCULONOMBRE,
REPORTE_39_PARAM_USUARIOLOG
```

### CLASE

#### SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDORreportesJasperDAO.java

```
//GCABRERA 19/12/2011
//Reporte 39
//Se obtienen los parámetros para enviar al procedimiento
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_39_PARAM_PFECHAINI);
listaParametros.add(AWReportes.REPORTE_39_PARAM_PFECHAFIN);
listaParametros.add(AWReportes.REPORTE_39_PARAM_PCIRCULONOMBRE);
mapaReportes.put(AWReportes.REPORTE_39_PARAM_CMDKEY, new Object[]{
    "SELECT SIR.SEQ_Reporte_39.NEXTVAL FROM dual",
    "sp_reporte_39(?, ?, ?, ?)", listaParametros,
    "DELETE FROM sir.sir_op_reporte_39 WHERE sec_proceso = ?});
```

### JSP

#### SIR-APP\web\WEB-

#### INF\jspSeguro\administracion\ADMINREPORTES\_REPORT24\_PARAMFORM.jsp

```
<%-- GCABRERA 19/12/2011
 Se agrega parámetro REPORTE_39_PARAM_USUARIOLOG--%>
<%
    Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
%>
<input type="hidden"
name="<%="AWReportes.REPORTE_39_PARAM_USUARIOLOG%>"
id="<%="AWReportes.REPORTE_39_PARAM_USUARIOLOG%>"
value="<%="usuario.getUsername()%>">
```

### 019 CARATULA-REPARTO NOTARIAL

#### ACTA -CARATULA- REPARTO NOTARIAL

El aplicativo está presentando las siguientes inconsistencias en el módulo de Reparto Notarial:

- En la reimpresión de los recibos de radicación de Minutas como fecha de reimpresión muestra título Reimpreso por Corrección, el cual no corresponde a este proceso, Lo correcto es que muestre la fecha de la impresión y suprima el título incorrecto.

- En la Caratula del Reparto Notarial el Aplicativo, está reflejando el Municipio del Circulo donde se ejecuta el Reparto y no el Municipio al cual corresponde la Minuta .Lo correcto es que muestre el Municipio correcto.

- Cuando se realiza la reimpresión del acta de reparto notarial, el sistema la esta imprimiendo con información diferente a la impresa en el momento de la ejecución del reparto, las dos impresiones deben contener el mismo texto y en este momento el correcto es el que imprime automáticamente en el momento de ejecutar el reparto.

Casos reportados por las ORIP Bucaramanga y Valledupar.

De: Oficina de Registro Bucaramanga [mailto:[ofiregusbucaramanga@supernotariado.gov.co](mailto:ofiregusbucaramanga@supernotariado.gov.co)]

Enviado el: Jueves, 10 de Febrero de 2011 03:22 p.m.

Para: Soporte SIR SNR

CC: Dora Esperanza Pachón, Rodríguez; Martha Ligia Ricaurte Córdoba; Luis Emilio Romero

Mogollón

Asunto: caratulas de reparto de minutos

señores

mesa de ayuda sir

SNR

Cuando una minuta se reparte para el municipio de FLORIDABLANCA, la caratula sale con el municipio de Bucaramanga, y por ninguna parte dice que la notaria asignada corresponde al municipio de Floridablanca, revisar el turno 2011-300-7-40, 2011-300-7-41 y 2011-300-7-44, por favor revisar gracias.

Oficina de Registro de Instrumentos Públicos de Bucaramanga

Teléfono: 6521311 ext. .217

Ing. José Luis García Blanco

**Aclaración:**

Después de analizar el requerimiento con Javier Rocha, se determino que este era lo suficiente claro para realizar los cambios, por lo tanto se desestimo la solicitud de datos realizada por el Ing. Ellery Robles.

**Desarrollo**

- Script Bases de Datos:
  - Ninguno
- Creacion De Imagenes:
  - Ninguno
- Se modificaron los siguientes archivos:
  - Ninguno

**CLASE:**

**gov.sir.core.negocio.acciones.administracion. ANImprimirRepartoNotarial.java**

**Método:**

**private EvnRespImprimirRepartoNotarial imprimirActa(EvnImprimirRepartoNotarial evento) throws EventoException{**

**Código Insertado Linea 885:**

```
/* JAlcazar caso Mantis 0009055: Acta - Requerimiento No 019_151_Caratula - Reparto Notarial
 * Se coloca en la reimpresion del Acta de Reparto Notarial el usuario que ejecuto el proceso. SET en el metodo
setDirectorReparto(funcionarioReparto).
*/
Usuario usuarioreparto = null;
Turno turnominuta = hermod.getTurnobyWF(temp.getIdMinuta());
List historial = turnominuta.getHistorials();
Iterator itera = historial.iterator();
while(itera.hasNext()){
    TurnoHistoria th = (TurnoHistoria) itera.next();
    if(CFase.REP_REPARTO.equals(th.getFase())){
        usuarioreparto = th.getUsuarioAtiende();
    }
}
if (usuarioreparto != null){
    String funcionarioReparto = usuarioreparto.getNombre();
    funcionarioReparto += usuarioreparto.getApellido1()!=null?(" "+ usuarioreparto.getApellido1()):"";
    funcionarioReparto += usuarioreparto.getApellido2()!=null?(" "+ usuarioreparto.getApellido2()):"";
    impActaReparto.setDirectorReparto(funcionarioReparto);
}
```

<b>CLASE:</b>
<b>gov.sir.core.negocio.modelo.imprimible.ImprimibleConstanciaRepartoMinuta.java</b>
<b>Método:</b>
<b>private void imprimirDatosMinuta() {</b>
<b>Cambio Linea 186:</b>
<pre>if (this.circulo!=null &amp;&amp; this.circulo.getOficinaOrigen()!=null &amp;&amp;     this.circulo.getOficinaOrigen().getVereda()!=null &amp;&amp;     this.circulo.getOficinaOrigen().getVereda().getMunicipio()!=null &amp;&amp;     this.circulo.getOficinaOrigen().getVereda().getMunicipio().getNombre()!=null){     municipio = this.circulo.getOficinaOrigen().getVereda().getMunicipio().getIdMunicipio() + " " +     this.circulo.getOficinaOrigen().getVereda().getMunicipio().getNombre();}</pre>
<b>Por:</b>
<pre>/* JAlcazar caso Mantis 0009055: Acta - Requerimiento No 019_151_Caratula - Reparto Notarial  * Se coloca en la la Caratula de Reparto Notarial el municipio al cual corresponde la minuta.  */ if (this.minuta!=null &amp;&amp; this.minuta.getOficinaCategoriaAsignada()!=null &amp;&amp;     this.minuta.getOficinaCategoriaAsignada().getOficinaOrigen()!=null &amp;&amp;     this.minuta.getOficinaCategoriaAsignada().getOficinaOrigen().getVereda()!=null &amp;&amp;     this.minuta.getOficinaCategoriaAsignada().getOficinaOrigen().getVereda().getMunicipio()!=null &amp;&amp;      this.minuta.getOficinaCategoriaAsignada().getOficinaOrigen().getVereda().getMunicipio().getNombre()!=null){     municipio = this.minuta.getOficinaCategoriaAsignada().getOficinaOrigen().getVereda().getMunicipio().getIdMunicipio() + " " + this.minuta.getOficinaCategoriaAsignada().getOficinaOrigen().getVereda().getMunicipio().getNombre();}</pre>
<b>CLASE:</b>
<b>gov.sir.core.negocio.modelo.imprimibles.ImprimibleRecibo.java</b>
<b>Método:</b>
<b>protected void makeNewPage() {</b>
<b>Cambio Linea 1755:</b>
<pre>if (this.solicitud instanceof SolicitudRepartoNotarial &amp;&amp; this.isReimpresion()) {     this.imprimirLinea(ImprimibleConstantes.PLANO,         (int)(ImprimibleConstantes.MARGEN_IZQ * 5.0d), "Reimpreso por Corrección"); }</pre>
<b>Por:</b>
<pre>/* JAlcazar caso Mantis 0009055: Acta - Requerimiento No 019_151_Caratula - Reparto Notarial  * Se elimino el mensaje "Reimpreso por Corrección" en la reimpresión de los recibos de radicación de minutos.  */</pre>
<b>023 Error en los parametros de busqueda para la reimpresión</b>
<b>Requerimiento GLPI No.27928</b>
Turno: 2011-234-2-2271
Motivo Inconsistencia: Se comunica la funcionaria Sandra Hernández para reportar que al realizar la reimpresión de la Constancia de No Propiedad aparece un número de documento diferente al que salió impreso en la primera impresión.
Se realizan pruebas en el aplicativo y al realizar la reimpresión de la constancia de no propiedad aparecen los datos del Solicitante, favor verificar ya que el sistema no está tomando los parámetros de búsqueda para la reimpresión. Se anexan imágenes.
Grupo: Puerto López
<b>Desarrollo</b>
<ul style="list-style-type: none"> <li>➤ Script Bases de Datos: Ninguno</li> <li>➤ Creacion De Imagenes: Ninguno</li> <li>➤ Se modificaron los siguientes archivos: Ninguno</li> </ul>
<b>CLASE:</b>
<b>gov.sir.core.negocio.acciones.administracion.ANImpimirFolio.java</b>

<p><b>Método:</b></p> <pre><b>private EvnRespImprimirFolio reImprimirConsulta(EvnImprimirFolio evento) throws EventoException</b></pre> <p><b>Cambio Linea 184:</b></p> <pre>//Busqueda busqueda = (Busqueda)solConsulta.getBusquedas().get(0);  Ciudadano ciudadano = new Ciudadano(); ciudadano.setTipoDoc(solConsulta.getCiudadano().getTipoDoc()); ciudadano.setDocumento(solConsulta.getCiudadano().getDocumento()); ciudadano.setNombre(solConsulta.getCiudadano().getNombre()); ciudadano.setApellido1(solConsulta.getCiudadano().getApellido1()); ciudadano.setApellido2(solConsulta.getCiudadano().getApellido2());</pre> <p><b>Por:</b></p> <ul style="list-style-type: none"> <li>* @author: Diana Lora</li> <li>* @change: Caso Mantis 8463: Acta - Requerimiento No 023_151 - Error en los parametros</li> <li>* de busqueda para la reimpresión. Cancela el comentario de Fernando Padilla.</li> <li>*</li> <li>**/</li> </ul> <pre>Busqueda busqueda = (Busqueda)solConsulta.getBusquedas().get(0);  Ciudadano ciudadano = new Ciudadano(); ciudadano.setTipoDoc(busqueda.getTipoDocCiudadano()); ciudadano.setDocumento(busqueda.getNumeroDocCiudadano()); ciudadano.setNombre(busqueda.getNombreCiudadano()); ciudadano.setApellido1(busqueda.getApellido1Ciudadano()); ciudadano.setApellido2(busqueda.getApellido2Ciudadano());</pre>		
<b>039_151_Error en salvedades del folio (Informativo)</b>		
<p>Se han reportado turnos de correcciones que están presentando el siguiente error al tratar de avanzar de la fase Revisar y Aprobar.</p> 		
<p>SNR :: Superintendencia de Notariado y Registro :: Ministerio del interior y de Justicia :: República de Colombia  <a href="http://192.168.202.9:8100/SNR/correcion.do">http://192.168.202.9:8100/SNR/correcion.do</a>      Recorte de pantalla realizado: 07/09/2011, 12:14 p.m.</p> <p>El error se presenta porque hay folios con salvedades que no tienen actualizada la columna flio_last_id_salvedad. Por esta razón al realizar correcciones sobre el folio el id no coincide con el valor real de salvedades.</p> <p>Se identificó que el problema se presenta para los folios relacionados en el proceso de interrelación que tienen null en la columna flio_last_id_salvedad.</p> <p>En el proceso se interrelacionaron <b>1197812</b> folios, de los cuales <b>1095698</b> quedaron con el id consistente y <b>102114</b> quedaron inconsistentes.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;"><b>Desarrollo</b></td> <td style="width: 70%; text-align: right;">Crear un procedimiento almacenado a nivel de base de datos para actualizar los 102114 folios</td> </tr> </table>	<b>Desarrollo</b>	Crear un procedimiento almacenado a nivel de base de datos para actualizar los 102114 folios
<b>Desarrollo</b>	Crear un procedimiento almacenado a nivel de base de datos para actualizar los 102114 folios	

inconsistentes. Se debe corregir la columna flio\_last\_id\_salvedad de la tabla SIR\_NE\_FOLIO para que quede consistente de acuerdo al número de salvedades. (SIR\_NE\_SALVEDAD\_FO)

Adjunto el archivo SP\_UPDATE\_FO\_SALVEDAD.sql, el cual contiene el código fuente del procedimiento almacenado SP\_UPDATE\_FO\_SALVEDAD.

El error se presenta porque hay folios con salvedades que no tienen actualizada la columna flio\_last\_id\_salvedad. Por esta razón al realizar correcciones sobre el folio el id no coincide con el valor real de salvedades.

Se identificó que el problema se presenta para los folios relacionados en el proceso de interrelación que tienen null en la columna flio\_last\_id\_salvedad.

En el proceso se interrelacionaron 1197812 folios, de los cuales 1095698 quedaron con el id consistente y 102114 quedaron inconsistentes.

Este script se ejecuto el día 19-09-2011 con el RFC RFC\_actualización procedimientos almacenados del esquema SIR.docx

#### **040 Impresión de Folios - Nivel Central**

Actualmente la funcionalidad de impresión simple de folio, permite generar en un archivo con extensión .pdf, la información inmobiliaria de un predio, para lo cual se digita el número de matrícula (CIRCULO—MATRICULA), este procedimiento es dispendioso en los casos en que las diferentes entidades requieran información de muchas matrículas.

Por lo anterior, la SNR requiere que se puedan generar estos archivos (con extensión pdf), cargando la información de las matrículas a partir de un archivo Excel.

La ejecución de dicha implementación, será realizada en ambiente de producción por el administrador de la base de datos del SIR y se debe crear el mecanismo para que lo ejecute el personal de la SNR en la base de datos del Datacenter Alterno, a esta funcionalidad no podrá acceder ningún usuario de los Círculos Registrales.

#### **Validaciones a realizar:**

De acuerdo a la cantidad de folios de matrícula, ésta implementación deberá ser ejecutada de tal manera que no se afecte la prestación del servicio registral.

Se deberá indicar la cantidad de archivos .pdf que se puede generar diariamente, con el fin de poder dimensionar y controlar el número de solicitudes realizadas por los entes internos y externos.

Las que considere necesarias el contratista (si considera necesario, socializarlas con la SNR tanto para la).

#### **Desarrollo**

Se solicitó a través de correo y mesa de ayuda a la SNR especificar el rol a asignar a la pantalla Impresión Simple de folio para habilitar la sección de impresión masiva de folio; al no recibir respuesta se asignó el rol SIR\_ROL\_IMPRESION\_FOLIO\_ESPECIAL teniendo en cuenta que se realizaran pruebas para la versión 2.7.4; este rol podrá ser modificado en las pruebas lo cual debe quedar estipulado en el acta y se deberán asignar a desarrollo.

- Script Bases de Datos:
  - **Script\_mantis\_9422.sql.**
- Creacion De Imagenes:
  - Ninguno.
- Se crearon clases nuevas en SIR:
  - Ninguno.
- Se creo un proyecto nuevo:
  - **EJBTareaSirCliente.**
- Se modificaron los siguientes archivos en SIR:
  - WEB-INF\jspSeguro\administracion\IMPRIMIR\_FOLIO.jsp
  - sir.core.web.acciones.administracion.AWImpresionFolio.java

- sir.core.eventos.administracion.EvnImprimirFolio.java
- sir.core.negocio.acciones.administracion.ANImpimirFolio.java
- sir.core.eventos.administracion.EvnRespImprimirFolio.java
- sir.core.web.navegacion.administracion.ControlNavegacionImpresionFolio.java
- Archivo de propiedades:**gov.sir.print.properties**
- src\org.auriga.smart.web.xml
- gov.sir.print.server.PrintJobsProperties.java
- Se crearon clases nuevas en GeneralSIR:
  - co.com.iridium.generalSIR.dao.TareasSIRDAO.java
  - co.com.iridium.generalSIR.negocio.tareas.TareasSIR.java

#### ARCHIVO JSP:

WEB-INF\jspSeguro\administracion\IMPRIMIR\_FOLIO.jsp

#### Código Insertado Línea 1 - 8:

```
<%@page import="gov.sir.core.negocio.modelo.Circulo"%>
<%@page import="gov.sir.core.negocio.modelo.constantes.CRoles"%>
<%@page import="java.util.List"%>
<%@page import="java.util.Iterator"%>
<%@page import="java.util.ArrayList"%>
<%@page import="gov.sir.core.web.helpers.comun.TablaMatriculaHelper"%>
<%@page import="gov.sir.core.negocio.modelo.constantes.CTipoTarifa"%>
<%@page import="org.auriga.core.modelo.transferObjects.Rol"%>
```

#### Código Insertado Línea 15:

```
<script type="text/javascript" src="<% =request.getContextPath()%>/jsp/plantillas/calendario.js"/>
```

#### Código Insertado Línea 42 - 60:

```
String rtaTarea = "";
String imp_masiva_simple_folio_ftp = "";
if(session.getAttribute(AWImpresionFolio.PROGRAMAR_TAREA_EXITO) != null) {
    if(session.getAttribute(AWImpresionFolio.PROGRAMAR_TAREA_EXITO).equals("true")) {
        rtaTarea = "true";
        imp_masiva_simple_folio_ftp =
    }
    (String)session.getAttribute(AWImpresionFolio.IMP_MASIVA_SIMPLE_FOLIO_FTP);
}

if(session.getAttribute(AWImpresionFolio.PROGRAMAR_TAREA_EXITO).equals("existeTarea")) {
    rtaTarea = "existeTarea";
}

if(session.getAttribute(AWImpresionFolio.PROGRAMAR_TAREA_EXITO).equals("fechaInvalida")) {
    rtaTarea = "fechaInvalida";
}
session.removeAttribute(AWImpresionFolio.PROGRAMAR_TAREA_EXITO);
session.removeAttribute(AWImpresionFolio.IMP_MASIVA_SIMPLE_FOLIO_FTP);
}
```

#### Código Insertado Línea 190 - 243:

```
<%
    /**
     * @author : Ellery David Robles Gómez.
     * @change : Se determina si el usuario tiene el rol SIR_ROL_IMPRESION_FOLIO_ESPECIAL para
     * permitirle programar tareas.
     * Caso_Mantis : 09422: Acta - Requerimiento No 040_151 - Impresión de Folios - Nivel Central.
     */
    Rol rol = null;
    boolean folioimpresioespecial = false;
    List roles = (List)session.getAttribute(WebKeys.LISTA_ROLES);
    Iterator itRoles = roles.iterator();
    while (itRoles.hasNext()) {
        rol = (Rol)itRoles.next();
        if (rol.getRoleId().equals(CRoles.SIR_ROL_IMPRESION_FOLIO_ESPECIAL))
            folioimpresioespecial = true;
    }
    if (folioimpresioespecial) { %>
        <form name="form1" action="impresionFolio.do" method="post">
            <table width="100%" class="camposform">
                <% if(rtaTarea.equals("true")) { %>
                    <tr>
                        <td width="20"></td>
```

```

        <td>Se realizó satisfactoriamente la programación de la tarea, descargar PDFs en : <%
imp_masiva_simple_folio_ftp %></td>
        </tr>
        <% } %>
        <% if(rtaTarea.equals("existeTarea")) { %>
        <tr>
            <td width="20"></td>
            <td style="color: red">El nombre de la tarea ya existe, inténtelo nuevamente.</td>
        </tr>
        <% } %>
        <% if(rtaTarea.equals("fechaInvalida")) { %>
        <tr>
            <td width="20"></td>
            <td style="color: red">La fecha seleccionada es inválida, inténtelo nuevamente.</td>
        </tr>
        <% } %>
    </table>
    <table width="100%" class="camposform">
        <tr>
            <td></td>
            <td>Programar tarea impresión masiva simple de folios</td>
        </tr>
        <tr>
            <td width="20">&nbsp;</td>
            <td>
                <table width="100%">
                    <tr>
                        <td width="100%">
                            <table width="100%" border="0" cellpadding="0" cellspacing="2"
class="camposform">
                                <tr>
                                    <td align="left" width="25%">
                                        Nombre:&nbsp;&nbsp;
                                        <input type="hidden" name="<%=>WebKeys.ACCION %>" value="<%=>
AWImpresionFolio.PROGRAMAR_TAREA %>"/>
                                        <input name="nombreTarea" type="text" class="camposformtext"/>

```

**Código Insertado Línea 245 - 303:**

```

<td align="left" width="25%">
            <table border="0" cellpadding="0" cellspacing="2" class="camposform"
style="border: none">
                <tr>
                    <td align="left">
                        Fecha Ejecución:&nbsp;&nbsp;
                        <% try {
                            textHelper.setNombre("fechaTarea");
                            textHelper.setCssClase("camposformtext");
                            textHelper.setId("fechaTarea");
                            textHelper.render(request,out);
                        } catch(HelperException re) {
                            out.println("ERROR " + re.getMessage());
                        }
                    %>
                    </td>
                    <td align="left">
                        <a href="javascript:NewCal('fechaTarea','ddmmmyyyy',true,24)">request.getContextPath()%>')"/></a>
                        </td>
                    </td>
                    <td align="left">Hora:&nbsp;&nbsp;
                        <select name="hora" class="camposformtext">
                            <option value="20">8 PM</option>
                            <option value="21">9 PM</option>
                            <option value="22">10 PM</option>
                            <option value="23">11 PM</option>
                            <option value="0">12 AM</option>
                            <option value="1">1 AM</option>

```

```

        <option value="2">2 AM</option>
        <option value="3">3 AM</option>
        <option value="4">4 AM</option>
        <option value="5">5 AM</option>
    </select>
</td>
<td align="left" width="30%">Seleccione archivo csv:&nbsp;&nbsp;
    <input name="filename" type="file" class="camposformtext"/>
</td>
</tr>
</table>
</td>
</tr>
<tr>
<td width="100%">
    <table border="0" align="right" cellpadding="0" cellspacing="2" class="camposform">
        <tr>
            <td width="20">/jsp/images/ind_vinculo.gif" width="20" height="15"/></td>
            <td>Programar Tarea</td>
            <td><input name="imageField222" type="image"
src="<%="request.getContextPath()%>/jsp/images/btn_short_anadir.gif" width="35" height="25" border="0"/></td>
        <tr>
            <td></td>
            </tr>
        </table>
        </td>
        </tr>
    </table>
    </td>
</tr>
</table>
</form>
<% } %>

```

## Código Modificado Línea 126 - 127:

```
<!--<form action="impresionFolio.do" method="POST" name="BUSCAR" id="BUSCAR">
    <input type="hidden" name="%= WebKeys.ACION %>" id="%= AWImpresionFolio.IMPRIMIR_FOLIO%" value="%= AWImpresionFolio.IMPRIMIR_FOLIO %>"-->
```

## Código Modificado Linea 132 - 135:

```
<td valign="top" bgcolor="#79849B" class="tdtablacentral">
    <form action="impresionFolio.do" method="POST" name="BUSCAR" id="BUSCAR">
        <input type="hidden" name="<% WebKeys.ACCTION %>" id="<%=
AWImpresionFolio.IMPRIMIR_FOLIO%>" value="<%=
AWImpresionFolio.IMPRIMIR_FOLIO %>">
        <table width="100%" border="0" cellpadding="0" cellspacing="0">
```

## ARCHIVO JAVA:

sir.core.web.acciones.administracion.AWImpresionFolio.java

## Código Insertado Línea 29:

```
import com.csvreader.CsvReader;
```

**Código Insertado Linea 91 - 102:**

```
/** Constante que identifica el tipo de tarea a programar */
public static final String IMP_MASIVA_SIMPLE_FOLIO = "IMP_MASIVA_SIMPLE_FOLIO";

/** Constante que identifica la acción de programar tarea */
public static final String PROGRAMAR_TAREA = "PROGRAMAR_TAREA";

/** Constante que identifica la acción de programar tarea */
public static final String PROGRAMAR_TAREA_EXITO = "PROGRAMAR_TAREA_EXITO";

/** Constante que identifica la acción de programar tarea */
public static final String IMP_MASIVA_SIMPLE_FOLIO_FTP = "IMP_MASIVA_SIMPLE_FOLIO_FTP";
```

## Código Insertado Linea 137 - 138:

```
} else if (accion.equals(AWImpresionFolio.PROGRAMAR_TAREA)) {  
    return programarTarea(request);
```

### **Código Insertado Linea 520 - 587:**

```
/**  
 * @author : Ellery David Robles Gómez.  
 * @change : Metodo que se encarga de leer el archivo csv con sus folios y cargarlos a un StringBuffer.  
 * Caso_Mantis : 09422: Acta - Requerimiento No 040_151 - Impresión de Folios - Nivel Central.  
 */
```

```

private EvnImprimirFolio programarTarea(HttpServletRequest request) throws AccionWebException {
    HttpSession session = request.getSession();
    ImprimirFolioException exception = new ImprimirFolioException();
    org.auriga.core.modelo.transferObjects.Usuario usuarioAuriga = (org.auriga.core.modelo.transferObjects.Usuario)
    session.getAttribute(SMARTKeys.USUARIO_EN_SESION);
    StringBuffer folios = null;
    String nombreTarea=null, fechaTarea=null, dirOrigen=null;
    try {
        if (request.getParameter("nombreTarea").isEmpty()) {
            exception.addError("Debe digitar un número de matrícula.");
        } else {
            nombreTarea = request.getParameter("nombreTarea");
        }
        if (request.getParameter("fechaTarea").isEmpty() || request.getParameter("hora").isEmpty()) {
            exception.addError("Debe digitar la fecha de ejecución de la tarea.");
        } else {
            fechaTarea = request.getParameter("fechaTarea")+" "+request.getParameter("hora");
        }
        if (request.getParameter("filename").isEmpty()) {
            exception.addError("Debe seleccionar un archivo csv.");
        } else {
            dirOrigen = "C:\\sir_tareas\\\"+request.getParameter("filename");
        }
        if (exception.getErrores().isEmpty()) {
            dirOrigen = dirOrigen.replace('\\','/');
        }
        if (!dirOrigen.endsWith(".csv")) {
            exception.addError("El archivo debe tener extensión (csv).");
        } else {
            // instanciar el objeto readerCSV
            CsvReader reader = new CsvReader(dirOrigen);

            // asigno separador de valores punto y coma, si no lo cambian queda por defecto la coma
            reader.setDelimiter(';');

            // instanciamos folios en un buffer de string;
            folios = new StringBuffer();

            // recorremos las filas del fichero
            while (reader.readRecord()) {
                if (reader.get(0) != null && reader.get(1) != null) {
                    folios.append(reader.get(0)).append("-").append(reader.get(1)).append(";");
                }

                if (reader.get(2) != null && reader.get(3) != null) {
                    folios.append(reader.get(2)).append("-").append(reader.get(3)).append(";");
                }

                if (reader.get(4) != null && reader.get(5) != null) {
                    folios.append(reader.get(4)).append("-").append(reader.get(5)).append(";");
                }
            }
            reader.close();
        }
    } catch (Exception ex) {
        exception.addError("El archivo no existe.");
        if (exception.getErrores().size() > 0) { throw exception; }
    }
    if (exception.getErrores().size() > 0) { throw exception; }

    return new EvnImprimirFolio(usuarioAuriga, nombreTarea, fechaTarea, folios,
    AWImpresionFolio.IMP_MASIVA_SIMPLE_FOLIO, EvnImprimirFolio.PROGRAMAR_TAREA);
}

```

**Código Insertado Línea 647 - 650:**

```

} else if(tipoEvento.equals(EvnRespImprimirFolio.PROGRAMAR_TAREA_OK)){
    session.setAttribute(AWImpresionFolio.PROGRAMAR_TAREA_EXITO, respuesta.getExito());
    session.setAttribute(AWImpresionFolio.IMP_MASIVA_SIMPLE_FOLIO_FTP,
    respuesta.getImp_masiva_simple_folio_ftp());
    return;
}

```

**Código Modificado Línea 495 - 496:**

```
session.removeAttribute(AWIImpresionFolio.PROGRAMAR_TAREA_EXITO);
session.removeAttribute(AWIImpresionFolio.IMP_MASIVA_SIMPLE_FOLIO_FTP);
```

ARCHIVO JAVA:

sir.core.eventos.administracion.EvnImprimirFolio.java

**Código Insertado Línea 43 - 46:**

```
/** Constante que identifica la acción de programar tarea */
public static final String PROGRAMAR_TAREA = "PROGRAMAR_TAREA";
```

**Código Insertado Línea 92 - 101:**

```
private String nombreTarea;
private String fechaTarea;
private String urlArchivo;
private String tipoTarea;
private StringBuffer folios;
```

**Código Insertado Línea 126 - 137:**

```
* @param usuario
* @param tipoEvento
*/
public EvnImprimirFolio(Usuario usuario, String nombreTarea, String fechaTarea, StringBuffer folios, String tipoTarea,
String tipoEvento) {
    super(usuario, tipoEvento);
    this.nombreTarea = nombreTarea;
    this.fechaTarea = fechaTarea;
    this.tipoTarea = tipoTarea;
    this.folios = folios;
}
```

**Código Insertado Línea 323 - 325:**

```
public String getFechaTarea() {
    return fechaTarea;
```

**Código Insertado Línea 327 - 363:**

```
public void setFechaTarea(String fechaTarea) {
    this.fechaTarea = fechaTarea;
}

public StringBuffer getFolios() {
    return folios;
}

public void setFolios(StringBuffer folios) {
    this.folios = folios;
}

public String getNombreTarea() {
    return nombreTarea;
}

public void setNombreTarea(String nombreTarea) {
    this.nombreTarea = nombreTarea;
}

public String getTipoTarea() {
    return tipoTarea;
}

public void setTipoTarea(String tipoTarea) {
    this.tipoTarea = tipoTarea;
}
```

```

public String getUrlArchivo() {
    return urlArchivo;
}

public void setUrlArchivo(String urlArchivo) {
    this.urlArchivo = urlArchivo;
}
}

```

#### ARCHIVO JAVA:

sir.core.negocio.acciones.administracion.ANIImprimirFolio.java

#### Código Insertado Línea 113 - 117:

```

import co.com.iridium.generalSIR.comun.GeneralSIRException;
import co.com.iridium.generalSIR.negocio.tareas.TareasSIR;
import gov.sir.print.server.PrintJobsProperties;
import java.util.logging.Level;
import java.util.logging.Logger;

```

#### Código Insertado Línea 248 - 249:

```

} else if (tipoEvento.equals(EvnImprimirFolio.PROGRAMAR_TAREA)) {
    return programarTarea(evento);
}

```

#### Código Insertado Línea 2444 - 2450:

```

private EvnRespImprimirFolio programarTarea(EvnImprimirFolio evento) throws EventoException {
    TareasSIR tarea = new TareasSIR();
    String exito = null;
    try {
        exito = tarea.programarTareas(evento.getNombreTarea(), evento.getFechaTarea(), evento.getFolios(),
            evento.getTipoTarea());
    } catch (GeneralSIRException ex) {
        Logger.getLogger(ANIImprimirFolio.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```

#### Código Insertado Línea 2452 - 2458:

```

PrintJobsProperties prop = PrintJobsProperties.getInstance();
EvnRespImprimirFolio evRespuesta = new EvnRespImprimirFolio(null,
EvnRespImprimirFolio.PROGRAMAR_TAREA_OK);
evRespuesta.setExito(exito);

evRespuesta.setImp_masiva_simple_folio_ftp(prop.getProperty(PrintJobsProperties.IMP_MASIVA_SIMPLE_FOLIO_FTP)+e
vento.getNombreTarea()+"/");
return evRespuesta;
}
}

```

#### ARCHIVO JAVA:

sir.core.eventos.administracion.EvnRespImprimirFolio.java

#### Código Insertado Línea 37 - 42:

```

/** Constante que identifica la acción programar tarea */
public static final String PROGRAMAR_TAREA_OK = "PROGRAMAR_TAREA_OK";

String exito;

```

String imp\_masiva\_simple\_folio\_ftp;

#### Código Insertado Línea 71 - 95:

```

/**
 * @param payload
 * @param tipoEvento
 */
public EvnRespImprimirFolio(Object payload, String exito, String tipoEvento) {
    super(payload, tipoEvento);
    this.exito = exito;
}

public String getExito() {
    return exito;
}

public void setExito(String exito) {
    this.exito = exito;
}

public String getImp_masiva_simple_folio_ftp() {
    return imp_masiva_simple_folio_ftp;
}

```

```

    }

    public void setImp_masiva_simple_folio_ftp(String imp_masiva_simple_folio_ftp) {
        this.imp_masiva_simple_folio_ftp = imp_masiva_simple_folio_ftp;
    }

```

#### ARCHIVO JAVA:

sir.core.web.navegacion.administracion.ControlNavegacionImpresionFolio.java

#### Código Insertado Línea 29 - 30:

```
public static final String PROGRAMAR_TAREA_OK = "PROGRAMAR_TAREA_OK";
```

#### Código Insertado Línea 85 - 86:

```
} else if (accion.equals(AWImpresionFolio.PROGRAMAR_TAREA)) {
    return PROGRAMAR_TAREA_OK;
```

#### ARCHIVO PROPERTIES: gov.sir.print.properties

#### Código Insertado Línea 48 - 50:

```
#IP FTP IMP_MASIVA_SIMPLE_FOLIO
imp_masiva_simple_folio.ftp=ftp://192.168.202.26/sir_tareas/imp_masiva_simple_folio/
```

#### ARCHIVO XML:

src\org.auriga.smart.web.xml

#### Código Insertado Línea 4802 - 4805:

```
<navegacion>
    <resultado>PROGRAMAR_TAREA_OK</resultado>
    <url-redireccion>admin.imprimirfolio.view</url-redireccion>
</navegacion>
```

#### ARCHIVO JAVA:

gov.sir.print.server.PrintJobsProperties.java

#### Código Insertado Línea 57 - 63:

```
/**
 * @author : Ellery David Robles Gómez
 * @change : Constante para obtener la IP del servidor FTP.
 * Caso_Mantis : 09422: Acta - Requerimiento No 040_151 - Impresión de Folios - Nivel Central.
 */
public static final String IMP_MASIVA_SIMPLE_FOLIO_FTP ="imp_masiva_simple_folio.ftp";
```

#### Modificaciones 2

#### ARCHIVO JSP:

WEB-INF\jspSeguro\administracion\IMPRIMIR\_FOLIO.jsp

#### Código Insertado Línea 207 - 249:

```
<table width="100%" class="camposform">
    <tr>
        <td></td>
        <td>Programar tarea impresión masiva simple de folios</td>
    </tr>
    <tr>
        <td width="20">&nbsp;</td>
        <td>
            <table width="100%" class="camposform">
                <tr>
                    <td></td>
                    <td>Seleccione archivo csv:</td>
                </tr>
                <tr>
                    <td width="20">&nbsp;</td>
                    <td>
                        <table width="100%">
                            <tr>
                                <td width="81%">
                                    <input name="filename" type="file" class="camposformtext"/>
                                </td>
                                <td width="19%">
                                    <table border="0" align="right" cellpadding="0" cellspacing="2">
```

## Código Insertado Línea 325:

```
<input name="listaFolios" type="hidden" class="camposformtext" value="<%=\nrequest.getSession().getAttribute(AWImpresionFolio.LISTA_MATRICULAS) %>">/
```

**Código Insertado Linea 347:**

<% } %>

## Código Modificado Línea 206:

```
<form name="form1" enctype="multipart/form-data" method="post" action="impresionFolio.do">
```

#### **Código Modificado Línea 323 - 324:**

-000.NOMBRE\_ARCHIVO

sir.core.web.acciones.administracion.AWImpresionFolio.java

### Código Eliminado Línea 29:

Código Insertado Línea 30 - 36:

```
Código Insertado Línea 36 - 36.
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.util.ArrayList;
import java.util.Iterator;
import org.apache.commons.fileupload.DiskFileUpload;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.FileUpload;
```

**Código Insertado Línea 98 - 106:**

```
        } else if (accion.equals(AWIMpresionFolio.PROGRAMAR_TAREA)) {      /** Constante que identifica que se van agregar  
matriculas desde archivo */  
        public final static String AGREGAR_DE_ARCHIVO = "AGREGAR_DE_ARCHIVO";  
  
        /** Constante que identifica que se van nombre del archivo cargado */  
        public final static String NOMBRE_ARCHIVO_CSV = "NOMBRE_ARCHIVO_CSV";  
  
        /** Constante que identifica el listado de matriculas cargadas desde archivo csv */  
        public final static String LISTA_MATRICULAS = "LISTA_MATRICULAS";
```

Código Insertado Línea 109 - 124:

```
private static final String CONTENT_TYPE = "application/vnd.ms-excel";
```

```
private String nombreArchivo;
```

```
private StringBuffer listaFolios;
```

## Código Insertado Linea 132 - 137:

```
boolean isMultipart = FileUpload.isMultipartContent(request);
```

```
    if ( isMultipart ) {  
        return UploadFile(request);
```

```

        }
        if ((accion == null) || (accion.trim().length() == 0)) {
Código Insertado Línea 51 - 642:
private EvnImprimirFolio UploadFile(HttpServletRequest request) throws AccionWebException {
    HttpSession session = request.getSession();
    ImprimirFolioException exception = new ImprimirFolioException();
    org.auriga.core.modelo.transferObjects.Usuario usuarioAuriga = (org.auriga.core.modelo.transferObjects.Usuario)
    session.getAttribute(SMARTKeys.USUARIO_EN_SESION);
    String nombreArchivo = null;
    StringBuffer listaFolios = null;

    boolean errorCopia = false, errorFormato = false, errorMatriculasInvalidas = false;
    List lstCopia = new ArrayList(), lstFormato = new ArrayList();

    try {

        DiskFileUpload upload = new DiskFileUpload();
        List list = upload.parseRequest(request);
        Iterator it = list.iterator();
        while ( it.hasNext() ) {
            FileItem fi = (FileItem) it.next();

            if ( !fi.isFormField() ) {
                nombreArchivo = fi.getName();

                if ( !nombreArchivo.endsWith(".csv") ) {
                    exception.addError("El archivo debe tener extensión .csv, y el
procesado no corresponde: "+nombreArchivo);
                } else {
                    if ( fi.getContentType().equals(CONTENT_TYPE) ) {
                        BufferedReader in = new BufferedReader(new InputStreamReader(fi.getInputStream()));
                        String record = "", recArray[] = null, strMat = "";

                        // instanciamos folios en un buffer de string;
                        listaFolios = new StringBuffer();

                        while((record = in.readLine()) != null) {
                            recArray = record.split(";");
                            if ( recArray.length < 2 ) {
                                lstFormato.add(record);
                                errorFormato = true;
                            } else {
                                strMat = recArray[0].trim()+"-"+recArray[1].trim();
                                if ( recArray[0] != null && recArray[1] != null ) {
                                    listaFolios.append(recArray[0].trim()).append("-"
                                ").append(recArray[1].trim()).append(";");
                                }

                                if ( recArray[2] != null && recArray[3] != null ) {
                                    listaFolios.append(recArray[2].trim()).append("-"
                                ").append(recArray[3].trim()).append(";");
                                }

                                if ( recArray[4] != null && recArray[5] != null ) {
                                    listaFolios.append(recArray[4].trim()).append("-"
                                ").append(recArray[5].trim()).append(";");
                                }
                            }
                        }
                    }
                }
            }
        }

        String r = "C:\sir_tareas\""+nombreArchivo;
        // instanciar el objeto readerCSV
        CsvReader reader = new CsvReader(r);
        System.out.print("Línea de error: "+r);
        // asigno separador de valores punto y coma, si no lo cambian queda por defecto la coma
        reader.setDelimiter('.');

        // instanciamos folios en un buffer de string;
    }
}

```

```

//         listaFolios = new StringBuffer();
//
//         // recorremos las filas del fichero
//         while (reader.readRecord()) {
//             if (reader.get(0) != null && reader.get(1) != null) {
//                 listaFolios.append(reader.get(0)).append("-").append(reader.get(1)).append(";");
//             }
//
//             if (reader.get(2) != null && reader.get(3) != null) {
//                 listaFolios.append(reader.get(2)).append("-").append(reader.get(3)).append(";");
//             }
//
//             if (reader.get(4) != null && reader.get(5) != null) {
//                 listaFolios.append(reader.get(4)).append("-").append(reader.get(5)).append(";");
//             }
//
//             reader.close();
//         } catch (Exception ex) {
//             exception.addError("El archivo no existe.");
//             if (exception.getErrores().size() > 0) { throw exception; }
//         }
//     }
//     return new EvnImprimirFolio(usuarioAuriga, listaFolios, nombreArchivo,
AWImpresionFolio.AGREGAR_DE_ARCHIVO);
}

/**
 * @author : Ellery David Robles Gómez.
 * @change : Metodo que se encarga de leer el archivo csv con sus folios y cargarlos a un StringBuffer.
 * Caso_Mantis : 09422: Acta - Requerimiento No 040_151 - Impresión de Folios - Nivel Central.
 */

```

#### **Código Modificado Línea 130:**

String accion = request.getParameter(WebKeys.ACION);

#### **Código Modificado Línea 647 - 650:**

```

String nombreTarea=null, fechaTarea=null;
StringBuffer listaFolios2 = null;

```

```
//     try {
```

#### **Código Modificado Línea 661 - 665:**

```

listaFolios2 = (StringBuffer) session.getAttribute(AWImpresionFolio.LISTA_MATRICULAS);
//     } catch (Exception ex) {
//         exception.addError("El archivo no existe.");
//         if (exception.getErrores().size() > 0) { throw exception; }
//     }

```

#### **Código Modificado Línea 668:**

```

return new EvnImprimirFolio(usuarioAuriga, nombreTarea, fechaTarea, listaFolios2,
AWImpresionFolio.IMP_MASIVA_SIMPLE_FOLIO, EvnImprimirFolio.PROGRAMAR_TAREA);

```

#### **Código Modificado Línea 729 - 731:**

```

} else if(tipoEvento.equals(EvnRespImprimirFolio.AGREGAR_DE_ARCHIVO)){
    session.setAttribute(AWImpresionFolio.NOMBRE_ARCHIVO_CSV, respuesta.getNombreArchivo());
    session.setAttribute(AWImpresionFolio.LISTA_MATRICULAS, respuesta.getListadoFolios());

```

### ARCHIVO JAVA:

sir.core.eventos.administracion.EvnImprimirFolio.java

#### **Código Insertado Línea 45 - 47:**

```
public static final String AGREGAR_DE_ARCHIVO = "AGREGAR_DE_ARCHIVO";
```

```
/** Constante que identifica la acción de programar tarea */
```

#### **Código Insertado Línea 105 - 106:**

```
private StringBuffer listaFolios;
```

#### **Código Insertado Línea 132 - 133:**

```
* @param nombreArchivo
* @param listaFolios
```

#### **Código Insertado Línea 138 - 147:**

```
this.nombreArchivo = nombreArchivo;
this.listaFolios = listaFolios;
}
```

```

    /**
     * @param usuario
     * @param tipoEvento
     */
    public EvnImprimirFolio(Usuario usuario, String nombreTarea, String fechaTarea, StringBuffer listaFolios, String
    tipoTarea, String tipoEvento) {
        super(usuario, tipoEvento);

```

**Código Insertado Línea 357 - 364:**

```

    public StringBuffer getListaFolios() {
        return listaFolios;
    }

    public void setListaFolios(StringBuffer listaFolios) {
        this.listaFolios = listaFolios;
    }

```

**Código Modificado Línea 103:**

```
    private String nombreArchivo;
```

**Código Modificado Línea 136:**

```
public EvnImprimirFolio(Usuario usuario, StringBuffer listaFolios, String nombreArchvio, String tipoEvento) {
```

**Código Modificado Línea 151:**

```
this.listaFolios = listaFolios;
```

**Código Modificado Línea 349 - 350:**

```
public String getNombreArchivo() {
    return nombreArchivo;
}
```

**Código Modificado Línea 353 - 354:**

```
public void setNombreArchivo(String nombreArchivo) {
    this.nombreArchivo = nombreArchivo;
}
```

ARCHIVO JAVA:

sir.core.negocio.acciones.administracion.ANImpimirFolio.java

**Código Insertado Línea 248 - 249:**

```
} else if (tipoEvento.equals(EvnImprimirFolio.AGREGAR_DE_ARCHIVO)) {
    return UploadFile(evento);
```

**Código Insertado Línea 2446 - 2455:**

```
private EvnRespImprimirFolio UploadFile(EvnImprimirFolio evento) throws EventoException {
    String nombreArchivo = evento.getNombreArchivo();
    StringBuffer listaFolios = evento.getListaFolios();

    EvnRespImprimirFolio evRespuesta = new EvnRespImprimirFolio(null,
EvnRespImprimirFolio.AGREGAR_DE_ARCHIVO);
    evRespuesta.setNombreArchivo(nombreArchivo);
    evRespuesta.setListaFolios(listaFolios);
    return evRespuesta;
}
```

**Código Modificado Línea 2448:**

```
exito = tarea.programarTareas(evento.getNombreTarea(), evento.getFechaTarea(), evento.getListaFolios(),
evento.getTipoTarea());
```

ARCHIVO JAVA:

sir.core.eventos.administracion.EvnRespImprimirFolio.java

**Código Insertado Línea 37 - 39:**

```
/** Constante que identifica la accion agregar_de_archivo_ok */
public static final String AGREGAR_DE_ARCHIVO = "AGREGAR_DE_ARCHIVO";
```

**Código Insertado Línea 43 - 46:**

```
private String nombreArchivo;

private StringBuffer listaFolios;
```

**Código Insertado Línea 86 - 95:**

```

    /**
     * @param payload
     * @param tipoEvento
     */
    public EvnRespImprimirFolio(Object payload, StringBuffer listaFolios, String nombreArchvio, String tipoEvento) {
        super(payload, tipoEvento);
        this.nombreArchivo = nombreArchvio;
    }

```

```
        this.listaFolios = listaFolios;
    }
```

**Código Insertado Línea 173 - 175:**

```
public String getNombreArchivo() {
    return nombreArchivo;
```

**Código Insertado Línea 177 - 189:**

```
public void setNombreArchivo(String nombreArchivo) {
    this.nombreArchivo = nombreArchivo;
}

public StringBuffer getListaFolios() {
    return listaFolios;
}

public void setListaFolios(StringBuffer listaFolios) {
    this.listaFolios = listaFolios;
}
```

**ARCHIVO JAVA:**

sir.core.web.navegacion.administracion.ControlNavegacionImpresionFolio.java

**Código Insertado Línea 14- 15:**

```
import org.apache.commons.fileupload.FileUpload;
import org.auriga.smart.SMARTKeys;
```

**Código Insertado Línea 31 - 32:**

```
public static final String AGREGAR_DE_ARCHIVO_OK = "AGREGAR_DE_ARCHIVO_OK";
```

**Código Insertado Línea 62 - 64:**

```
boolean isMultipart = FileUpload.isMultipartContent(request);
```

```
if ( !isMultipart ) {
```

**Código Insertado Línea 94 - 100:**

```
}
```

```
} else {
```

```
String ultimaVista = (String)request.getSession().getAttribute(SMARTKeys.VISTA_ACTUAL);
```

```
if(ultimaVista.equals("admin.imprimirfolio.view")){
```

```
    return AGREGAR_DE_ARCHIVO_OK;
```

```
}
```

```
return null;
```

**ARCHIVO PROPERTIES: gov.sir.print.properties**

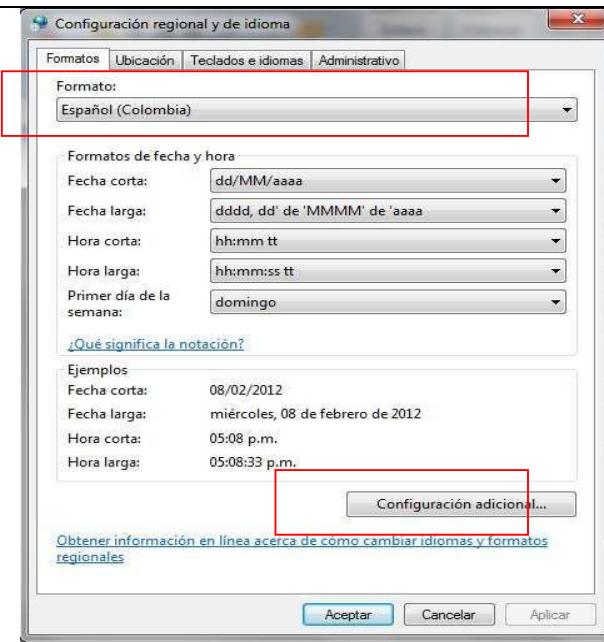
**Código Insertado Línea 50:**

imp\_masiva\_simple\_folio.ftp=ftp://192.168.202.26/

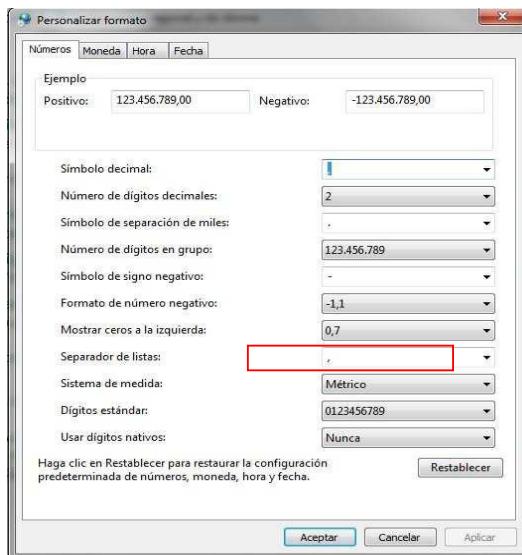
**Configuración del archivo de matrículas para generar sus PDFS**

- Para que el archivo que se cree con la extensión csv (delimitado por comas) y no genere conflicto se debe configurar el sistema por panel de control/configuración regional.

En la opción formato se debe colocar Español (Colombia), luego dar clic en configuración adicional como se muestra en la siguiente imagen.



Despues de dar clic en configuración adicional se despliega una nueva ventana donde se debe verificar la opción separador de listas la cual debe estar ( , ), luego dar aceptar. Ver siguiente imagen.



- Para crear el archivo incialmente se debe abrir el programa Microsoft oficce Excel, se debe ingresar el código del circulo regstral en una columna y seguido en otra columna el numero de la matricula, de forma secuencial. Debe generarse mas de tres columnas con información como se observa en la imagen.

	A	B	C	D	E	F	G	H	I	J	K
1	999	4650	999	4750	999	4850					
2	999	4651	999	4751	999	4851					
3											
4											
5											
6	Codigo		Matricula								
7	Registral										

- Despues de ingresar los datos debe guardarse el archivo con extension csv (delimitado por comas).

Guardado como

Nombre: corto\_archivo\_csv.csv

Tipo: CSV (delimitado por comas) (\*.csv)

Autores: FHERSHO

Etiquetas: Agregar una etiqueta

Ocultar carpetas

Guardar Cancelar

	A	B	C	D	E	F
1	999	4650	999	4750	999	4850
2	999	4651	999	4751	999	4851
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						

## 060 Adición de nota al certificado de Tradición y Libertad

### JUSTIFICACIÓN:

El texto que se incluirá en el Certificados de Tradición informará a 1 usuario que en él se REFLEJA LA SITUACIÓN JURIDICA DEL INMUEBLE HSTA LA FECHA Y HORA DE SU EXPEDICIÓN. Se solicita incluir dentro de los imprimibles de todos los certificados de tradición el siguiente texto “ESTE CERTIFICADOS REFLEJA LA SITUACIÓN JURÍDICA DEL INMUEBLE HASTA LA FECHA Y HORA DE SU EXPEDICIÓN”, este texto se debe incluir en letra mayúscula y negrita suficientemente legible, en la parte inicial del certificado debajo de la fecha y hora de impreso.

Validaciones a realizar: Que cualquier certificado de tradición expedido, contenga el texto requerido.

	<p style="text-align: center;">            Superintendencia de Notariado y Registro          Ministerio del Interior y de Justicia          República de Colombia          Libertad y Orden       </p> <p style="text-align: right;">Prosperidad para todos</p> <p style="text-align: center;">I.E.014334</p> <p style="text-align: center;">Bogotá, D.C., Noviembre 02 de 2011</p> <p style="text-align: center;">Doctora  <b>ELBA LUCIA CORREDOR</b>          Jefe Oficina de Informática          Superintendencia de Notariado y Registro          Presente.</p> <p style="text-align: center;">Ref. SOLICITUD DE ADICION NOTA AL CERTIFICADO DE TRADICION Y LIBERTAD</p> <p style="text-align: center;">Doctora Elba Lucia:</p> <p style="text-align: center;">Comedidamente le solicito se sirva adicionar al certificado de tradición y libertad que expedien las Oficinas de Registro de Instrumentos Públicos del país que operan los aplicativos SIR y FOLIO, la siguiente nota:</p> <p style="text-align: center;"><b>"ESTE CERTIFICADO REFLEJA LA SITUACION JURIDICA DEL INMUEBLE HASTA LA FECHA Y HORA DE SU EXPEDICION"</b></p> <p style="text-align: center;">Esta nota deberá figurar en letra mayúscula y negrita suficientemente legible, en la parte inicial del certificado, debajo de la fecha y hora de impresión.</p> <p style="text-align: center;">Cordialmente,</p> <p style="text-align: center;">   <b>ENRIQUE JOSE NATÉS GUERRA</b>          Superintendente Delegado para el Registro       </p> <p style="text-align: center;">C.C. Ing. José Luis Ariza (VUR)</p> <p style="text-align: center;">Proyecto: Pedro Malagón</p> <p style="text-align: center;">     <b>51 años</b>          Garantizando la guarda de la fi publica en Colombia       </p> <p style="text-align: center;">           Superintendencia de Notariado y Registro          Calle 26 No 13-49 M. 2 - Piso 10 - Tel. (1) 328-21-21          Bogotá D.C. Colombia  <a href="http://www.supnotariado.gov.co">www.supnotariado.gov.co</a> </p>	
--	---	--

### Desarrollo

- Script Bases de Datos:
  - Ninguno.
- Creacion De Imagenes:
  - Ninguno.
- Se crearon jsp nuevos en SIR:
  - Ninguno.
- Se modificaron los siguientes archivos en SIR:
  - sir.core.negocio.modelo.imprimibles.ImprimibleCertificado.java

### ARCHIVO JAVA:

**sir.core.negocio.modelo.imprimibles.ImprimibleCertificado.java**

### Código Insertado Línea 529 - 538:

```

/*
 * @author : Ellery David Robles Gómez.
 * @change : Impresión de nota certificado de tradicion.
 * Caso_Mantis : 010207 : Acta - Requerimiento No 060_151 - Adición de nota al Certificado de Tradición y
Libertad.
 */
String notaCertofocadoTradicion1 = StringFormat.getCentrada("\"ESTE CERTIFICADO REFLEJA LA SITUACION
JURIDICA DEL INMUEBLE", 61, 15);
String notaCertofocadoTradicion2 = StringFormat.getCentrada("HASTA LA FECHA Y HORA DE SU EXPEDICION\"", 61, 15);
this.imprimirLinea(ImprimibleConstantes.TITULO1, notaCertofocadoTradicion1);
this.imprimirLinea(ImprimibleConstantes.TITULO1, notaCertofocadoTradicion2);


```

### 064\_151\_Inconsistencias en redondeo

#### Error en el redondeo - Glpi 31761

Motivo Inconsistencia:

El usuario Jose Luis García informa que al calificar el turno 2012-300-6-473 al crear la anotación 03 del folio 351653 y colocar el valor de 4.028.674,68 el sistema lo redondea dejandolo como 4.028.700. Se requiere verificar porque al colocar un valor con centavos separado con coma el sistema lo redondea como en este caso que al colocar el valor de 4.028.674,68 el sistema lo deja en 4.028.700, se adjunta imagen de lo descrito anteriormente.

#### Reporte 25 v 51 - Glpi 31802

Motivo Inconsistencia: Se comunica la funcionaria Cenith Majarrez e informa que al sumar el valor total de derechos de registro de los usuarios JULIO.DELAHOZ y CARLOS.MANJARREZ del reporte 51 hay una diferencia de \$630 con el reporte 25 (Tiracaja) de cada uno de los usuarios del día 16-01-2012.

Al verificar en el sistema el reporte 25 (Tiracaja) de cada uno de los funcionarios y el reporte 51, en el valor de la columna de derechos de las Nuevas Entradas se encuentra una diferencia de \$630.además en el reporte 51 para el usuario JULIO.DELAHOZ en la columna de derechos para las nuevas entradas se observa un valor de \$10730 el cual no coincide con el valor de \$11400 de la columna del valor aplicado.

Favor verificar e informa que ocasionando esta inconsistencia reportada por la oficina y corregir el inconveniente.

### **Reporte 99: Ingresos diarios por documentos - Glpi 31867**

Motivo Inconsistencia:

Buenas tardes, reenvío el reporte de ingresos diarios por documento enviado por la ORIP de Honda, donde se refleja que para las nuevas entradas de documentos, el valor no se está redondeando en el imprimible, solicto su colaboración para subsanar este inconveniente.

Gracias

Cordialmente,

Dora Esperanza Pachon Rodriguez  
Grupo Desarrollo Informático

Se requiere verificar porque el valor total de este reporte no aparece redondeado tal como lo muestra la imagen adjunta.

#### **Análisis/Solución del Requerimiento:**

Se detecto que al momento de actualizar una anotación está teniendo en cuenta una función que se creó para aplicar redondeo; se requiere aplicar una condición para evitar el redondeo al momento de editar una anotación.

- Se modificaron los siguientes archivos en SIR:
  - sir.core.web.acciones.comun. AWTurno.java
  - sir.core.negocio.acciones.comun. ANTurno.java
  - generalSIR.negocio.validaciones.ValidacionesSIR.java

#### **ARCHIVO JS:**

#### **web.jsp.plantillas.common.js**

#### **Código Insertado Línea 221 - 285:**

```
/**  
 * Caso Mantis 10432 : Verificar redondeo del valor asignado.  
 * @Autor : Ellery David Robles Gómez.  
 */  
function formatoValor(my_str,id) {  
    var miles=1;  
    if(my_str.indexOf(".") == -1){  
        if(my_str.indexOf(",") == -1){  
            var nStr = "";  
            for(var i=1;i<=my_str.length;i++){  
                var desde = my_str.length-i*3;  
                var hasta = my_str.length-(3*(i-1));  
                var temp = my_str.slice(desde,hasta);  
                var separador="";  
                if(hasta>3){  
                    if(miles==1){  
                        miles=0;  
                        separador=",";  
                    } else {  
                        miles=miles-1;  
                        separador=",";  
                    }  
                }  
                nStr+=temp;  
            }  
        }  
    }  
    return nStr;  
}
```

```

        miles=1
        separador=",";
    }
    nStr=separador+temp+nStr;
}
else {
    if(hasta>0){
        temp=my_str.slice(0,hasta);
        nStr=temp+nStr;
    }
}
nStr=nStr+".00";
document.getElementById(id).value = nStr;
}
else {
    var largo = my_str.indexOf(".");
    var centavos = my_str.substr(largo,my_str.length);
    if(my_str.indexOf(",") == -1){
        var nStr = "";
        for(var i=1;i<=largo;i++){
            var desde = largo-i*3;
            var hasta = largo-(3*(i-1));
            var temp = my_str.slice(desde,hasta);
            var separador="";
            if(hasta>3){
                if(miles==1{
                    miles=0;
                    separador=",";
                } else {
                    miles=1
                    separador=",";
                }
                nStr=separador+temp+nStr;
            }
            else {
                if(hasta>0){
                    temp=my_str.slice(0,hasta);
                    nStr=temp+nStr;
                }
            }
        }
        nStr = nStr+centavos;
        document.getElementById(id).value = nStr;
    }
}
}

```

## **ARCHIVO JAVA:**

## sir.core.web.helpers.registro.AnotacionCancelacionHelper.java

## Código Insertado Linea 74 - 75:

```
private String tipoFormulario = "";
```

## Código Insertado Línea 520 - 522:

```
if(this.getTipoFormulario().equals("EDITAR_CANCELACION")) {  
    textHelper.setFuncion("onBlur=\"formatoValor(this.value,this.id)\"");  
} else {
```

**Código Insertado Linea 524:**

}

**Código Insertado Linea 799 - 801:**

```
public String getTipoFormulario() {  
    return tipoFormulario;  
}
```

**Código Insertado Linea 803 - 804:**

```
public void setTipoFormulario(String tipoFormulario) {  
    this.tipoFormulario = tipoFormulario;
```

## **Código Insertado Linea 806:**

}

## **ARCHIVO JSP:**

WEB-INF.jspSeguro.registro.EDITAR\_CANCELACION.jsp.

## Código Insertado Línea 61:

**ANALISIS**

Para dar solución a las inconsistencias reportadas para los reportes 25, 51 y 99 se actualizan los siguientes objetos en base de datos:

FNC\_VAL\_MASIVO.sql  
 FNC\_VALOR\_DOC\_PAGO.sql  
 SP\_Reporte\_25.sql  
 SP\_Reporte\_51.sql  
 SP\_Reporte\_99.sql

**DESPLIEGUE – 2.7.3****16/01/2012****006 Reespecificacion \_Tarifas Resolucion\_81\_2010-imm**

La Oficina de Informática de la SNR, teniendo en cuenta el concepto Jurídico de fecha 17 de Marzo de 2010, emitido por la Oficina Asesora Jurídica de la Superintendencia de Notariado y Registro y conforme a la Resolución 81 del 13 Enero de 2010 en el parágrafo del “Artículo 19 que de manera textual dice: **Para facilitar el recaudo y contabilización de los valores resultantes de la liquidación de los derechos de registro, estos se aproximarán a la centena más cercana**” (el subrayado y en negrilla viene en el oficio de referencia del concepto jurídico); solicita para dar cumplimiento a la resolución, se desarrolle el requerimiento para los actos que son objeto de registro causan un derecho y estos se deben cobrar y generar para el usuario un recibo de caja en el cual, entre otros ítems pre establecidos, debe reflejar el valor de cada acto a registrar:

- Se requiere que el valor de cada uno de estos actos se redondee a la centena más próxima. El redondeo que se requiere debe efectuarse de manera individual por acto y esto incluye todos los actos que deben pagar derechos de registro, entre los cuales se encuentran los certificados de tradición (cubre todos los tipos de certificados que tengan tarifa), proceso de fotocopias, todos los actos sin cuantía, actos con tarifas especiales, actos con cuantía, consultas, ingresos de mayores valores, devoluciones de dinero entre otros. Se requiere que el redondeo que se efectúe a la centena más próxima, se refleje en la impresión de recibos, en la impresión de listados y reportes. Es decir se debe reflejar el valor del acto ya ajustado.
- Se requiere que para el cobro de más de un certificado de tipo: normal, de ampliación de tradición, de pertenencia, mitad de tarifa, primero se redondee el valor individual pre- establecido para cada certificado y luego se multiplique por el número de copias solicitadas, en caso de que sean de la misma matrícula.
- Se requiere que para el cobro de más de un acto sin cuantía, con tarifa normal, mitad de tarifa o tarifa(s) especial(es), el cobro de hipotecas especiales como 70%, 40%, 10%-, que se encuentren dentro de un mismo turno de documento, (se pueden presentar una o varias combinaciones de actos y por lo tanto de tarifas) primero se debe redondear el valor individual pre-establecido para cada acto y luego de redondear cada acto se realice la sumatoria del valor total a pagar.
- Se requiere que los recibos de radicación de documentos contengan la liquidación de cada acto o actos que se redondeen de forma individual y el redondeo se haga a la centena más próxima. Ej. En un solo turno de radicación de documento existe una cancelación de patrimonio de familia, cuyo costo es de catorce mil doscientos ochenta pesos (\$14.280.00), en el mismo turno existe constitución de usufructo y la tarifa es de catorce mil doscientos ochenta pesos (\$14.280.00) y el mismo turno, contiene una constitución de afectación de vivienda cuyo costo es de catorce mil doscientos ochenta pesos (\$14.280.00); cada acto debe redondearse de manera individual y cada acto quedaría con un valor de catorce mil trescientos pesos (\$14.300.00) y luego del redondeo de cada acto se procede a realizar la sumatoria o total.
- Se requiere que el redondeo o ajuste a la centena más próxima no debe realizarse en el valor de

la sumatoria o total liquidado.

- Se requiere que los valores liquidados por acto y redondeados o ajustados a la centena más próxima se reflejen en la liquidación tanto en pantalla como en los impresos de manera individual o sea por cada acto y luego se realice la sumatoria o total liquidado.
- Se requiere para las nuevas entradas de documentos y/o certificados de tradición que incluyan actos que no hayan sido cobrados, nuevas entradas de documentos y/o certificados de tradición cuyas tarifas deban ser reajustadas y para sanciones (multas), que los valores liquidados por acto y redondeados o ajustados a la centena más próxima se reflejen en la liquidación tanto en pantalla como en los impresos de manera individual o sea por cada acto y luego se realice la sumatoria o total liquidado.
- Se requiere que en el Total a pagar se incluya el texto: Total ajustado a la centena por actos \$....
- Se requiere que al final de cada reporte existente en el resumen de totales debe modificarse el texto existente: Total Recaudo por Total Recaudo Ajustado a la Centena y en los reportes donde el texto es: Total Derechos se debe modificar por Total Derechos Ajustados a la centena.
- Se requiere que el redondeo de tarifas se refleje en los reportes pre-establecidos y en las columnas correspondientes.

Se requiere que el redondeo se establezca para el cobro de IMPUESTO DE REGISTRO, para las Oficinas de Registro de II.PP que realicen este recaudo y que el redondeo se refleje en los recibos y reportes (decreto reglamentario 650 de 1996, artículo 16).

#### **DESARROLLO:**

- Script Bases de Datos:
  - Ninguno
- Creacion De Imagenes:
  - Ninguno
- Se modificaron los siguientes archivos:
  - Ninguno

#### **ARCHIVO JSP:**

#### **WEB-INF.jspSeguro.antiguosistema.REVISION\_INICIAL\_MAYOR\_VALOR.jsp**

#### **Código Insertado Linea 50:**

```
/* JAcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Redondeo del Valor
 */
var round = valor / 100;
round = Math.round(round);
round = round * 100;
var my_str = new String(round);
```

#### **WEB-INF.jspSeguro.comun.REGISTRO\_PAGO.jsp**

#### **Código Insertado Linea 42:**

```
/* JAcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Se determina que tipos de liquidacion deben ser redondeadas.
 */
if(session.getAttribute(WebKeys.LIQUIDACION) instanceof LiquidacionTurnoCertificado ||
session.getAttribute(WebKeys.LIQUIDACION) instanceof LiquidacionTurnoCertificadoMasivo ||
session.getAttribute(WebKeys.LIQUIDACION) instanceof LiquidacionTurnoFotocopia ||
session.getAttribute(WebKeys.SOLICITUD) instanceof SolicitudConsulta){
    datosPagoHelper.setliquidacionRedondeo(true);
}
```

#### **WEB-INF.jspSeguro.correccion.CORRECCION\_MAYOR\_VALOR.jsp**

#### **Código Insertado Linea 53:**

```
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Redondeo del Valor
 */
var round = valor / 100;
round = Math.round(round);
round = round * 100;
var my_str = new String(round);
```

#### WEB-INF.jspSeguro.fotocopia.TURNO\_FOTOCOPIA\_DATOS.jsp

**Cambio Linea 404:**

```
<td width="179">Valor Liquidado</td>
```

**Por:**

```
<!--
```

```
JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
Cambio del texto "Valor Liquidado" por "Valor Total Liquidado Ajustado a la Centena por Actos"
```

```
-->
```

```
<td width="179">Valor Total Liquidado Ajustado a la Centena por Actos</td>
```

#### WEB-INF.jspSeguro.fotocopia.TURNO\_FOTOCOPIA\_DATOS\_PAGO.jsp

**Cambio Linea 404:**

```
<td width="179">Valor Total Liquidado</td>
```

**Por:**

```
<!--
```

```
JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
Cambio del texto "Valor Total Liquidado" por "Valor Total Liquidado Ajustado a la Centena por Actos"
```

```
-->
```

```
<td width="179">Valor Total Liquidado Ajustado a la Centena por Actos</td>
```

#### WEB-INF.jspSeguro.registro.CALIFICACION\_MAYOR\_VALOR.jsp

**Código Insertado Linea 50:**

```
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Redondeo del Valor
 */
var round = valor / 100;
round = Math.round(round);
round = round * 100;
var my_str = new String(round);
```

#### WEB-INF.jsp.plantillas.common.js

**Código Insertado Linea 154:**

```
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Redondeo del Valor
 */
var round = valor / 100;
round = Math.round(round);
round = round * 100;
var my_str = new String(round);
```

**CLASE:**

**gov.sir.core.negocio.acciones.administracion. ANTurnoManualRegistro.java**

**Método:**

**private EventoRespuesta liquidar(EvnTurnoManualRegistro evento) throws  
EventoException {**

**Cambio Linea 217:**

```
float valorOtroImpuesto = 0;
```

**Por:**

```
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Cambio de tipo de dato
 */
double valorOtroImpuesto = 0;
```

**CLASE:**

**gov.sir.core.web.acciones.registro.AWLiquidacionRegistro.java**

**Método:**

**private EvnLiquidacionRegistro liquidar(HttpServletRequest request)  
throws AccionWebException {**

**Cambio Linea 2482:**

```
liquidacion.setValorOtroImp(Float.parseFloat(valorOtro));
```

<b>Por:</b>
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc * Cambio de tipo de dato */ liquidacion.setValorOtroImp(Double.valueOf(valorOtro));
<b>Método:</b>
<b>private EvnLiquidacionRegistro editarLiquidacion(HttpServletRequest request) throws AccionWebException</b>
<b>Cambio Linea 3149:</b>
liquidacion.setValorOtroImp(Float.parseFloat(valorOtro));
<b>Por:</b>
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc * Cambio de tipo de dato */ liquidacion.setValorOtroImp(Double.valueOf(valorOtro));
<b>Método:</b>
<b>private void doEndValidarTurnoAnterior(HttpServletRequest request, HttpSession session, EvnRespLiquidacionRegistro evento) {</b>
<b>Cambio Linea 4581:</b>
float otroImpuesto = liqReq.getValorOtroImp();
<b>Por:</b>
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc * Cambio de tipo de dato */ double otroImpuesto = liqReq.getValorOtroImp();
<b>Método:</b>
<b>private void doEndBuscarSolicitudEdicion(HttpServletRequest session, EvnRespLiquidacionRegistro evento) {</b>
<b>Cambio Linea 5212:</b>
session.setAttribute(CActo.VALOR_OTRO_IMPUESTO,"" + liquidacion.getValorOtroImp());
<b>Por:</b>
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc * Cambio de formato */  session.setAttribute(CLiquidacionTurnoRegistro.VALOR_OTRO_IMPUESTO, StringFormat.getNumeroFormateado(liquidacion.getValorOtroImp().replaceAll(",",".")));
<b>CLASE:</b>
<b>gov.sir.core.web.helpers.comun.DatosPagoHelper.java</b>
<b>Código Insertado Linea 126:</b>
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc * Nueva Variable para manejar eventos de redondeo. */ private boolean liquidacionRedondeo = false;
<b>Código Insertado Linea 1646:</b>
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc * Nuevos metodos para el manejo del evento de redondeo. */ public boolean isliquidacionRedondeo() { return liquidacionRedondeo; }  public void setliquidacionRedondeo(boolean liquidacionRedondeo) { this.liquidacionRedondeo = liquidacionRedondeo; }
<b>Método:</b>
<b>public void drawGUI(HttpServletRequest request, JspWriter out) throws IOException, HelperException {</b>
<b>Cambio Linea 429:</b>
out.write("<td width=\"293\" class=\"campostile\">Valor total</td>");

**Por:**

```
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Cambio del Texto
 */
out.write("<td width=\"293\" class=\"campostitle\" >Valor total ajustado a la centena por actos</td>");
```

**Código Insertado Línea 446:**

```
else if (liquidacionRedondeo) {
out.write(
"<table width=\"100%\" border=\"0\" cellpadding=\"0\" cellspacing=\"2\" class=\"camposform\">";
out.write("<tr>");
out.write("<td width=\"20\"><img src=\"\" + request.getContextPath() +
"/jsp/images/ico_pago.gif\" width=\"16\" height=\"21\"></td>");
out.write("<td width=\"293\">Valor total ajustado a la centena por actos</td>");
out.write("<td width=\"11\">$</td>");
out.write("<td width=\"445\" class=\"campositem\">" +
formatador.format(valorLiquidacion) + "</td>");
out.write("</tr>");
out.write("</table>");}
```

**CLASE:**

**gov.sir.hermod.dao.impl.jdogenie.JDOLiquidacionesDAO.java**

**Método:**

**private double roundValue (double valor)**

**Cambio del ajuste de decena a centena del método.**

**CLASE:**

**gov.sir.core.negocio.modelo.LiquidacionTurnoRegistro.java**

**Cambio Línea 22:**

```
private float valorOtroImp;
```

**Por:**

```
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Cambio tipo de la variable valorOtroImp de float a double
 */
private double valorOtroImp;
```

**Cambio Línea 64:**

```
public float getValorOtroImp() {
```

**Por:**

```
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Cambio tipo de la variable valorOtroImp de float a double
 */
public double getValorOtroImp() {
```

**Cambio Línea 81:**

```
public void setValorOtroImp(float f) {
```

**Por:**

```
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Cambio tipo de la variable valorOtroImp de float a double
 */
public void setValorOtroImp(double f) {
```

**CLASE:**

**gov.sir.core.negocio.modelo.jdogenie.LiquidacionTurnoRegistroEnhanced.java**

**Cambio Línea 26:**

```
private float valorOtroImp;
```

**Por:**

```
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Cambio tipo de la variable valorOtroImp de float a double
 */
private double valorOtroImp;
```

<b>Cambio Linea 60:</b>
public float getValorOtroImp() {
<b>Por:</b>
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
* Cambio tipo de la variable valorOtroImp de float a double
*/
public double getValorOtroImp() {
<b>Cambio Linea 77:</b>
public void setValorOtroImp(float f) {
<b>Por:</b>
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
* Cambio tipo de la variable valorOtroImp de float a double
*/
public void setValorOtroImp(double f) {
<b>CLASE:</b>
<b>gov.sir.hermod.pagos.LiquidadorCertificados.java</b>
<b>Método:</b>
<b>public Liquidacion liquidar(Liquidacion liquidacion) throws LiquidarException {</b>
<b>Cambio Linea 93:</b>
valorTotal = tarifaApl.getValor();
<b>Por:</b>
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
* Redondeo de la variable tarifaApl.getValor()
*/
valorTotal = roundValue(tarifaApl.getValor());
<b>Cambio Linea 103:</b>
valorTotal = (valorCertificado * tarifaAplicada)* s.getNumeroCertificados();
<b>Por:</b>
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
* Redondeo del calculo (valorCertificado * tarifaAplicada)
*/
valorTotal = roundValue((valorCertificado * tarifaAplicada))* s.getNumeroCertificados();
<b>Método:</b>
<b>private double roundValue (double valor)</b>
<b>Cambio del ajuste de decena a centena del método.</b>
<b>CLASE:</b>
<b>gov.sir.hermod.pagos.LiquidadorFotocopia.java</b>
<b>Método:</b>
<b>public Liquidacion liquidar(Liquidacion liquidacion) throws LiquidarException {</b>
<b>Cambio Linea 115:</b>
valorDoc = valorTarifa*doc.getNumCopias()*doc.getNumHojas();
<b>Por:</b>
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
* Redondeo de valorTarifa antes del cálculo de copias y numero de hojas.
*/
valorDoc = roundValue(valorTarifa)*doc.getNumCopias()*doc.getNumHojas();
<b>Método:</b>
<b>private double roundValue (double valor)</b>
<b>Cambio del ajuste de decena a centena del método.</b>
<b>CLASE:</b>
<b>gov.sir.hermod.pagos.LiquidadorConsultas.java</b>
<b>Método:</b>
<b>public Liquidacion liquidar(Liquidacion liquidacion) throws LiquidarException {</b>
<b>Código Insertado Linea 87:</b>
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
* Redondeo del valor de la consulta.

<pre>/* valor = roundValue(valor);</pre>
<p style="text-align: center;"><b>Método:</b></p> <p style="text-align: center;"><b>private double roundValue (double valor)</b></p>
<p><b>Cambio del ajuste de decena a centena del método.</b></p>
<p style="text-align: center;"><b>CLASE:</b></p> <p style="text-align: center;"><b>gov.sir.hermiod.pagos.LiquidadorRegistro.java</b></p>
<p style="text-align: center;"><b>Método:</b></p> <p style="text-align: center;"><b>public Liquidacion liquidar(Liquidacion liquidacion) throws LiquidarException {</b></p>
<p><b>Cambio Línea 102:</b></p> <pre>double valOtrosImpuestosLiq = liqReg.getValorOtroImp();</pre>
<p><b>Por:</b></p> <pre>/* JAICaza caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc  * Se elimina esta linea que no es necesaria para el redondeo de liqReg.getValorOtroImp().</pre>
<p><b>Cambio Línea 106:</b></p> <pre>Float floatImpuesto = new Float(valOtrosImpuestosLiq); liqReg.setValorOtroImp(floatImpuesto.floatValue());</pre>
<p><b>Por:</b></p> <pre>/* JAICaza caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc  * Se aplica redondeo a liqReg.getValorOtroImp() que posee el valor de otros impuestos.</pre>
<pre>*/ liqReg.setValorOtroImp(roundValue(liqReg.getValorOtroImp()));</pre>
<p style="text-align: center;"><b>Método:</b></p> <p style="text-align: center;"><b>public Liquidacion reliquidar(Liquidacion liquidacion) throws LiquidarException {</b></p>
<p><b>Código Insertado Línea 649:</b></p>
<pre>/* JAICazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc  * Redondeo del valor de la variable liqReg.getValorOtroImp().</pre>
<pre>/* liqReg.setValorOtroImp(roundValue(liqReg.getValorOtroImp()));</pre>
<p style="text-align: center;"><b>Método:</b></p> <p style="text-align: center;"><b>private void calcularValores(SolicitudRegistro s, LiquidacionTurnoRegistro liqReg, List actos) throws LiquidarException {</b></p>
<p><b>Cambio Línea 787:</b></p> <pre>liqReg.setValorDerechos(liqReg.getValorDerechos() - actoAnterior.getValor()); liqReg.setValorImpuestos(liqReg.getValorImpuestos() - actoAnterior.getValorImpuestos());</pre>
<p><b>Por:</b></p> <pre>/* JAICazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc  * Redondeo del valor de las variables actoAnterior.getValor() y actoAnterior.getValorImpuestos().</pre>
<pre>/* liqReg.setValorDerechos(liqReg.getValorDerechos() - roundValue(actoAnterior.getValor())); liqReg.setValorImpuestos(liqReg.getValorImpuestos() - roundValue(actoAnterior.getValorImpuestos()));</pre>
<p><b>Cambio Línea 814:</b></p> <pre>liqReg.setValorOtroImp(liqReg.getValorOtroImp() - (float)valorOtroImpAnt);</pre>
<p><b>Por:</b></p> <pre>/* JAICazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc  * Redondeo del valor de la variable valorOtroImpAnt.</pre>
<pre>/* liqReg.setValorOtroImp(liqReg.getValorOtroImp() - roundValue(valorOtroImpAnt));</pre>
<p style="text-align: center;"><b>Método:</b></p> <p style="text-align: center;"><b>private double roundValue (double valor)</b></p>
<p><b>Cambio del ajuste de decena a centena del método.</b></p>
<p style="text-align: center;"><b>CLASE:</b></p> <p style="text-align: center;"><b>gov.sir.core.negocio.modelo.imprimibles. ImprimibleConstanciaLiquidacion.java</b></p>
<p style="text-align: center;"><b>Método:</b></p> <p style="text-align: center;"><b>private void imprimirCuerpo() {</b></p>
<p><b>Cambio Línea 186:</b></p>

```

this.imprimirLinea(ImprimibleConstantes.TITULO_GRANDE,
    " VALOR TOTAL " + this.getNumeroFormatead0(lqTurno.getValor()), true);
Por:
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Cambio del texto "VALOR TOTAL " por "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS ".
 */
this.imprimirLinea(ImprimibleConstantes.TITULO_GRANDE,
    "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS " + this.getNumeroFormatead0(lqTurno.getValor()),
true);
CLASE:
gov.sir.core.negocio.modelo.imprimibles. ImprimibleFotocopiaLiquidarSolicitud.java
Método:
protected void imprimirCuerpo_Items(PageFormat pageFormat) {
Cambio Linea 66:
this.imprimirLinea(ImprimibleConstantes.TITULO2, "Valor Liquidado : " + decimalFormat.format(valorLiquidado),
false);
Por:
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Cambio del texto "Valor Liquidado : " por "Valor Liquidado Ajustado a la Centena por Actos: ".
 */
this.imprimirLinea(ImprimibleConstantes.TITULO2, "Valor Liquidado Ajustado a la Centena por Actos: " +
decimalFormat.format(valorLiquidado), false);

Cambio Linea 66:
this.imprimirLinea(ImprimibleConstantes.TITULO2, "Valor liquidado (negativo) :" +
decimalFormat.format(valorLiquidado), false);
Por:
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Cambio del texto "Valor liquidado (negativo) :" por "Valor liquidado (negativo) Ajustado a la Centena por Actos: ".
 */
this.imprimirLinea(ImprimibleConstantes.TITULO2, "Valor liquidado (negativo) Ajustado a la Centena por Actos: " +
decimalFormat.format(valorLiquidado), false);

CLASE:
gov.sir.core.negocio.modelo.imprimibles. ImprimibleFotocopiaPago_SolicitudData.java
Método:
protected void imprimirFormaPago() {
Cambio Linea 251:
this.imprimirLinea(ImprimibleConstantes.TITULO2, "VALOR TOTAL: $ ",false);
this.imprimirLinea(ImprimibleConstantes.TITULO_GRANDE2, 105, valorTotal);
Por:
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Cambio del texto "VALOR TOTAL " por "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS " y cambio en la
posición de impresión.
*/
this.imprimirLinea(ImprimibleConstantes.TITULO2, "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: $ ",false);
this.imprimirLinea(ImprimibleConstantes.TITULO_GRANDE2, 260, valorTotal);
CLASE:
gov.sir.core.negocio.modelo.imprimibles.ImprimibleRecibo.java
Método:
protected void imprimirFormaPago() {
Código Insertado Linea 2918:
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
 * Se coloca el texto "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS " para los tipos de solicitud
SolicitudRegistro, SolicitudConsulta, SolicitudCertificado y SolicitudCertificadoMasivo.
*/
if(solicitud instanceof SolicitudRegistro || solicitud instanceof SolicitudConsulta || solicitud instanceof
SolicitudCertificado|| solicitud instanceof SolicitudCertificadoMasivo){
    this.imprimirLinea(ImprimibleConstantes.TITULO2, "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: $ "
",false);
    this.imprimirLinea(ImprimibleConstantes.TITULO_GRANDE2, 260, valorTotal);
}

```

<pre> }else De acuerdo a las observaciones de SNR el 18 de Octubre del 2011, se realizan los siguientes cambios: <b>CLASE:</b> <b>gov.sir.core.negocio.modelo.imprimibles.ImprimibleRecibo.java</b> </pre>
<p>Cambio Linea 2918:</p> <pre> /* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010- imm.doc * Se coloca el texto "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS " para los tipos de solicitud SolicitudRegistro, SolicitudConsulta, SolicitudCertificado y SolicitudCertificadoMasivo. */ if(solicitud instanceof SolicitudRegistro    solicitud instanceof SolicitudConsulta    solicitud instanceof SolicitudCertificado   solicitud instanceof SolicitudCertificadoMasivo){     this.imprimirLinea(ImprimibleConstantes.TITULO2, "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: \$ ",false);     this.imprimirLinea(ImprimibleConstantes.TITULO_GRANDE2, 260, valorTotal); } </pre> <p>Por:</p> <pre> /* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010- imm.doc * Se coloca el texto "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS " para los tipos de solicitud SolicitudConsulta,SolicitudCertificado y SolicitudCertificadoMasivo, y se coloca el texto" VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS:para el tipo de solicitud SolicitudRegistro. */ if(solicitud instanceof SolicitudConsulta    solicitud instanceof SolicitudCertificado   solicitud instanceof SolicitudCertificadoMasivo){     this.imprimirLinea(ImprimibleConstantes.TITULO2, "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS: \$ ", false);     this.imprimirLinea(ImprimibleConstantes.TITULO_GRANDE2, 260, valorTotal); }else if(solicitud instanceof SolicitudRegistro ){     this.imprimirLinea(ImprimibleConstantes.TITULO2, "VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS: \$ ",false);     this.imprimirLinea(ImprimibleConstantes.TITULO_GRANDE2, 300, valorTotal); } </pre>
<b>gov.sir.core.negocio.modelo.imprimibles. ImprimibleConstanciaLiquidacion.java</b>
<p>Cambio Linea 186:</p> <pre> /* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010- imm.doc * Cambio del texto "VALOR TOTAL " por "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS ". */ this.imprimirLinea(ImprimibleConstantes.TITULO_GRANDE,"VALOR TOTALAJUSTADO A LA CENTENA POR ACTOS " + this.getNumeroFormatoado(liqTurno.getValor()), true); </pre> <p>Por:</p> <pre> /* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010- imm.doc * Cambio del texto "VALOR TOTAL " por "VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS ". */ this.imprimirLinea(ImprimibleConstantes.TITULO1,"VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS " + this.getNumeroFormatoado(liqTurno.getValor()), true); </pre> <p>De acuerdo a las observaciones de SNR el 19 de Octubre del 2011, se realizan los siguientes cambios:</p>
<b>gov.sir.hermod.pagos.LiquidadorConsultas.java</b>
<p>Cambio Linea 89:</p> <pre> valor = roundValue(valor); </pre> <p>Por</p> <pre> /* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Re especificación _Tarifas Resolucion_81_2010- imm.doc </pre>

\* Se elimina esta línea para aplicar el redondeo despues del cálculo por número de búsquedas.  
\*/

### **gov.sir.hermode.pagos.LiquidadorFotocopia.java**

/\* JAlcaza caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion \_Tarifas Resolucion\_81\_2010-imm.doc  
\* Redondeo de valorTarifa despues del cálculo de copias y numero de hojas.  
\*/

valorLiq = roundValue(valorLiq);

De acuerdo a las observaciones de SNR el 20 de Octubre del 2011, se realizan los siguientes cambios:

### **gov.sir.core.negocio.modelo.imprimibles.ImprimibleRecibo.java**

Cambio Linea 2918:

/\* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion \_Tarifas Resolucion\_81\_2010-imm.doc

\* Se coloca el texto "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS " para los tipos de solicitud SolicitudConsulta,SolicitudCertificado y SolicitudCertificadoMasivo, y se coloca el texto" VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS:para el tipo de solicitud SolicitudRegistro.

\*/

if(solicitud instanceof SolicitudConsulta || solicitud instanceof SolicitudCertificado || solicitud instanceof SolicitudCertificadoMasivo){

Por

/\* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion \_Tarifas Resolucion\_81\_2010-imm.doc

\* Se coloca el texto "VALOR TOTAL AJUSTADO A LA CENTENA POR ACTOS " para los tipos de solicitud SolicitudConsulta,SolicitudCertificado, SolicitudCertificadoMasivo y Pagos Mayor Valor Correciones, y se coloca el texto" VALOR TOTAL A PAGAR AJUSTADO A LA CENTENA POR ACTOS:para el tipo de solicitud SolicitudRegistro.

\*/

if(solicitud instanceof SolicitudConsulta || solicitud instanceof SolicitudCertificado|| solicitud instanceof SolicitudCertificadoMasivo || this.esMayorValorCorrecciones){

De acuerdo a la revisión y solicitudes realizadas por parte de la SNR el dia 16 de Noviembre se modifican los siguientes archivos.

### **gov.sir.hermode.pagos.LiquidadorConsultas.java**

Cambio Linea 85:

// if (kind.equals(TipoConsulta.TIPO\_COMPLEJO))

// {

/\* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion \_Tarifas Resolucion\_81\_2010-imm.doc

\* Redondeo del valor de la consulta.

\*/

valor = roundValue(valor);

valor \*= s.getNumeroMaximoBusquedas();

// }

valor = roundValue(valor);

Por

/\* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion \_Tarifas Resolucion\_81\_2010-imm.doc

\* Se elimina codigo comentariado y se aplica el redondeo antes de el calculo de numeros de busquedas.

\*/

valor = roundValue(valor);

valor \*= s.getNumeroMaximoBusquedas();

### **gov.sir.hermode.dao.impl.jdogenie.JDOLiquidacionesDAO.java**

Cambio Linea 1899:

valorActo = cargoFijo \* porcentajeTarifa \* acto.getCuantia();

Por

/\* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion \_Tarifas Resolucion\_81\_2010-imm.doc

\* Para los tipos de acto con cargo fijo primero se redondea la tarifa y luego se calcula con el numero de actos.

\*/

```
valorActo = roundValue(cargoFijo * porcentajeTarifa) ;  
valorActo = valorActo * acto.getCuantia();
```

De acuerdo a las observaciones hechas el 18 de Noviembre se hace la siguiente modificación.

**gov.sir.hermod.pagos.LiquidadorRegistro.java**

Se modifican Archivos.

Cambio Linea 465:

```
liqCertAux.setValor(tarifa.getValor());
```

Por

```
/* JAlcazar caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
```

\* Redondeo del valor de tarifa.getValor()

\*/

```
liqCertAux.setValor(roundValue(tarifa.getValor()));
```

De acuerdo a las observaciones realizadas el 22 de noviembre del 2011, se modifican los archivos.

**gov.sir.hermod.pagos.LiquidadorFotocopia.java**

Cambio linea 115

```
valorDoc = valorTarifa*doc.getNumCopias()*doc.getNumHojas();
```

```
valorLiq = valorLiq+valorDoc;
```

```
/* JAlcaza caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
```

\* Redondeo de valorTarifa despues del cálculo de copias y numero de hojas.

\*/

```
valorLiq = roundValue(valorLiq);
```

Por

```
/* JAlcaza caso Mantis 05648: Acta - Requerimiento No 006 -Reespecificacion _Tarifas Resolucion_81_2010-imm.doc
```

\* Redondeo de valorTarifa antes del calculo de copias y numero de hojas.

\*/

```
valorDoc = roundValue(valorTarifa)*doc.getNumCopias()*doc.getNumHojas();
```

```
valorLiq = valorLiq+valorDoc;
```

**SALVEDAD:** La modificación del diseño de los reportes aplica para la generación sin tener en cuenta parámetros.

**BASE DE DATOS**

- Procedimiento SP\_REPORTES\_145. Se adjunta procedimiento.

**ARCHIVOS**

**Reporte 05 – Diario mayores valores y devoluciones del día**

ANTES

Untitled Document - Windows Internet Explorer  
 http://192.168.202.22:778/SNP/reportes.do

Superintendencia de Notariado y Registro  
 Ministerio del Interior y de Justicia  
 República de Colombia  
 www.supernotariado.gov.co

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS      BUCARAMANGA - 300  
 DIARIO MAYORES VALORES DEL DIA      05-10-2011

Impreso el 06 de Octubre de 2011 a las 12:06:49 PM

Página 1

**MAYORES VALORES**

Proceso:	REGISTRO						
Radicacion	Valor	Efectivo	Cheque	Consignacion	Derechos	Impuestos	Hora Pago
2011-300-6-42552	\$ 15,000	\$ 15,000	\$ 0	\$ 0	\$ 15,000	\$ 0	04:00:04 PM
<b>Total Derechos:</b>	<b>\$ 15,000</b>						
<b>Total Impuestos:</b>	<b>\$ 0</b>						
<b>TOTAL MAYORES VALORES REGISTRO</b>	<b>\$ 15,000</b>						
<b>TOTAL MAYORES VALORES</b>	<b>\$ 15,000</b>						

AHORA

Untitled Document - Windows Internet Explorer  
 http://192.168.190.106:8080/SNP/jsp/plantillas/report-displayer/reportes.do

Superintendencia de Notariado y Registro  
 Ministerio del Interior y de Justicia  
 República de Colombia  
 www.supernotariado.gov.co

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS      BUCARAMANGA - 300  
 DIARIO MAYORES VALORES DEL DIA      05-10-2011

Impreso el 06 de Octubre de 2011 a las 16:26:46 PM

Página 1

**MAYORES VALORES**

Proceso:	REGISTRO						
Radicacion	Valor	Efectivo	Cheque	Consignacion	Derechos	Impuestos	Hora Pago
2011-300-6-42552	\$ 15,000	\$ 15,000	\$ 0	\$ 0	\$ 15,000	\$ 0	04:00:04 PM
<b>Total Derechos Ajustados a la Centena:</b>	<b>\$ 15,000</b>						
<b>Total Impuestos:</b>	<b>\$ 0</b>						
<b>TOTAL MAYORES VALORES REGISTRO</b>	<b>\$ 15,000</b>						
<b>TOTAL MAYORES VALORES</b>	<b>\$ 15,000</b>						

Reporte 06 - Diario radicador de certificados del día  
 ANTES

CERTIFICADOS DE OTRAS ORIPS									
Círculo	Cantidad	Valor							
SANTA MARTA (080)	1	\$ 12,460							
SIMI TI (068)		\$ 12,460							
VALLEDUPAR (190)	1	\$ 12,460							
MONTERIA (140)	1	\$ 12,460							
PUERTO LOPEZ (234)	2	\$ 24,920							
CARTAGENA (060)	4	\$ 49,840							
SAN VICENTE DE CHUCURI (020)	2	\$ 24,920							
PIEDECUESTA (314)	41	\$ 510,860							
TUNJA (070)	2	\$ 24,920							
<b>Total certificados entre ORIPS'S</b>									
		<b>\$ 685,390</b>							

#### AHORA

CERTIFICADOS DE OTRAS ORIPS									
Círculo	Cantidad	Valor							
SANTA MARTA (080)	1	\$ 12,460							
SIMI TI (068)		\$ 12,460							
VALLEDUPAR (190)	1	\$ 12,460							
MONTERIA (140)	1	\$ 12,460							
PUERTO LOPEZ (234)	2	\$ 24,920							
CARTAGENA (060)	4	\$ 49,840							
SAN VICENTE DE CHUCURI (020)	2	\$ 24,920							
PIEDECUESTA (314)	41	\$ 510,860							
TUNJA (070)	2	\$ 24,920							
<b>Total certificados entre ORIPS'S</b>									
		<b>\$ 685,390</b>							

#### REPORTE 6B - Diario radicador de certificados masivos del día

ANTES

2011-300-1-219592	300-263978
2011-300-1-219593	300-333650
2011-300-1-219594	300-92786
2011-300-1-219595	300-46606
2011-300-1-219596	300-207024
2011-300-1-219597	300-10966
2011-300-1-219598	300-102653
2011-300-1-219599	300-98526
2011-300-1-219600	300-99092
2011-300-1-219601	300-227816
2011-300-1-219602	300-100419
2011-300-1-219603	300-314576
2011-300-1-219604	300-102704
2011-300-1-219585	300-228723
2011-300-1-219584	300-250840
2011-300-1-219583	300-213636
2011-300-1-219582	300-207025
2011-300-1-219581	300-279070
2011-300-1-219590	300-9557

**Total derechos:** \$ 0  
**Total masivos:** 4  
**Total masivos anulados:** 0  
**Total individuales exentos:** 30  
**Total individuales con pago:** 0

### AHORA

2011-300-1-219593	300-333650
2011-300-1-219594	300-92786
2011-300-1-219595	300-46606
2011-300-1-219596	300-207024
2011-300-1-219597	300-10966
2011-300-1-219598	300-102653
2011-300-1-219599	300-98526
2011-300-1-219600	300-99092
2011-300-1-219601	300-227816
2011-300-1-219602	300-100419
2011-300-1-219603	300-314576
2011-300-1-219604	300-102704
2011-300-1-219585	300-228723
2011-300-1-219584	300-250840
2011-300-1-219583	300-213636
2011-300-1-219582	300-207025
2011-300-1-219581	300-279070
2011-300-1-219590	300-9557

**Total Derechos Ajustados a la Centena:** \$ 0  
**Total masivos:** 4  
**Total masivos anulados:** 0  
**Total individuales exentos:** 30  
**Total individuales con pago:** 0

**REPORTE 09 - Diario radicador de documentos del dia**

**ANTES**

Hora	Turno Ant	Turno	Acto	Nro Doc.	Fecha	Oficina	Ciudad	Total Derechos	Recibo	Cajero
15:38:37		45507	VENTA,AFECTACION A VIVIENDA FAMILIAR,REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000) VENTA	(3) 3081	28-09-2011	NOTARIA PRIMERA	BUCARAMANGA	\$ 888,280	59827559	52689
15:43:32		45508	HIPOTECA	(1) 1373	24-08-2011	NOTARIA PRIMERA	FLORIDABLANCA	\$ 815,000	59841256	52694
15:45:23		45509	REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000),HIPOTECA ESPECIAL,HIPOTECA,CANCELACION HIPOTECA,VENTA,VARIOS	(1) 5290	04-10-2011	NOTARIA SEPTIMA	BUCARAMANGA	\$ 500,000	59841257	52694
15:51:42		45510	REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000),HIPOTECA ESPECIAL,HIPOTECA,CANCELACION HIPOTECA,VENTA,VARIOS	(6) 2985	21-09-2011	NOTARIA PRIMERA	BUCARAMANGA	\$ 1,088,120	59827562	52689
15:51:49		45511	Liquidacion Sociedad CONYUGAL (CON CUANTIA PARA IMPUTO)	(1) .	04-10-2011	JUZGADO TERCERO DE FAMILIA	CUCUTA	\$ 200,000	59841258	52694
15:53:37		45512	CANCELACION VALORIZACION	(1) 3776	05-10-2011	AREA METROPOLITANA	BUCARAMANGA	\$ 0	59841259	52694
15:54:19		45513	CANCELACION VALORIZACION	(1) AMB- SAF-V- 476- 2011	05-10-2011	AREA METROPOLITANA	BUCARAMANGA	\$ 0	59844417	52777
15:54:45		45514	REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000) VARIOS	(1) 2985	21-09-2011	NOTARIA PRIMERA	BUCARAMANGA	\$ 9,060	59827563	52689
15:56:25		45515	REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000) VARIOS	(1) 1229	28-09-2011	JUZGADO SEGUNDO PENAL MUNICIPAL CON FUNCIONES DE CONOCIMIENTO	BUCARAMANGA	\$ 0	59844418	52777
15:57:54		45516	HIPOTECA	(1) 4758	03-10-2011	NOTARIA QUINTA	BUCARAMANGA	\$ 682,500	59841260	52694
15:59:44		45517	CANCELACION HIPOTECA,HIPOTECA,HIPOTECA	(2) 4776	04-10-2011	NOTARIA QUINTA	BUCARAMANGA	\$ 150,000	59841261	52694
16:02:55		45518	HIPOTECA,PATRIMONIO DE FAMILIA	(2) 4780	04-10-2011	NOTARIA QUINTA	BUCARAMANGA	\$ 139,730	59841262	52694
<b>Total Derechos:</b>								<b>\$ 55,348,970</b>		
<b>Total Turnos:</b>								<b>260</b>		

**AHORA**

Hora	Turno Ant	Turno	Acto	Nro Doc.	Fecha	Oficina	Ciudad	Total Derechos Ajustados a la Centena	Recibo	Cajero
15:18:44		45500	CANCELACION HIPOTECA	(1) 1444	30-09-2011	NOTARIA CUARTA	BUCARAMANGA	\$ 60,000	59841250	52694
15:19:50	2011-300-6- 41266	45501	CANCELACION AFECTACION FAMILIAR,Liquidacion SOCIEDAD CONYUGAL (CON CUANTIA PARA IMPUTO),ACLARACION CANCELACION HIPOTECA	(3) 4729	02-09-2011	NOTARIA SEPTIMA	BUCARAMANGA	\$ 14,730	59827556	52689
15:20:48		45502	VENTA	(1) 5287	04-10-2011	NOTARIA SEPTIMA	BUCARAMANGA	\$ 25,000	59827557	52689
15:21:54		45503	EMBARGO	(1) 5302	04-10-2011	NOTARIA SEPTIMA	BUCARAMANGA	\$ 436,250	59841251	52694
15:27:19		45504	EMBARGO	(1) 3025	03-10-2011	JUZGADO SEPTIMO CIVIL MUNICIPAL	BUCARAMANGA	\$ 14,730	59841252	52694
15:29:30		45505	EMBARGO	(1) 3026	03-10-2011	JUZGADO SEPTIMO CIVIL MUNICIPAL	BUCARAMANGA	\$ 14,730	59841254	52694
15:36:07	2011-300-6- 43569	45506	VENTA	(1) 2529	08-08-2011	NOTARIA PRIMERA	BUCARAMANGA	\$ 0	59827558	52689
15:38:37		45507	VENTA,AFECTACION A VIVIENDA FAMILIAR,REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000) VENTA	(3) 3081	28-09-2011	NOTARIA PRIMERA	BUCARAMANGA	\$ 888,280	59827559	52689
15:43:32		45508	HIPOTECA	(1) 1373	24-08-2011	NOTARIA PRIMERA	FLORIDABLANCA	\$ 815,000	59841256	52694
15:45:23		45509	REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000),HIPOTECA ESPECIAL,HIPOTECA,CANCELACION HIPOTECA,VENTA,VARIOS	(1) 5290	04-10-2011	NOTARIA SEPTIMA	BUCARAMANGA	\$ 500,000	59841257	52694
15:51:42		45510	REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000),HIPOTECA ESPECIAL,HIPOTECA,CANCELACION HIPOTECA,VENTA,VARIOS	(6) 2985	21-09-2011	NOTARIA PRIMERA	BUCARAMANGA	\$ 1,088,120	59827562	52689
15:51:49		45511	Liquidacion Sociedad CONYUGAL (CON CUANTIA PARA IMPUTO)	(1) .	04-10-2011	JUZGADO TERCERO DE FAMILIA	CUCUTA	\$ 200,000	59841258	52694
15:53:37		45512	CANCELACION VALORIZACION	(1) 3776	05-10-2011	AREA METROPOLITANA	BUCARAMANGA	\$ 0	59841259	52694
15:54:19		45513	CANCELACION VALORIZACION	(1) AMB- SAF-V- 476- 2011	05-10-2011	AREA METROPOLITANA	BUCARAMANGA	\$ 0	59844417	52777
15:54:45		45514	REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000) VARIOS	(1) 2985	21-09-2011	NOTARIA PRIMERA	BUCARAMANGA	\$ 9,060	59827563	52689
15:56:25		45515	REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000) VARIOS	(1) 1229	28-09-2011	JUZGADO SEGUNDO PENAL MUNICIPAL CON FUNCIONES DE CONOCIMIENTO	BUCARAMANGA	\$ 0	59844418	52777
15:57:54		45516	HIPOTECA	(1) 4758	03-10-2011	NOTARIA QUINTA	BUCARAMANGA	\$ 682,500	59841260	52694
15:59:44		45517	CANCELACION HIPOTECA,HIPOTECA,HIPOTECA	(2) 4776	04-10-2011	NOTARIA QUINTA	BUCARAMANGA	\$ 150,000	59841261	52694
16:02:55		45518	HIPOTECA,PATRIMONIO DE FAMILIA	(2) 4780	04-10-2011	NOTARIA QUINTA	BUCARAMANGA	\$ 139,730	59841262	52694
<b>Total Derechos Ajustados a la Centena:</b>								<b>\$ 55,348,970</b>		
<b>Total Turnos:</b>								<b>260</b>		
<b>Total Anulados:</b>								<b>1</b>		

**Reporte 99 - Ingresos diarios por documentos**

**ANTES**

Untitled Document - Windows Internet Explorer  
 http://192.168.202.22:7779/SNR/reportes.do

2011-300-6-45512	59841259 AREA METROPOLITANA DE	CANCELACION VALORIZACION REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000)	N	1	\$ 0
2011-300-6-45514	59827563 URBANAS S A 6387466		N	1	\$ 9,060
2011-300-6-45515	59844418 SEGUNDO PENAL DE CONOCIMIENTO MUNICIPAL	VARIOS	E	1	\$ 0
2011-300-6-45516	59841260 LUZ MARINA VARGAS 6991653	HIPOTECA	N	136,500,000	\$ 682,500
2011-300-6-45517	59841261 LUZ MARINA VARGAS 6991653	CANCELACION	N	3,500,000	\$ 17,500
2011-300-6-45517	59841261 LUZ MARINA VARGAS 6991653	HIPOTECA	N	26,500,000	\$ 132,500
2011-300-6-45518	59841262 LUZ MARINA VARGAS 6991653	HIPOTECA	N	25,000,000	\$ 125,000
2011-300-6-45518	59841262 LUZ MARINA VARGAS 6991653	PATRIMONIO DE FAMILIA	N	1	\$ 14,730
Total recaudo 05 de Octubre de 2011:					<b>\$ 55,348,970</b>

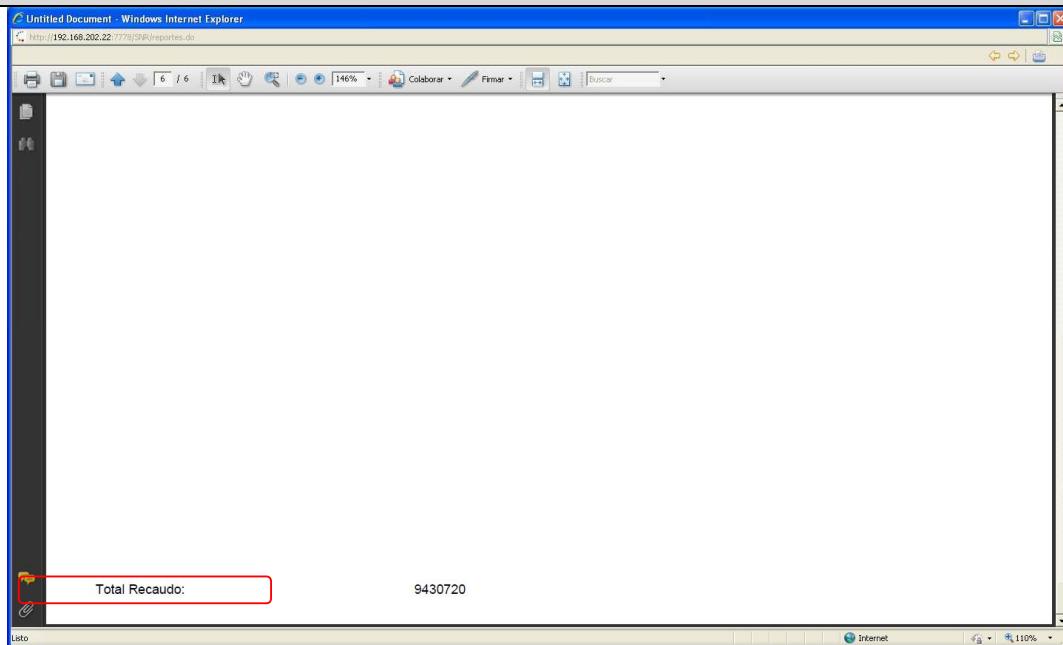
**AHORA**

Untitled Document - Windows Internet Explorer  
 http://192.168.150.106:8080/SNR/reportes.do

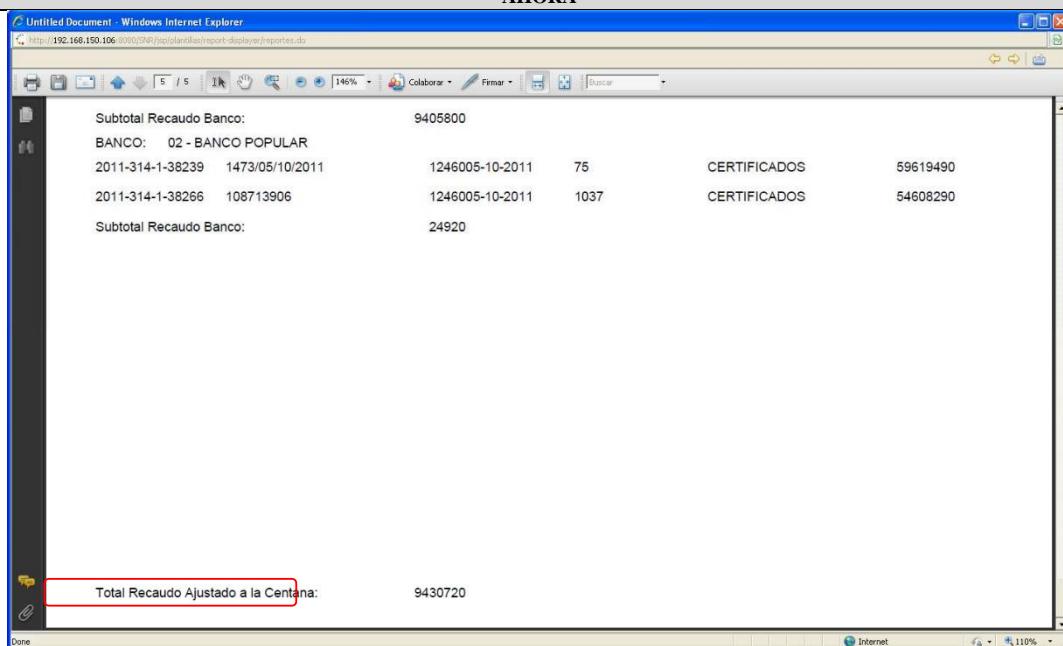
2011-300-6-45511	59841258 LUZ MARY RAMIREZ HERNANDEZ 3014196304	LIQUIDACION SOCIEDAD CONYUGAL (CON CUANTIA PARA IMPTO)	N	40,000,000	\$ 200,000
2011-300-6-45512	59841259 AREA METROPOLITANA DE BUCARAMANGA	CANCELACION	N	1	\$ 0
2011-300-6-45513	59844417 AREA METROPOLITANA	CANCELACION VALORIZACION	N	1	\$ 0
2011-300-6-45514	59827563 URBANAS S.A 6387466	REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000)	N	1	\$ 9,060
2011-300-6-45515	59844418 SEGUNDO PENAL DE CONOCIMIENTO MUNICIPAL	VARIOS	E	1	\$ 0
2011-300-6-45516	59841260 LUZ MARINA VARGAS 6991653	HIPOTECA	N	136,500,000	\$ 682,500
2011-300-6-45517	59841261 LUZ MARINA VARGAS 6991653	HIPOTECA	N	3,500,000	\$ 17,500
2011-300-6-45517	59841261 LUZ MARINA VARGAS 6991653	HIPOTECA	N	26,500,000	\$ 132,500
2011-300-6-45518	59841262 LUZ MARINA VARGAS 6991653	HIPOTECA	N	25,000,000	\$ 125,000
2011-300-6-45518	59841262 LUZ MARINA VARGAS 6991653	PATRIMONIO DE FAMILIA	N	1	\$ 14,730
Total recaudo ajustado a la centena 05 de Octubre de 2011:					<b>\$ 55,348,970</b>

### Reporte 145 - Informe consignaciones bancarias del día

ANTES



AHORA



### Reporte 151 - Diario radicador de testamentos

ANTES

Untitled Document - Windows Internet Explorer  
http://192.168.202.22:1770/jsp/plantillas/report-display/reportes.dv

Superintendencia de Notariado y Registro  
Ministerio del Interior y de Justicia  
República de Colombia  
www.supernotariado.gov.co

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS MANIZALES - 100  
DIARIO RADICADOR DE TESTAMENTOS  
FECHA 02-09-2011

Impreso el 06 de Octubre de 2011 a las 15:32:29 PM

Página 1

Hora	Turno Ant.	Turno	No Doc.	Fecha	Oficina	Ciudad	Solicitante	Derechos	Recibo	Usuario	Estado
12:28:08		2011-100-6-25042	7081	2011-09-01	NOTARIA SEGUNDA	MANIZALES	NOTARIA SEGUNDA DE MILES	\$ 14,730	60535322	2651	ACTIVO

Total Derechos \$ 14,730  
Total Turnos 1  
Total Turnos Anulados 0

AHORA

Untitled Document - Windows Internet Explorer  
http://192.168.180.106:8080/SNRI/jsp/plantillas/report-display/reportes.dv

Superintendencia de Notariado y Registro  
Ministerio del Interior y de Justicia  
República de Colombia  
www.supernotariado.gov.co

OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS MANIZALES - 100  
DIARIO RADICADOR DE TESTAMENTOS  
FECHA 02-09-2011

Impreso el 06 de Octubre de 2011 a las 16:47:15 PM

Página 1

Hora	Turno Ant.	Turno	No Doc.	Fecha	Oficina	Ciudad	Solicitante	Derechos	Recibo	Usuario	Estado
12:28:08		2011-100-6-25042	7081	2011-09-01	NOTARIA SEGUNDA	MANIZALES	NOTARIA SEGUNDA DE MILES	\$ 14,730	60535322	2651	ACTIVO

Total Derechos Ajustados a la Centena \$ 14,730  
Total Turnos 1  
Total Turnos Anulados 0

**Reporte 157 - Diario Radicador de certificados del día OTRAS ORIPS**

**ANTES**

Hora	Turno	Matrícula	Nombre o Razón Social	Copias	Derechos	Cajero	Recibo	Tipo	Observaciones
15:21:32	2011-314-1-38314	314-1481	RAFAEL RODRIGUEZ	1	12460	52824	56855076	NAC	IMPRESO
15:31:17	2011-314-1-38318	314-30118	LUI PINZON	1	12460	52790	56859947	NAC	IMPRESO
15:39:34	2011-314-1-38318	314-55999	EDWIN OVIDEO	1	12460	52824	56855016	NAC	IMPRESO
16:43:03	2011-314-1-38319	314-30215	LUIS PINCON	1	12460	52790	56859956	NAC	IMPRESO
16:51:52	2011-314-1-38320	314-3075	NOTA 10	1	12460	52790	56859955	NAC	IMPRESO
16:52:28	2011-314-1-38321	314-40895	NOTA 10	1	12460	52790	56859956	NAC	IMPRESO

**Total derechos:** \$685300

Total certificados expedidos: 55  
Total certificados reimprimos: 0  
Total certificados anulados:

**AHORA**

Hora	Turno	Matrícula	Nombre o Razón Social	Copias	Derechos	Cajero	Recibo	Tipo	Observaciones
15:21:32	2011-314-1-38314	314-1481	RAFAEL RODRIGUEZ	1	12460	52824	56855076	NAC	IMPRESO
15:31:17	2011-314-1-38318	314-30118	LUI PINZON	1	12460	52790	56859947	NAC	IMPRESO
15:39:34	2011-314-1-38318	314-55999	EDWIN OVIDEO	1	12460	52824	56855016	NAC	IMPRESO
16:43:03	2011-314-1-38319	314-30215	LUIS PINCON	1	12460	52790	56859956	NAC	IMPRESO
16:51:52	2011-314-1-38320	314-3075	NOTA 10	1	12460	52790	56859955	NAC	IMPRESO
16:52:28	2011-314-1-38321	314-40895	NOTA 10	1	12460	52790	56859956	NAC	IMPRESO

**Total derechos ajustados a la caja:** \$685300

Total certificados expedidos: 55  
Total certificados reimprimos: 0  
Total certificados anulados:

## **022 ACTOS 80-81-30-Tipos de Certificados**

En el proceso de radicación registro se presentan las siguientes inconsistencias en el Aplicativo SIR

- 1) En los turnos de Registro con acto 30-CANCELACION USUFRUCTO con solicitud de certificado asociado, el aplicativo clasifica el turno de certificado como “TIPO\_ASOCIADO\_CESION” siendo lo correcto “ASOCIADO”.
- 2) a) En el proceso de radicación Registro cuando se ingresa el Acto 81 VENTA IDENT. PLANOS CATASTRALES y se genera una solicitud de certificado asociado, se avanza a la pantalla de Registrar Pago se observa que aplica la tarifa correcta para el certificado asociado al Acto 81 de Liquidación (\$2.270.00) y al dar clic en el botón Editar regresar a la pantalla Liquidación y seleccionar nuevamente el botón de Certificados Asociados se elimina la solicitud de

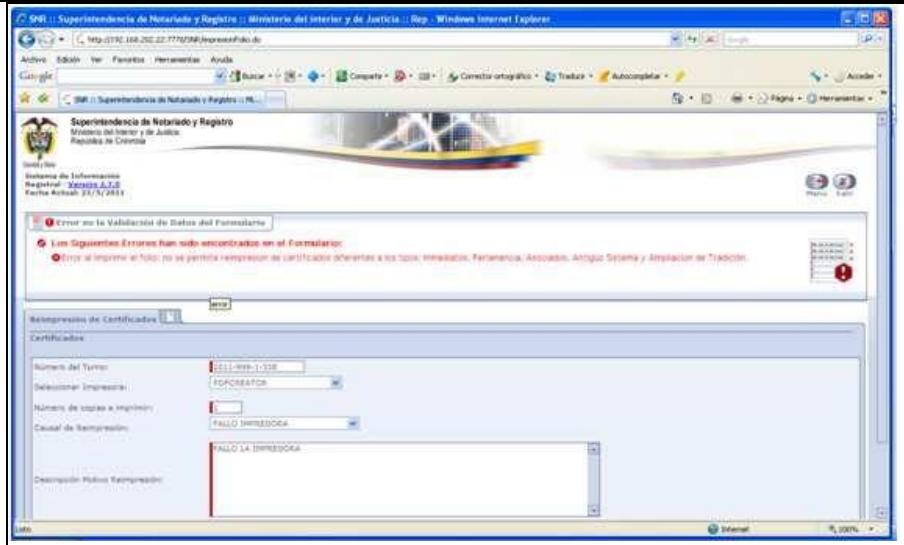
certificado generado, luego se ingresa una solicitud de certificado asociado y se avanza a la pantalla de pago se observa que aplica una tarifa diferente a la establecida (\$12.460.00), lo correcto es que debe aplicar la tarifa por \$2.270.00.oo como lo establece el Artículo 14 – Parágrafo- de la Resolución 069 de 2011, así mismo al generarse el turno de Registro y su respectivo turno de certificado, luego en la pantalla Administrativa al consultar el turno de certificado se observa que el tipo de Certificado que muestra es “ASOCIADO” lo cual es correcto, la inconsistencia se presenta en la aplicación de la tarifa.

- b) En el proceso de radicación Registro cuando se ingresa el Acto 81 VENTA IDENT. PLANOS CATASTRALES y se genera una solicitud de certificado asociado, se avanza a la pantalla de Registrar Pago se observa que aplica la tarifa correcta para el certificado asociado al Acto 81 de Liquidación (\$2.270.00) y se continua con el proceso de radicación sin devolverse a la pantalla anterior ,se genera el turno de registro y su respectivo turno de certificado asociado , luego en la pantalla Administrativa al consultar de turno de certificado se observa que el tipo de Certificado que muestra es “TIPO\_ASOCIADO\_81” lo cual no es correcto , el tipo correcto de Certificado es “ASOCIADO”
- 3) a) En el proceso de radicación Registro cuando se ingresa el Acto 80 CESIÓN DE BIENES FISCALES y se genera una solicitud de certificado asociado, se avanza a la pantalla de Registrar Pago se observa que aplica la tarifa para el certificado asociado al Acto 80 de Liquidación (\$1.130.00) valor de tarifa incorrecto, al dar clic en el botón Editar y luego se regresa a la pantalla Liquidación y selecciona nuevamente el botón de Certificados Asociados se elimina la solicitud de certificado generado, luego se ingresa una solicitud de certificado asociado y se avanza a la pantalla de registrar, pago se observa que aplica una tarifa de (\$12.460.00), lo correcto es que para la tarifa normal debe aplicar **\$ 12.460** desde **el inicio del proceso de la liquidación**, así mismo al generarse el turno de Registro y su respectivo turno de certificado asociado y al consultarse el turno de certificado por pantallas administrativas se observa que el tipo de Certificado que muestra es “ASOCIADO” lo cual es correcto, **la inconsistencia se presenta en la aplicación de la tarifa.**  
b) En el proceso de radicación Registro cuando se ingresa el Acto 80 CESIÓN DE BIENES FISCALES y se genera una solicitud de certificado asociado, y se continua con el proceso hasta generarse el turno de Registro y Certificado Asociado sin haberse regresado a la pantalla de certificados asociados e eliminado solicitudes de certificados, luego en la pantalla Administrativa al consultar el turno de certificado se observa que el tipo de Certificado que muestra es “TIPO\_ASOCIADO\_80” lo cual no es correcto , el tipo correcto de Certificado es “ASOCIADO”

NOTA: La tarifa normal del certificado asociado para el acto 80 de Liquidación –“Cesión de Bienes Fiscales” es de \$12.460.oo, para este certificado no hay tarifa especial.

- 4) En el reporte 74 INFORME DE INGRESOS POR CERTIFICADOS también se observa el tipo de certificado errado.

Esta inconsistencia no permite la reimpresión de los turnos de certificados asociados a turnos de registro que tengan incluidos los actos mencionados en este requerimiento, lo correcto que se categoricen con el tipo de certificado correcto “ASOCIADO”. A continuación mensaje que muestra el sistema.



Respecto al tema de las inconsistencias reportadas por la ORIPS de Aguachica y Villavicencio sobre la reimpresión de turnos de certificados con matrículas de mayor extensión, se realizaron pruebas en ambiente de calidad con los mismos números de matrículas citados en los casos de GLPI y con otros números de matrículas, observándose que el aplicativo realizó el proceso de reimpresión y cambio de matrícula sin inconvenientes, de igual forma se probó en ambiente de producción – Usuario. Pruebas con matrículas de 35000 y 15000 anotaciones, observándose que el aplicativo realizó la reimpresión y cambio de matrícula sin inconvenientes, sin embargo se solicita a la mesa de ayuda realizar seguimiento y en caso de recibir reporte de esta inconsistencia dar solución puntual y general.

Conversación telefónica Yamile Moreno 10/08/2011

Se debe tener en cuenta la tarifa especial solo para un certificado, en el acto

81 - VENTA IDENT. PLANOS CATASTRALES \$2.270

Si se ingresa otro u otros certificado asociado se cobra tarifa normal.

## DESARROLLO

- Script Bases de Datos:
  - Ninguno.
- Creacion De Imagenes:
  - Ninguno.
- Se crearon clases nuevas en SIR:
  - Ninguno.
- Se modificaron los siguientes archivos en SIR:
  - sir.hermode.pagos.LiquidadorCertificados.java
  - sir.hermode.pagos. LiquidadorRegistro.java
  - sir.core.negocio.modelo.Liquidacion.java
  - sir.core.web.acciones.registro.AWLiquidacionRegistro.java

### ARCHIVO JAVA:

sir.hermode.pagos.LiquidadorCertificados.java.

#### Código Insertado Línea 17 - 18:

```
import gov.sir.core.negocio.modelo.constant.CRoles;
import gov.sir.core.negocio.modelo.constant.CTipoCertificado;
```

#### Código Insertado Línea 129 - 135:

```
/**
 * @author Ellery D. Robles Gómez.
 * @change Mantis: 8575 - Se seta el tipo de asociado como ASOCIADO".
 */
if(liqCert.getRol() != null && (CRoles.LIQUIDADOR_REGISTRO).equals(liqCert.getRol().getRoleId())){
```

```
    iqCert.getTipoCertificado().setIdTipoCertificado(CTipoCertificado.TIPO_ASOCIADO_ID);
}
```

#### ARCHIVO JAVA:

```
sir.hermiod.pagos.LiquidadorRegistro.java.
```

#### Código Insertado Línea 306 - 311:

```
/*
 * @author Ellery D. Robles Gómez.
 * @change Mantis: 8575 - Se hacen comentario las líneas siguientes para que no se tengan en cuenta
 * los actos 80 y 30 en la lista de excepciones.
 */
/*
```

#### Código Modificado Línea 332:

```
*/
```

#### Código Insertado Línea 380 - 382:

```
SolicitudAsociada solAsoc = (SolicitudAsociada) it2.next();
SolicitudCertificado solCert = (SolicitudCertificado) solAsoc.getSolicitudHija();
if(!solCert.isEliminar()){
```

#### Código Modificado Línea 434 - 440:

```
/*
 * @author Ellery D. Robles Gómez.
 * @change Mantis: 8575 - Se comentan líneas".
 */
// SolicitudAsociada solAsoc = (SolicitudAsociada) it2.next();
// SolicitudCertificado solCert = (SolicitudCertificado) solAsoc.getSolicitudHija();
if(!solCert.isEliminar()){
```

#### Código Modificado Línea 457:

```
liq.setRol(liqReg.getRol());
```

#### ARCHIVO JAVA:

```
sir.core.negocio.modelo.Liquidacion.java.
```

#### Código Insertado Línea 7:

```
import org.auriga.core.modelo.transferObjects.Rol;
```

#### Código Insertado Línea 24:

```
private Rol rol;
```

#### Código Insertado Línea 26 - 33:

```
public Rol getRol() {
    return rol;
}
public void setRol(Rol rol) {
    this.rol = rol;
}
```

#### ARCHIVO JAVA:

```
sir.core.web.acciones.registro.AWLiquidacionRegistro.java.
```

#### Código Insertado Línea 2531:

```
liquidacion.setRol((Rol) request.getSession().getAttribute(WebKeys.ROL));
```

#### Código Insertado Línea 3209:

```
liquidacion.setRol((Rol) request.getSession().getAttribute(WebKeys.ROL));
```

## 038 Consulta datos para pereira reporte 161

#### CONSULTA DE DATOS PARA PEREIRA.

Construcción de un reporte general en SIR, con parámetros de consulta asociados a Fecha Inicio, fecha fin de la consulta; Departamento, Municipio y Vereda, que al ejecutarse muestre las columnas requeridas por el reporte de la solicitud (Matricula, anotación, naturaleza jurídica, dirección, cuantía, y fecha de radicación). Con este reporte la ORIP de Pereira, o cualquier otra ORIP que lo requiera, puede suministrar la información con la periodicidad que se establezca para la consulta. Esto significa una solución general con base inicial en los requerimientos de Pereira.

#### Especificación de la Consulta:

Parámetros de Entrada para la Consulta – La fecha de Inicio y la Fecha Fin Limita la consulta a las anotaciones cuya fecha de radicación se encuentra entre estas dos fechas.

Fecha Inicio- Fecha de radicación de la tabla sir.sir\_ne\_anotacion (NTCN\_FECHA\_RADICACION)

Fecha Fin - Fecha de radicación de la tabla sir.sir\_ne\_anotacion (NTCN\_FECHA\_RADICACION)

Los parámetros Circulo, Departamento, Vereda, Municipio completaría la selección de la Zona Registral a Reportar (ID\_ZONA\_REGISTRAL = ID\_CIRCULO + ID\_DEPARTAMENTO + ID\_MUNICIPIO + ID\_VEREDA)

#### Columnas a Reportar:

Tabla – Atributo - Nombre  
 Sir.sir\_ne\_folio ID\_MATRICULA Numero de Matricula  
 Sir.sir\_ne\_anotacion ID\_ANOTACION Numero de Anotación  
 Sir.sir\_ne\_naturaleza\_juridica NTJR\_NOMBRE Nombre de Naturaleza Jurídica  
 Sir.sir\_ne\_direccion DRCC\_ESPECIFICACION Dirección (Restricción: Como existe la posibilidad de tener varias direcciones asociadas a un ID\_MATRICULA se listara el máximo ID\_DIRECCION para lograr un único registro de retorno que acompaña al registro consultado)  
 Sir.sir\_ne\_anotacion NTCN\_VALOR Cantidad del Acto  
 Sir.sir\_ne\_anotacion NTCN\_FECHA\_RADICACION Fecha de Radicación.

Este tipo de consulta se limitara mediante las fechas de Inicio y Fin, que corresponden a un rango de fechas de radicación de la tabla sir.sir\_ne\_anotacion. Se realizara una restricción de tiempo de una diferencia no mayor a un mes calendario entre las dos fechas; para asegurar que el volumen de datos a extraer de la base de datos no genere un caída en comunicación (canal de comunicación por transporte de grandes archivos a la ORIP usuaria). Esta es una consulta de gran peso para base de datos debido a que integra dos de las tablas transaccionales más grandes de SIR. Sir.sir\_ne\_folio, sir.sir\_ne\_anotacion. Esta consulta debe ser ejecutada en lo posible en horarios especiales por su costo de proceso.

Este es un desarrollo que tardaría aproximadamente 15 días hábiles. Se desarrolla una forma que captura los parámetros de entrada a la consulta, una consulta de base de datos, un índice adecuado a la consulta, un procedimiento de materialización de un archivo plano separado por comas (csv) compatible con Excel, una nueva forma (jsp) que responda a la consulta con un archivo plano, y una relación con el esquema de roles de sir para su uso y asignación de privilegios.

El formato de salida del archivo será archivo plano separado por comas en formato CSV compatible con herramientas de manejo de Hoja Electrónica (Excel, openOffice)

***Para la liberación de este desarrollo se debe tener en cuenta lo consignado en el acta de despliegue 2.3.0 que a continuación se transcribe:***

“Se realizo prueba generando reporte de la Orip de Andes del 01/08/2009 al 30 de agosto de 2009, se observa que requiere ajustarse lo siguiente : el rango de fecha debe corregirse día, mes, año, la columna fecha debe trasladarse seguido de la columna No. anotación, en el título de fecha debe quedar así : Fecha de Inscripción”, para la selección de búsqueda el Departamento debe ser predeterminado y se deben listar únicamente los municipios que corresponden al círculo registral .Toda vez que este reporte será generado por cada una de las Orips que lo requieran , El sistema debe permitir realizar la búsqueda por el círculo registral dándole como parámetros el municipio que correspondan a este o todos, cuando se seleccione todos no desplegará las veredas, en caso contrario debe seleccionarse un Municipio y desplegar las veredas que corresponden a ese municipio para seleccionar una o todas.”

A demás de lo anterior se requiere adicionar lo siguiente:

Agregar el criterio de selección de Naturaleza Jurídica, el cual debe permitir seleccionar uno, varios o todos para generar el reporte.

El resultado debe mostrarse por Bloques de acuerdo al o los Códigos de Naturaleza Jurídica seleccionados.

Este reporte debe ser asignado al rol del Administrador Regional

El Administrador Nacional podrá generar el reporte para uno, varios o todos los círculos.

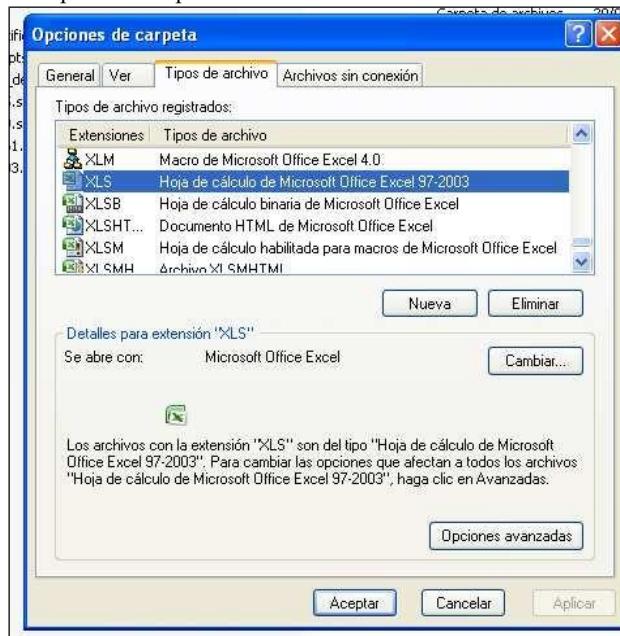
**DESARROLLO:**

Para la generación del Reporte 161 – Actos de Registro en un Rango de Fecha, se recomienda tener Internet Explorer 7 en adelante y Excel 2007.

Para que la generación del XLS (Excel) se abra en Internet Explorer se debe realizar la siguiente configuración en el equipo:



Se despliega ventana, seleccionar pestaña “Tipos de Archivo”



En el listado seleccionar extensión XLS, dar clic en Opciones Avanzadas, se despliega ventana



Seleccionar opción Explorar en la misma ventana y deshabilitar opción Confirmar apertura después de la descarga, dar

clic en aceptar.

NOTA: Esta configuración solo se realizara una vez por maquina.

- Se creó el procedimiento SP\_Reporte\_161

```
create or replace
PROCEDURE SP_Reporte_161
(
  P_FECHA_INICIO IN VARCHAR2,
  P_FECHA_FIN IN VARCHAR2,
  P_CIRCULO IN VARCHAR2,
  P_MUNICIPIO IN VARCHAR2,
  P_VEREDA IN VARCHAR2,
  P_NATURALEZAS IN VARCHAR2,
  SEC_PROCESO IN NUMBER )
IS
/*
***** AHERRENTO 24/09/2009
Reporte 161: ACTOS EN REGISTRO DE UN RANGO DE FECHA */
-----
CURSOR c_act (ID_NATURALEZA VARCHAR2)
IS
SELECT an.id_matricula AS MATRICULA,
       an.ntcn_orden AS ANOTACION,
       na.NTJR_NOMBRE ||'-'||na.id_naturaleza_juridica AS NATURALEZA_JURIDICA,
       (SELECT d.drcce_especificacion
        FROM sir.sir_ne_direccion d
       WHERE d.id_matricula = f.id_matricula
         AND d.id_direccion = (SELECT MAX(d1.id_direccion)
                               FROM sir.sir_ne_direccion d1
                              WHERE d1.id_matricula = f.id_matricula)) AS DIRECCION,
       AN.NTCN_VALOR AS CUANTIA,
       an.ntcn_fecha_radicacion AS FECHA_RADICACION,
       nd.dprt_nombre AS DEPARTAMENTO,
       nm.mnnp_nombre AS MUNICIPIO,
       nv.vrda_nombre AS VEREDA
  FROM sir.sir_ne_folio f INNER JOIN sir.sir_ne_anotacion an ON f.id_matricula = an.id_matricula
 INNER JOIN SIR.SIR_NE_ZONA_REGISTRAL ZR ON F.ID_ZONA_REGISTRAL = ZR.ID_ZONA_REGISTRAL
 INNER JOIN sir.sir_ne_naturaleza_juridica na ON na.id_naturaleza_juridica = an.id_naturaleza_juridica
 INNER JOIN sir.sir_ne_departamento nd ON zr.id_departamento = nd.id_departamento
 INNER JOIN sir.sir_ne_municipio nm ON zr.id_municipio = nm.id_municipio AND zr.id_departamento = nm.id_departamento
 INNER JOIN sir.sir_ne_vereda nv ON zr.id_vereda = nv.id_vereda AND zr.id_departamento = nv.id_departamento AND zr.id_municipio =
 nv.id_municipio
 WHERE an.ntcn_fecha_radicacion >= TO_DATE(P_FECHA_INICIO,'YYYY-MM-DD')
   AND an.ntcn_fecha_radicacion < TO_DATE(P_FECHA_FIN,'YYYY-MM-DD')+1
   AND an.id_naturaleza_juridica = ID_NATURALEZA
   AND f.id_circulo = P_CIRCULO
   AND an.id_circulo = f.id_circulo
   AND zr.id_municipio = DECODE(P_MUNICIPIO, 'TODOS', zr.id_municipio, SUBSTR(P_MUNICIPIO, INSTR(P_MUNICIPIO, '-') + 1, LENGTH(P_MUNICIPIO)))
   AND zr.id_vereda = DECODE(P_VEREDA, 'TODOS', zr.id_vereda, P_VEREDA)
 ORDER BY f.id_matricula, an.id_anotacion;
-----
TYPE t1
IS
TABLE OF c_act%ROWTYPE INDEX BY BINARY_INTEGER;
t_act t1;
NAT_JURIDICA VARCHAR2(50);
NUEVA_CADENA VARCHAR2(1000);
TAM_CADENA NUMBER;
-----
BEGIN
NUEVA_CADENA := P_NATURALEZAS;
TAM_CADENA := LENGTH(NUEVA_CADENA);

WHILE TAM_CADENA >= 1 LOOP
  IF(INSTR(NUEVA_CADENA, ',', 1, 1 )-1 = -1) THEN
    NAT_JURIDICA := NUEVA_CADENA;
    NUEVA_CADENA := NULL;
  ELSE
    NAT_JURIDICA := SUBSTR(NUEVA_CADENA, 1, INSTR(NUEVA_CADENA, ',', 1, 1 )-1);
    NUEVA_CADENA := SUBSTR(NUEVA_CADENA, INSTR(NUEVA_CADENA, ',', 1, 1 )+1, length(NUEVA_CADENA));
  END IF;
END;
```

```

TAM_CADENA := LENGTH(NUEVA_CADENA);
OPEN c_act(NAT_JURIDICA);
LOOP
  FETCH c_act BULK COLLECT INTO t_act LIMIT 1000;
  EXIT WHEN c_act%NOTFOUND AND t_act.COUNT = 0;
  FOR act IN t_act.FIRST..t_act.LAST
  LOOP
    INSERT
    INTO SIR_OP_Reporte_161
    (
      MATRICULA,
      ANOTACION,
      NAT_JURIDICA,
      DIRECCION,
      CUANTIA,
      FECHA,
      DEPARTAMENTO,
      MUNICIPIO,
      VEREDA,
      SEC_PROCESO
    )
    VALUES
    (
      t_act(act).matricula,
      t_act(act).anotacion,
      t_act(act).naturaleza_juridica,
      t_act(act).direccion,
      t_act(act).cuantia,
      t_act(act).fecha_radicacion,
      t_act(act).departamento,
      t_act(act).municipio,
      t_act(act).vereda,
      SEC_PROCESO
    );
  COMMIT;
END LOOP;
END LOOP;
CLOSE c_act;
END LOOP;
END;

```

- Se creó índice para la tabla SIR\_NE\_ANOTACION - IDX\_NTCN\_FECRAD  
`CREATE INDEX "SIR"."IDX_NTCN_FECRAD" ON "SIR"."SIR_NE_ANOTACION" ("NTCN_FECHA_RADICACION")
PCTFREE 10 INITTRANS 2 MAXTRANS 255 COMPUTE STATISTICS
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
TABLESPACE "TBS_SIR_D_1"
PARALLEL 16 ;`

- Se creó la secuencia SEQ\_Reporte\_161  
`CREATE SEQUENCE "SIR"."SEQ_Reporte_161" MINVALUE 1 MAXVALUE 99999999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 2 NOORDER NOCYCLE ;`

- Configuración Reporte 161: Actos de Registro en un Rango de Fecha  
`INSERT INTO SIR_OP_PANTALLA_ADMINISTRATIVA(id_pantalla_administrativa, id_tipo_pantalla, ptdm_nombre, ptdm_vista, ptdm_pagina)
VALUES(1116, 13, 'Reporte 161: Actos de Registro en un Rango de Fecha', 'admin.reportes.report-161.param-form.view', 'PG_REPORTES');`

- Configuración reporte 161  
`INSERT INTO SIR.SIR_OP_ROL_PANTALLA(id_rol,id_pantalla_administrativa)VALUES('SIR_ROL_ADMINISTRADOR REGIONAL',1116);`

- Creación Tabla SIR\_OP\_Reporte\_161  
`CREATE TABLE "SIR"."SIR_OP_Reporte_161"
(
 "MATRICULA" VARCHAR2(100 BYTE),
 "ANOTACION" NUMBER,
 "NAT_JURIDICA" VARCHAR2(255 BYTE),
 "DIRECCION" VARCHAR2(255 BYTE),
 "CUANTIA" NUMBER,
 "FECHA" DATE,
 "SEC_PROCESO" NUMBER,
 "DEPARTAMENTO" VARCHAR2(100 BYTE),
 "MUNICIPIO" VARCHAR2(255 BYTE),`

```

    "VEREDA" VARCHAR2(255 BYTE)
) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT
TABLESPACE "TBS_SIR_D_1";

```

## Configuración Reporte 161 – Actos de Registro en un Rango de Fechas

SIR-APP\web\WEB-INF\jspSeguro\administracion\ADMINREPORTES\_REPORT161\_PARAMFORM

Archivo Nuevo.

SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java

### LINEA 272

```

public static final String REPORTE_161_SEND_ACTION = "REPORTE_161_SEND_ACTION";
public static final String REPORTE_161_BACK_ACTION = "REPORTE_161_BACK_ACTION";

```

### LINEA 437

// AHERREN 22/09/2009

// Reporte 161

```

public static final String REPORTE_161_LOADCIRCULOSBYUSUARIO = "REPORTE_161_LOADCIRCULOSBYUSUARIO";
public static final String REPORTE_161_ONSELECT_CIRCULO_MUNICIPIOS_ACTION =
"REPORTE_161_ONSELECT_CIRCULO_MUNICIPIOS_ACTION";
public static final String REPORTE_161_ONSELECT_ZONAREGISTRAL_MUNICIPIO_ACTION =
"REPORTE_161_ONSELECT_ZONAREGISTRAL_MUNICIPIO_ACTION";
public static final String REPORTE_161_ONSELECT_ZONAREGISTRAL_VEREDA_ACTION =
"REPORTE_161_ONSELECT_ZONAREGISTRAL_VEREDA_ACTION";
public static final String REPORTE_161_ONSELECT_NATURALEZA_JURIDICA_ACTION =
"REPORTE_161_ONSELECT_NATURALEZA_JURIDICA_ACTION";

```

"REPORTE\_161\_ONSELECT\_NATURALEZA\_JURIDICA\_ACTION";

### LINEA 955

// AHERREN 22/09/2009

// Declaración de las Variables para el reporte 161

```

public static final String REPORTE_161_PARAM_CMDKEY      = "reporte-161.jasper";
public static final String REPORTE_161_PARAM_PFECHAINI   = "P_FECHAINI";
public static final String REPORTE_161_PARAM_PFECHAFIN   = "P_FECHAFIN";
public static final String REPORTE_161_PARAM_PCIRCULONOMBRE = "P_CIRCULO";
public static final String REPORTE_161_PARAM_PMUNICIPIO  = "P_MUNICIPIO";
public static final String REPORTE_161_PARAM_PVEREDA     = "P_VEREDA";
public static final String REPORTE_161_PARAM_PNATURIDICA = "P_NATURIDICA";
public static final String REPORTE_161_PARAM_USUARIOLOG  = "P_USUARIO_LOG";

```

### LINEA 1467

//AHERREN 22/09/2009

//Reporte 161

```

if (REPORTE_161_LOADCIRCULOSBYUSUARIO.equals(accion)) {
    return do_ReporteXX_LoadCirculosByUsuario_Action(request);
}

```

```

if (REPORTE_161_ONSELECT_CIRCULO_MUNICIPIOS_ACTION.equals(accion)) {
    try {
        session.removeAttribute(LISTA_OFICINAS_POR_VEREDA);
        salvarDatosReporte161EnSesion(request);
    }

```

return seleccionaDepartamento\_161(request);

```

} catch (Throwable ex) {
    Logger.getLogger(AWReportes.class.getName()).log(Level.SEVERE, null, ex);
}
}

```

```

if (REPORTE_161_ONSELECT_ZONAREGISTRAL_MUNICIPIO_ACTION.equals(accion)) {
    session.removeAttribute(LISTA_OFICINAS_POR_VEREDA);
    salvarDatosReporte161EnSesion(request);
    this.seleccionaZonaRegistralMunicipio(request);
}

```

```

if (REPORTE_161_ONSELECT_ZONAREGISTRAL_VEREDA_ACTION.equals(accion)) {
    this.seleccionaZonaRegistralVereda_161(request);
    salvarDatosReporte161EnSesion(request);
}

```

```

if (REPORTE_161_ONSELECT_NATURALEZA_JURIDICA_ACTION.equals(accion)) {
    salvarDatosReporte161EnSesion(request);
}

```

### LINEA 2216

// AHERREN 22/09/2009

// Reporte 161

```

if (REPORTER_161_SEND_ACTION.equals(accion)) {
    do_Reporte161_Send_Action(request);
}

if (REPORTER_161_BACK_ACTION.equals(accion)) {
    do_Reporte161_Back_Action(request);
}

LINEA 11296
//AHERRENO 22/09/2009
//Reporte 161
private void do_Reporte161_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException, AccionWebException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String[] natJuridicas = request.getParameterValues(CNaturalezaJuridica.ID_NATURALEZA_JURIDICA);
    String cadenaNatJur = null;
    boolean bandera = true;
    if (natJuridicas != null){
        for (int i = 0; i<natJuridicas.length; i++){
            String idNatJuridica = natJuridicas[i];
            if(bandera){
                cadenaNatJur = idNatJuridica;
                bandera = false;
            }else {
                cadenaNatJur += ","+idNatJuridica;
            }
        }
    }

    String param_01 = request.getParameter(REPORTER_161_PARAM_PFECHAINI);
    String param_02 = request.getParameter(REPORTER_161_PARAM_PFECHAFIN);
    String param_03 = request.getParameter(REPORTER_161_PARAM_PCIRCULONOMBRE);
    String param_04 = (String)request.getSession().getAttribute(CMunicipio.ID_MUNICIPIO);
    String param_05 = (String)request.getSession().getAttribute(CVereda.ID_VEREDA);
    String param_06 = cadenaNatJur;
    String param_07 = request.getParameter(REPORTER_161_PARAM_USUARIOLOG);

    GregorianCalendar capturaFecha1 = null;
    GregorianCalendar capturaFecha2 = null;
    int ano1 = 0;
    int ano2 = 0;
    int mes1 = 0;
    int mes2 = 0;
    int dia1 = 0;
    int dia2 = 0;
    //*****//****

    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {
        exception.addError("parametro: Fecha Inicio Radicación: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;

            String[] fechas = param_01.split("//yyyy-MM-dd - 0-1-2"
            ano1 = Integer.parseInt(fechas[0]);
            mes1 = Integer.parseInt(fechas[1]);
            dia1 = Integer.parseInt(fechas[2]);
            capturaFecha1 = new GregorianCalendar(ano1,mes1,dia1);

        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio Radicación: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Fecha Fin Radicación: Debe digitar un valor");
    }
}

```

```

} else {
    String send1 = null;
    Date fecha = null;
    try {
        fecha = DateFormatUtil.parse(param_02);
        send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
        param_02 = send1;

        String[] fechas = param_02.split("-");//yyyy-MM-dd - 0-1-2
        ano2 = Integer.parseInt(fechas[0]);
        mes2 = Integer.parseInt(fechas[1]);
        dia2 = Integer.parseInt(fechas[2]);
        capturaFecha2 = new GregorianCalendar(ano2,mes2-1,dia2); //Los meses van de 0 a 11

    } catch (Exception e) {
        exception.addError("parametro: Fecha Fin Radicación: La fecha es inválida" + e.getMessage());
    }
}

if(capturaFecha1 != null && capturaFecha2 != null){
    if(!(mes1 == mes2 && ano1 == ano2)) {

        Calendar valFec = Calendar.getInstance();
        valFec.set(ano1, mes1-1, dia1, 0, 0, 0); //Los meses van de 0 a 11
        valFec.add(Calendar.DATE, 30);
        System.out.println(valFec.getTime());
        System.out.println(capturaFecha2.getTime());

        if( capturaFecha2.after(valFec)){
            exception.addError("El Rango de fechas no puede exceder a un mes");
        }
    }
}

// apply validators
if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {
    exception.addError("parametro: Circulo: Debe digitar un valor");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
        exception.addError("parametro: Circulo: Debe Seleccionar un Circulo");
    }
}

if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
    exception.addError("parametro: Municipio: Debe Seleccionar un Municipio");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
        exception.addError("parametro: Municipio: Debe Seleccionar un Municipio");
    }
}

if ((null == param_05) || ("".equalsIgnoreCase(param_05))) {
    exception.addError("parametro: Vereda: Debe Seleccionar una Vereda");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_05)) {
        exception.addError("parametro: Vereda: Debe Seleccionar una Vereda");
    }
}

if ((null == param_06) || ("".equalsIgnoreCase(param_06))) {
    exception.addError("parametro: Naturaleza Juridica: Debe Seleccionar una Naturaleza Juridica");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_06)) {
        exception.addError("parametro: Naturaleza Juridica: Debe Seleccionar una Naturaleza Juridica");
    }
}

if ((null == param_07) || ("".equalsIgnoreCase(param_07))) {
    exception.addError("parametro: Usuario Logueado: Usuario no encontrado");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_07)) {
        exception.addError("parametro: Usuario Logueado: Usuario no encontrado");
    }
}
do_Reporte161_SaveState(request);

```

```

// raise -----
if(exception.getErrores().size() > 0) {
    throw exception;
}
// -----
String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY, REPORTE_161_PARAM_PFECHAINI, REPORTE_161_PARAM_PFECHAFIN,
    REPORTE_161_PARAM_PCIRCULONOMBRE, REPORTE_161_PARAM_PMUNICIPIO, REPORTE_161_PARAM_PVEREDA,
    REPORTE_161_PARAM_PNATJURIDICA, REPORTE_161_PARAM_USUARIOLOG
};
String[] values = new String[]{
    REPORTE_161_PARAM_CMDKEY, param_01, param_02, param_03, param_04, param_05, param_06, param_07
};

String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_Reporte161_RemoveState(request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

private void do_Reporte161_Back_Action(HttpServletRequest request) {

do_Reporte161_RemoveState(request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);

} // :do_Reporte161_Back_Action

private void do_Reporte161_SaveState(HttpServletRequest request) {
HttpSession session;
String[] itemIds = new String[]{
    REPORTE_161_PARAM_PFECHAINI, REPORTE_161_PARAM_PFECHAFIN, REPORTE_161_PARAM_PCIRCULONOMBRE,
    REPORTE_161_PARAM_PMUNICIPIO, REPORTE_161_PARAM_PVEREDA, REPORTE_161_PARAM_PNATJURIDICA,
    REPORTE_161_PARAM_USUARIOLOG
};

session = request.getSession();
save_PageItemsState(itemIds, request, session);

} // :do_Reporte161_SaveState

private void do_Reporte161_RemoveState(HttpServletRequest request) {

HttpSession session;

String[] itemIds = new String[]{
    REPORTE_161_PARAM_PFECHAINI, REPORTE_161_PARAM_PFECHAFIN, REPORTE_161_PARAM_PCIRCULONOMBRE,
    REPORTE_161_PARAM_PMUNICIPIO, REPORTE_161_PARAM_PVEREDA, REPORTE_161_PARAM_PNATJURIDICA,
    REPORTE_161_PARAM_USUARIOLOG
};

session = request.getSession();
delete_PageItemsState(itemIds, request, session);
session.removeAttribute(CMunicipio.ID_MUNICIPIO);
session.removeAttribute(CVereda.ID_VEREDA);
session.removeAttribute(CNaturalezaJuridica.ID_NATURALEZA_JURIDICA);

// DELETE LISTS OF VALUES IN CACHE
itemIds = new String[]{
    AWReportes.LISTA_DEPARTAMENTOS, AWReportes.LISTA_MUNICIPIOS, AWReportes.LISTA_VEREDAS
};
delete_PageItemsState(itemIds, request, session);

} // :do_Reporte161_RemoveState

LINEA 12444
//AHERRENO 23/09/2009
//Reporte 161
//salvarDatos
private void salvarDatosReporte161EnSesion(HttpServletRequest request) {

HttpSession session = request.getSession();
String FechaInicial = request.getParameter(REPORTE_161_PARAM_PFECHAINI);

```

```

        if (FechaInicial != null && !FechaInicial.trim().equals("")) {
            session.setAttribute(REPORTE_161_PARAM_PFECHAINI, FechaInicial);
        }

        String FechaFinal = request.getParameter(REPORTE_161_PARAM_PFECHAFIN);
        if (FechaFinal != null && !FechaFinal.trim().equals("")) {
            session.setAttribute(REPORTE_161_PARAM_PFECHAFIN, FechaFinal);
        }

        String circulo = request.getParameter(REPORTE_161_PARAM_PCIRCULONOMBRE);
        if (circulo != null && !circulo.trim().equals("")) {
            session.setAttribute(REPORTE_161_PARAM_PCIRCULONOMBRE, circulo);
        }
    }

LINEA 12515
/** AHERRENO 07/10/2010
 * Este método se encarga de tomar los valores del <code>HttpServletRequest</code> para
 * generar un evento para la selección del departamento del circulo seleccionado
 * @param request
 * @return evento <code>EvnAdministracionForseti</code> con la información del departamento
 * @throws AccionWebException
 */

private EvnAdministracionForseti seleccionaDepartamento_161(HttpServletRequest request)
    throws AccionWebException, ServicioNoEncontradoException, Throwable {

    HttpSession session = request.getSession();
    ServletContext context = session.getServletContext();

    Vereda idVereda = null;
    String idMunicipio = null;
    String idDepartamento = null;
    Map mapMunicipios = Collections.synchronizedMap(new TreeMap());

    String circuloCap = request.getParameter(AWReportes.REPORTE_161_PARAM_PCIRCULONOMBRE);
    List zonaR = null;

    if ((circuloCap == null) || (circuloCap.trim().length() == 0)) {
        throw new AccionInvalidaException("Debe Proporcionar un Código de Circulo");
    }

    service = ServiceLocator.getInstancia();
    try {
        forseti = (ForsetiServiceInterface) service.getServicio("gov.sir.forseti");
        if (forseti == null) {
            throw new ServicioNoEncontradoException("Servicio forseti no encontrado");
        }
        CirculoPk circuloPk = new CirculoPk();
        circuloPk.idCirculo = circuloCap;
        zonaR = forseti.getZonaRegistralesCirculo(circuloPk);

        for(Iterator itZona = zonaR.iterator(); itZona.hasNext();){
            ZonaRegistral zonaRegis = (ZonaRegistral) itZona.next();
            idVereda = zonaRegis.getVereda();
            idMunicipio = idVereda.getIdMunicipio();
            idDepartamento = idVereda.getIdDepartamento();
            Map map = (Map) context.getAttribute(WebKeys.MAPA_DEPARTAMENTOS);
            Departamento depto = null;
            List listaMunDep = new ArrayList();

            for (Iterator itDep = map.keySet().iterator(); itDep.hasNext();){
                String key = (String) itDep.next();
                depto = (Departamento) map.get(key);
                if(depto.getIdDepartamento().equals(idDepartamento)){
                    listaMunDep = depto.getMunicipios();
                    break;
                }
            }

            for (Iterator itMun = listaMunDep.iterator(); itMun.hasNext();){
                Municipio municipio = (Municipio) itMun.next();
                String llave = municipio.getNombre() + "-" + municipio.getIdMunicipio();
                if(municipio.getIdMunicipio().equals(idMunicipio) && municipio.getIdDepartamento().equals(idDepartamento)){
                    mapMunicipios.put(llave, municipio);
                    break;
                }
            }
        }
    }
}

```

```

        }
    }

}

} catch (ServiceLocatorException e) {
    throw new ServicioNoEncontradoException("Error instanciando el servicio forseti");
}

session.setAttribute(AWReportes.MAP_MUNICIPIOS, mapMunicipios);

List listaMunicipios = new ArrayList();
for (Iterator itMun = mapMunicipios.keySet().iterator(); itMun.hasNext();) {
    String key = (String) itMun.next();
    Municipio municipio = (Municipio) mapMunicipios.get(key);
    listaMunicipios.add(new ElementoLista(key, municipio.getNombre()));
}

if (listaMunicipios.size() > 0) {
    listaMunicipios.add(new ElementoLista("TODOS", " TODOS"));
}

session.setAttribute(AWReportes.LISTA_MUNICIPIOS, listaMunicipios);
session.removeAttribute(AWReportes.LISTA_VEREDAS);
session.removeAttribute(CMunicipio.ID_MUNICIPIO);
return null;
}

```

#### LINEA 12694

```

/*AHERRENO 23/09/2009
 * Este método se encarga de tomar los valores del <code>HttpServletRequest</code> para
 * generar un evento para la adición de una vereda
 * @param request
 * @return evento <code>EvnAdministracionForseti</code>
 * @throws AccionWebException
 **/

private EvnAdministracionForseti seleccionaZonaRegistralVereda_161(HttpServletRequest request)
    throws AccionWebException {
    HttpSession session = request.getSession();
    salvarDatosOficinaOrigenEnSesion(request);
    ValidacionOficinaOrigenException exception = new ValidacionOficinaOrigenException();

    String idMunicipio = request.getParameter(CMunicipio.ID_MUNICIPIO);
    if ((idMunicipio == null) || (idMunicipio.trim().length() == 0) || idMunicipio.equals(WebKeys.SIN_SELECCIONAR)) {
        exception.addError("Debe Seleccionar un Municipio.");
    } else {
        session.setAttribute(CMunicipio.ID_MUNICIPIO, idMunicipio);
    }

    String idVereda = request.getParameter(CVereda.ID_VEREDA);
    if ((idVereda == null) || (idVereda.trim().length() == 0) || idVereda.equals(WebKeys.SIN_SELECCIONAR)) {
        exception.addError("No se encontró la cabecera municipal");
    } else {
        session.setAttribute(CVereda.ID_VEREDA, idVereda);
    }
    if (exception.getErrores().size() > 0) {
        throw exception;
    }
    //session.setAttribute(CVereda.ID_VEREDA, idVereda);
    //Usuario usuario = (Usuario) session.getAttribute(SMARTKeys.USUARIO_EN_SESION);
    return null;
}

```

#### JASPER - reporte-161.jasper

Archivo Nuevo.

SIR-APP\src\gov\sir\core\web\navegacion\administracion\ControlNavegacionReportes.java

#### LINEA 294

```

//AHERRENO 22/09/2009
//Reporte 161
public static final String REPORTE_161_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK =
    "REPORTE_161_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK";
// report-161 - graph
protected static final String REPORTE_161_ONSELECT_CIRCULO_MUNICIPIOS_EVENTRESP_OK =
"REPORTE_161_ONSELECT_CIRCULO_MUNICIPIOS_EVENTRESP_OK";
protected static final String REPORTE_161_ONSELECT_ZONAREGISTRAL_MUNICIPIO_EVENTRESP_OK =
"REPORTE_161_ONSELECT_ZONAREGISTRAL_MUNICIPIO_EVENTRESP_OK";

```

```

protected static final String REPORTE_161_ONSELECT_ZONAREGISTRAL_VEREDA_EVENTRESP_OK = "REPORTE_161_ONSELECT_ZONAREGISTRAL_VEREDA_EVENTRESP_OK";
protected static final String REPORTE_161_ONSELECT_NATURALEZA_JURIDICA_EVENTRESP_OK = "REPORTE_161_ONSELECT_NATURALEZA_JURIDICA_EVENTRESP_OK";

```

**LINEA 425**

```

//AHERRENO 22/09/2009
//Reporte 161
listaReportesJasper.add(AWReportes.REPORTE_161_SEND_ACTION);

```

**LINEA 904**

```

//AHERRENO 22/09/2009
// -- navegacion sobre pagina
if( AWReportes.REPORTE_161_LOADCIRCULOSBYUSUARIO.equals( accion ) ){
    return REPORTE_161_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK;
}
if( AWReportes.REPORTE_161_ONSELECT_CIRCULO_MUNICIPIOS_ACTION.equals( accion ) ) {
    return REPORTE_161_ONSELECT_CIRCULO_MUNICIPIOS_EVENTRESP_OK;
}
if( AWReportes.REPORTE_161_ONSELECT_ZONAREGISTRAL_MUNICIPIO_ACTION.equals( accion ) ) {
    return REPORTE_161_ONSELECT_ZONAREGISTRAL_MUNICIPIO_EVENTRESP_OK;
}
if( AWReportes.REPORTE_161_ONSELECT_ZONAREGISTRAL_VEREDA_ACTION.equals( accion ) ) {
    return REPORTE_161_ONSELECT_ZONAREGISTRAL_VEREDA_EVENTRESP_OK;
}
if( AWReportes.REPORTE_161_ONSELECT_NATURALEZA_JURIDICA_ACTION.equals( accion ) ) {
    return REPORTE_161_ONSELECT_NATURALEZA_JURIDICA_EVENTRESP_OK;
}

```

**LINEA 1164**

```

//AHERRENO 22/09/2009
//Reporte 161
if( AWReportes.REPORTE_161_BACK_ACTION.equals( accion ) ) {
    return REPORTE_XX_BACK_EVENTRESP_OK;
}

```

**SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDORreportesJasperDAO.java**

**LINEA 329**

```

//Reporte 161
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_161_PARAM_PFECHAINI);
listaParametros.add(AWReportes.REPORTE_161_PARAM_PFECHAFIN);
listaParametros.add(AWReportes.REPORTE_161_PARAM_PCIRCULONOMBRE);
listaParametros.add(AWReportes.REPORTE_161_PARAM_PMUNICIPIO);
listaParametros.add(AWReportes.REPORTE_161_PARAM_PVEREDA);
listaParametros.add(AWReportes.REPORTE_161_PARAM_PNATJURIDICA);
mapaReportes.put(AWReportes.REPORTE_161_PARAM_CMDKEY, new Object[]{
    "SELECT seq_reporte_161.NEXTVAL FROM dual",
    "sp_reporte_161(?, ?, ?, ?, ?, ?)", listaParametros,
    "DELETE FROM sir_op_reporte_161 WHERE sec_proceso = "});

```

**LINEA 479**

```

/*AHERRENO 28/09/2009*/
// Se lee el fichero de propiedades "gov.sir.print.properties" con los nombres de los reportes en excel.
// El listado de reportes debe estar separado por comas en el fichero.
// Se verifica si el nombre del reporte a ejecutar se encuentra en el fichero.
// Si el nombre del reporte se encuentra en el fichero se renderiza el reporte en formato excel
// de lo contrario en formato PDF.

```

```

PrintJobsProperties prop;
prop = PrintJobsProperties.getInstancia();
String[] reportesExcel = (prop.getProperty(PrintJobsProperties.REPORTES_EXCEL)).split(",");
int tam = reportesExcel.length;
boolean isExcel = false;

for(int i = 0; i < tam; i++){
    if(nombreReporte.equals(reportesExcel[i]))
        isExcel = true;
}
if (isExcel) {

    ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
    JRXLsExporter exporterXLS = new JRXLsExporter();
    exporterXLS.setParameter(JRXLsExporterParameter.JASPER_PRINT,jasperPrint);
    exporterXLS.setParameter(JRXLsExporterParameter.OUTPUT_STREAM,byteArrayOutputStream);
    exporterXLS.exportReport();
    rta = byteArrayOutputStream.toByteArray();
}

```

**SIR-APP\src\org.auriga.smart.vistas.xml**

<pre> <b>LINEA 4568</b> &lt;!-- AHERRENO 22/09/2009    report-161 --&gt; &lt;/vista&gt; &lt;autenticacion&gt;true&lt;/autenticacion&gt; &lt;excepcion&gt;gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector&lt;/excepcion&gt; &lt;excepcion&gt;gov.sir.core.web.acciones.excepciones.ValidacionParametrosException&lt;/excepcion&gt; &lt;nombre&gt;admin.reportes.report-161.param-form.view&lt;/nombre&gt; &lt;parametro&gt;&lt;llave&gt;error&lt;/llave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/ERROR_MULTIPLE.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;parametro&gt;&lt;llave&gt;head&lt;/llave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/HEADER_LOGOUT.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;parametro&gt;&lt;llave&gt;body&lt;/llave&gt;&lt;valor&gt;/WEB- INF/jspSeguro/administracion/ADMINREPORTES_REPORT161_PARAMFORM.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;parametro&gt;&lt;llave&gt;bottom&lt;/llave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/FOOTER_DEFAULT.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;/vista&gt;</pre>	<b>SIR-APP\src\org.auriga.smart.web.xml</b>
<b>LINEA 875</b> <!-- AHERRENO 22/09/2009    report-161 --> <navegacion>   <resultado>REPORTE_161_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK</resultado>   <url-redireccion>admin.reportes.report-161.param-form.view</url-redireccion> </navegacion> <navegacion>   <resultado>REPORTE_161_ONSELECT_CIRCULO_MUNICIPIOS_EVENTRESP_OK</resultado>   <url-redireccion>admin.reportes.report-161.param-form.view</url-redireccion> </navegacion> <navegacion>   <resultado>REPORTE_161__ONSELECT_ZONAREGISTRAL_MUNICIPIO_EVENTRESP_OK</resultado>   <url-redireccion>admin.reportes.report-161.param-form.view</url-redireccion> </navegacion> <navegacion>   <resultado>REPORTE_161_ONSELECT_ZONAREGISTRAL_VEREDA_EVENTRESP_OK</resultado>   <url-redireccion>admin.reportes.report-161.param-form.view</url-redireccion> </navegacion> <navegacion>   <resultado>REPORTE_161_ONSELECT_NATURALEZA_JURIDICA_EVENTRESP_OK</resultado>   <url-redireccion>admin.reportes.report-161.param-form.view</url-redireccion> </navegacion>	
<b>mapas.jar</b>	
<ul style="list-style-type: none"> <li>● <a href="#">org.auriga.smart.vistas.xml</a></li> <li>● <a href="#">org.auriga.smart.web.xml</a></li> </ul>	

<b>DESPLIEGUE – 2.7.2</b> <b>16/01/2012</b>
<b>036 Impresión de Folios</b>
<p><b>REQUERIMIENTO</b></p> <p>Con el fin de realizar actividades propias del área informática se solicita incluir dentro de las pantallas administrativas, la impresión de simple de folio para todos los círculos a los usuarios de los funcionarios Héctor Elejalde, Imelda Muñoz, Luis Emilio Romero, Bibia Marisol Pulido, Yamile Moreno, Rodrigo Pérez, Dora Esperanza Pachón, Ricardo Parra, Digna Isabel Quiroz, Cilia Mary Castro, Miguel Galvis.</p> <p>Así mismo se solicita la creación de un usuario para el funcionario Nelson Rodríguez Nope identificado con C.C.. #79.868.512, de la Oficina Informática con los mismos roles con que cuentan los usuarios de los funcionarios mencionados.</p> <p>Validaciones a realizar: Que únicamente los usuarios mencionados puedan realizar las impresiones simples de los diferentes círculos, los demás deben quedar como están actualmente.</p> <p>Grupo: SNR</p>

### Análisis Requerimiento

Para permitir que algunos usuarios puedan realizar Impresión de Folios de cualquier circulo se decide crear un nuevo Rol que les permitirá a los usuarios que tienen permiso sobre la pantalla administrativa Impresión Simple de Folio realizar esta operación.

Característica del ROL:

- \* No aparece en las lista de Roles que carga a los usuarios.
- \* Permite Impresión de Folios de cualquier Circulo a aquellos usuarios que tengan permiso sobre la pantalla administrativa IMPRESION->Impresión Simple de Folio.

Script Bases de Datos:

Creacion de Rol rol.sql.

Creacion De Imagenes:

Ninguno

Se modificaron los siguientes archivos:

Ninguno

#### CLASE:

**gov.sir.core.web.acciones.administracion.AWImpresionFolio.java**

#### Método:

**private EvnImprimirFolio imprimirFolio(HttpServletRequest  
request) throws AccionWebException {**

#### Código Insertado Línea 206:

```
/**  
 * @author : Julio Alcazar  
 * @change : Se agrega el Circulo Actual y los Roles del Usuario al evento.  
 * Caso Mantis : 0009044: Acta - Requerimiento No 036_151 - Impresión de Folios  
 */  
Circulo circuloActual = (Circulo) session.getAttribute(WebKeys.CIRCULO);  
List roles=(List)session.getAttribute(WebKeys.LISTA_ROLES);
```

#### Código Eliminado Lineas 219:

```
/**  
 * @author : Julio Alcazar  
 * @change : se elimina el código del caso mantis 8005 para organizarlo en el  
negocio.  
 * Caso Mantis : 0009044: Acta - Requerimiento No 036_151 - Impresión de Folios  
 */
```

#### Código Insertado Línea 237:

```
/**  
 * @author : Julio Alcazar  
 * @change : Se agrega el Circulo Actual y los Roles del Usuario al evento.  
 * Caso Mantis : 0009044: Acta - Requerimiento No 036_151 - Impresión de Folios  
 */  
evento.setCirculo(circuloActual);  
evento.setRoles(roles);
```

#### CLASE:

**gov.sir.core.negocio.modelo.constantes.CRoles.java**

**Código Insertado Línea 69:**

```
/**  
 * @author : Julio Alcazar  
 * @change : Se agrega nuevo rol.  
 * Caso Mantis : 0009044: Acta - Requerimiento No 036_151 - Impresión de Folios  
 */  
public static final String SIR_ROL_IMPRESION_FOLIO_ESPECIAL =  
"SIR_ROL_IMPRESION_FOLIO_ESPECIAL";
```

**CLASE:****gov.sir.core.negocio.acciones.administracion.ANImprimirFolio.jav****a****Método:****private EvnRespImprimirFolio imprimirFolio(EvnImprimirFolio  
evento) throws EventoException {****Código Insertado Línea 605:**

```
/**  
 * @author : Julio Alcazar  
 * @change : nuevas variables para el flujo.  
 * Caso Mantis : 0009044: Acta - Requerimiento No 036_151 - Impresión de Folios  
 */  
Rol rol = null;  
boolean folioimpresioespecial = false;
```

**Código Insertado Línea 622:**

```
/**  
 * @author : Julio Alcazar  
 * @change : Se determina si el usuario tiene el rol  
SIR_ROL_IMPRESION_FOLIO_ESPECIAL para permitirle o no, imprimir  
* folios de otros circulos.  
* Caso Mantis : 0009044: Acta - Requerimiento No 036_151 - Impresión de Folios  
*/  
List roles = evento.getRoles();  
Iterator itRoles = roles.iterator();  
while(itRoles.hasNext()){  
    rol = (Rol)itRoles.next();  
    if(rol.getRoleId().equals(CRoles.SIR_ROL_IMPRESION_FOLIO_ESPECIAL)){  
        folioimpresioespecial = true;  
        break;  
    }  
}  
if(!folioimpresioespecial){  
    Circulo circuloactual = evento.getCirculo();  
    if(!circuloactual.getIdCirculo().equals(folioSimple.getCirculo())){  
        throw new EventoException("No se puede Imprimir Folios de otros circulos.");  
    }  
}
```

**CLASE:****gov.sir.core.web.helpers.comun.RolHelper.java****Método:****public void drawGUI(HttpServletRequest request, JspWriter out)  
throws HelperException, IOException {**

**Código Insertado Línea 91:**

```
/**  
 * @author : Julio Alcazar  
 * @change : NO se visualiza el rol SIR_ROL_IMPRESION_FOLIO_ESPECIAL de las  
listas de roles-procesos del usuario.  
 * Caso Mantis : 0009044: Acta - Requerimiento No 036_151 - Impresión de Folios  
*/  
&& (!rol.getRoleId().equals(CRoles.SIR_ROL_IMPRESION_FOLIO_ESPECIAL))}{
```

**DESPLIEGUE – 2.7.1****29/07/2011****058 Avance de Turnos  
(Informativo)****REQUERIMIENTO**

Al tratar de avanzarlos queden en la misma fase y la misma estación, se presentan varios turnos con este inconveniente donde el avance del turno indica que es exitoso pero el turno realmente no avanza.

**DESARROLLO**

Se realizaron, pruebas en los ambientes de desarrollo y producción, y los inconvenientes mencionados en el requerimiento no se detectaron por lo tanto no se logró generar una solución detallada.

En la revisión del código se evidenció que cada uno de los modelos utilizados para realizar el avance de turnos de una a fase a otra, podrían presentar algunos inconvenientes pero que su modificación conllevaría a mejoras de performance y cambios generales de la aplicación. Estos cambios se deberían enfocar a transacciones atómicas (los cambios o bien ocurren todos o ninguno), los cuales deben ser detallados como cambio de alto impacto de aplicación.

## 059 Asignacion de Estaciones a Turnos Informativo

### **REQUERIMIENTO**

Asignación de Estaciones a Turnos. Frecuentemente se presenta el caso, donde turnos de cualquier proceso quedan sin fases ni estaciones, estos turnos después de su primera fase que es la fase solicitud, quedan perdidos teniendo en cuenta que no llegan a la fase siguiente.

En algunas ocasiones al radicar un turno, no avanza a la siguiente fase y queda sin fase ni estaciones y en su historial solo queda la fase solicitud.

Se requiere escalar este caso a desarrollo para solucionar este inconveniente ya que se presenta con frecuencia en todos los círculos. Los turnos siempre deben avanzar a la siguiente fase según el flujo del proceso que corresponda a cada turno. Referencia Id Glpi 11295.

### **DESARROLLO**

Se realizaron, pruebas en los ambientes de desarrollo y producción, y los inconvenientes mencionados en el requerimiento no se detectaron por lo tanto no se logró generar una solución detallada.

En la revisión del código se evidenció que cada uno de los modelos utilizados para realizar el avance de turnos de una a fase a otra, podrían presentar algunos inconvenientes pero que su modificación conllevaría a mejoras de performance y cambios generales de la aplicación. Estos cambios se deberían enfocar a transacciones atómicas (los cambios o bien ocurren todos o ninguno), los cuales deben ser detallados como cambio de alto impacto de aplicación.

## 146 Rango de fechas de Reportes No 88\_04\_90\_89\_57

### **REQUERIMIENTO**

Para los REPORTES: “# 88 Certificados Entregados en el día, # 04 Diario desanotador de documentos del día, # 90 Diario Desanotador de Certificados del día y # 89 listado de documentos entregados en el día, # 57 Diario radicador de matrículas” se requiere que el aplicativo permita seleccionar el rango de fecha que desea obtener la información del reporte “DESDE ----- HASTA “Además se requiere que el sistema totalice el numero de turnos listados en estos reportes.

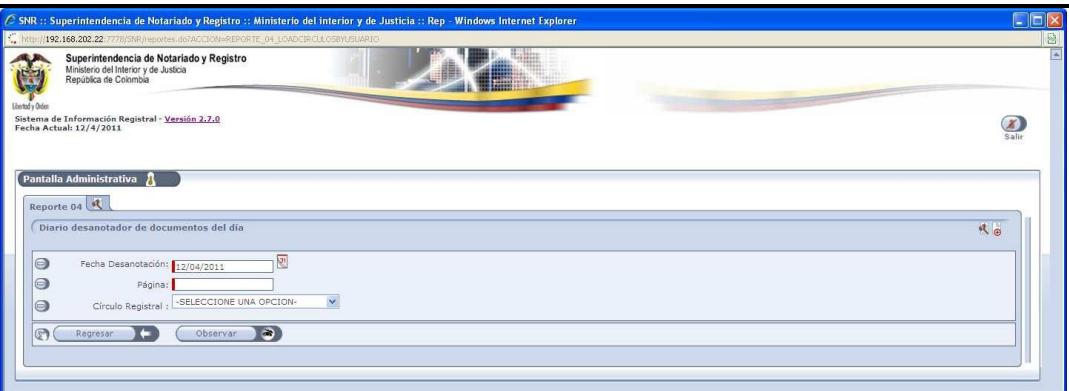
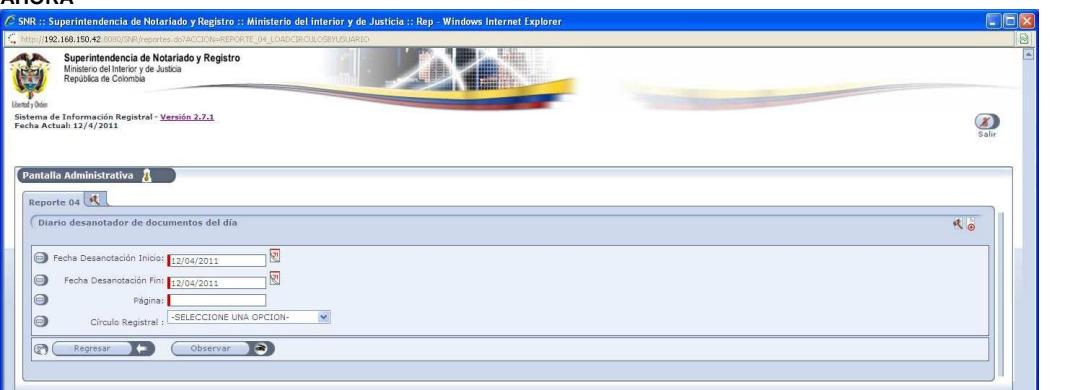
### **DESARROLLO**

#### **Configuración Reporte 04 – DIARIO DESANOTADOR DE DOCUMENTOS DEL DÍA**

Pantalla Reporte

ANTES

**AHORA**

```

SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES_REPORT04_PARAMFORM

<!--AHERRENO 11/04/2011
Se agrega campo Fecha desanotacion Fin--&gt;
&lt;tr&gt;
&lt;td width="20"&gt;&lt;img src="<%=&gt;request.getContextPath()%&gt;/jsp/images/ind_campotxt.gif" width="20"
height="15"&gt;&lt;/td&gt;
&lt;td align="left" width="150"&gt;&lt;div align="right" &gt;Fecha Desanotaci&amp;oacute;n Fin:&lt;/div&gt;&lt;/td&gt;
&lt;td width="200"&gt;
&lt;%
try {
    if( null == session.getAttribute( AWReportes.REPORTE_04_PARAM_PFECHADESANOTACIONFIN ) ) {
        session.setAttribute( AWReportes.REPORTE_04_PARAM_PFECHADESANOTACIONFIN, fechaAct );
    }
    TextHelper textHelper = new TextHelper();
    textHelper.setId( AWReportes.REPORTE_04_PARAM_PFECHADESANOTACIONFIN );
    textHelper.setNombre( AWReportes.REPORTE_04_PARAM_PFECHADESANOTACIONFIN );
    textHelper.setCssClase("camposformtext");
    textHelper.render(request,out);
}
catch(HelperException re){
    out.println("ERROR " + re.getMessage());
}
%&gt;
&lt;a href="javascript:NewCal('&lt;%=AWReportes.REPORTE_04__PARAM_PFECHADESANOTACIONFIN%&gt;','ddmmmyyyy',true,24)"&gt;&lt;img src="<%=&gt;request.getContextPath()%&gt;/jsp/images/ico_calendario.gif" alt="Fecha" width="15" height="21" border="0" onClick="javascript:Valores('&lt;%=request.getContextPath()%&gt;')"/&gt;&lt;/a&gt;
&lt;/td&gt;&amp;nbsp;&lt;/td&gt;
&lt;/tr&gt;

<!--AHERRENO 11/04/2011
Se captura el usuario logueado--&gt;
&lt;%
Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
%&gt;
&lt;input type="hidden" name="<%=&gt;AWReportes.REPORTE_04_PARAM_USUARIOLOG%" id="<%=&gt;AWReportes.REPORTE_04_PARAM_USUARIOLOG%" value="<%=&gt;usuario.getUsername()%&gt;"&gt;
</pre>


SIR-APP\src\gov\lsir\core\web\acciones\administracion\AwReportes.java


```

```

/* Reporte 04 - DIARIO DESANOTADOR DE DOCUMENTOS DEL DÍA
 * AHERRENO 11/04/2011
 * Se agregan variables: Fecha Final y Usuario que genera el reporte
 */
private void do_Reporte04_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_04_PARAM_PFECHADESANOTACION);
    String param_02 = request.getParameter(REPORTE_04_PARAM_PCIRCULONOMBRE);
    String param_03 = request.getParameter(REPORTE_04_PARAM_PNUMPAGINA);
    String param_04 = request.getParameter(REPORTE_04_PARAM_PFECHADESANOTACIONFIN);
    String param_05 = request.getParameter(REPORTE_04_PARAM_USUARIOLOG);

    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {
        exception.addError("parametro: Fecha Desanotacion Inicio: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Desanotacion Inicio: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_02)) {
            exception.addError("parametro: Circulo: Debe digitar un valor");
        }
    }

    // apply validators
    if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {
        exception.addError("parametro: No. de Página: Debe digitar un valor");
    } else {
        for (int i = 0; i < param_03.length(); i++) {
            if (!Character.isDigit(param_03.charAt(i))) {
                exception.addError("parametro: No. de Página: Debe ser un valor numerico");
                i = param_03.length();
            }
        }
    }

    // apply validators
    if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
        exception.addError("parametro: Fecha Desanotacion Fin: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_04);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_04 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Desanotacion Fin: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_05) || ("".equalsIgnoreCase(param_05))) {

```

```

        exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_05)) {
            exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
        }
    }

    do_Reporte04_SaveState(request);

    // raise ----

    if (exception.getErrores().size() > 0) {
        throw exception;
    }

    // ----

    String[] params = new String[]{
        REPORTE_XX.PARAM_CMDKEY, REPORTE_04.PARAM_PFECHADESNOTACION,
        REPORTE_04.PARAM_PCIRCULONOMBRE, REPORTE_04.PARAM_PNUMPAGINA,
        REPORTE_04.PARAM_PFECHADESNOTACIONFIN, REPORTE_04.PARAM_USUARIOLOG
    };

    String[] values = new String[]{
        REPORTE_04.PARAM_CMDKEY, param_01, param_02, param_03, param_04, param_05
    };

    String url = makeUrl(params, values);
    reportsServices_ConfigureParameterUri(request, url);
    do_Reporte04_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

private void do_Reporte04_Back_Action(HttpServletRequest request) {
    do_Reporte04_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);

} // :do_Reporte04_Back_Action

private void do_Reporte04_SaveState(HttpServletRequest request) {
    HttpSession session;
    String[] itemIds = new String[]{
        REPORTE_04.PARAM_PFECHADESNOTACION, REPORTE_04.PARAM_PCIRCULONOMBRE,
        REPORTE_04.PARAM_PNUMPAGINA, REPORTE_04.PARAM_PFECHADESNOTACIONFIN,
        REPORTE_04.PARAM_USUARIOLOG
    };

    session = request.getSession();
    save_PagelItemsState(itemIds, request, session);

} // :do_Reporte04_SaveState

private void do_Reporte04_RemoveState(HttpServletRequest request) {

    HttpSession session;
    String[] itemIds = new String[]{
        REPORTE_04.PARAM_PFECHADESNOTACION, REPORTE_04.PARAM_PCIRCULONOMBRE,
        REPORTE_04.PARAM_PNUMPAGINA, REPORTE_04.PARAM_PFECHADESNOTACIONFIN,
        REPORTE_04.PARAM_USUARIOLOG
    };

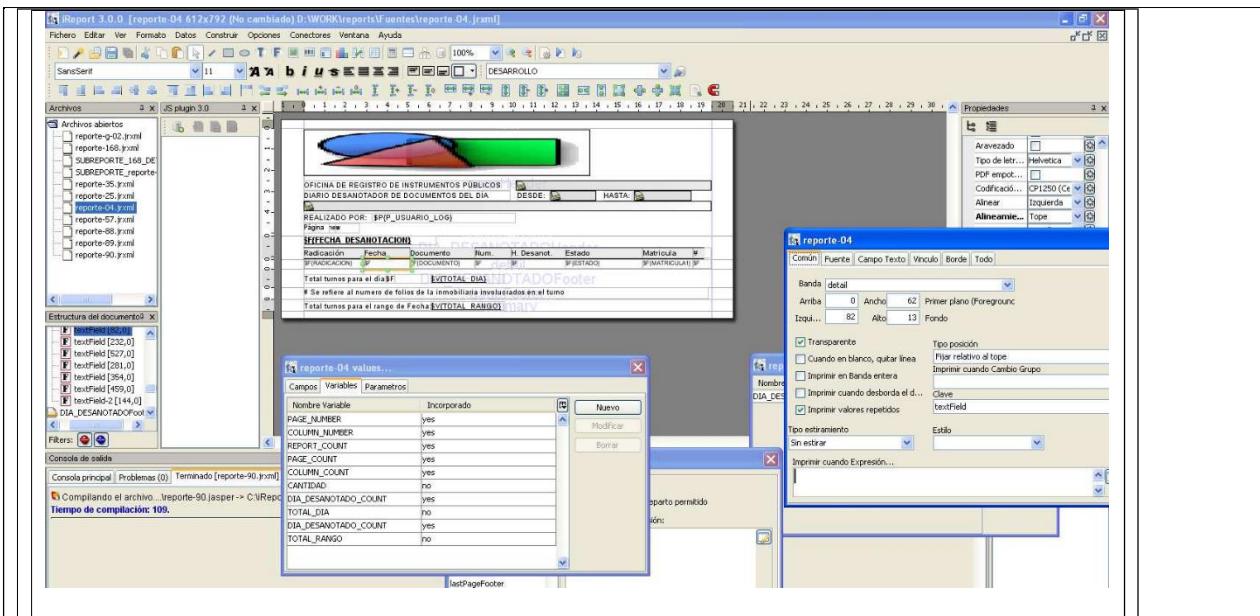
    session = request.getSession();
    delete_PagelItemsState(itemIds, request, session);

} // :do_Reporte04_RemoveState

```

#### JASPER – reporte-04.jasper

Se agrega grupo por Fecha de Desanotación, total turnos por día y un total por el rango.



## Configuración Reporte 57 – Informe de Matrículas Trasladas a este Circulo

### Pantalla Reporte

ANTES

AHORA

SNR :: Superintendencia de Notariado y Registro :: Ministerio del interior y de Justicia :: Rep - Windows Internet Explorer  
[http://192.168.150.42:3000/SNR/reportes.dosACCION=REPORTES\\_57\\_LCADCIRCOLOUSUARIO](http://192.168.150.42:3000/SNR/reportes.dosACCION=REPORTES_57_LCADCIRCOLOUSUARIO)

**Pantalla Administrativa**

Reporte 57

Diario radicador de matriculas

Fecha Apertura Folio Inicio: 12/04/2011  
 Fecha Apertura Folio Fin: 12/04/2011  
 Página: \_\_\_\_\_  
 Círculo Registrat: SELECCIONE UNA OPCIÓN

Regresar Observar

SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES\_REPORT57\_PARAMFORM  
 <-- AHERRENO 11/04/2011  
 Se agrega campo Fecha Apertura folio fin-->

```

<tr>
    <td width="20"></td>
    <td align="left" width="150"><div align="right" >Fecha Apertura Folio Fin:</div></td>
    <td width="200">
    <%
    try {
        if( null == session.getAttribute( AWReportes.REPORTE_57_PARAM_PFECHAFIN ) ) {
            session.setAttribute( AWReportes.REPORTE_57_PARAM_PFECHAFIN, fechaAct );
        }
        TextHelper textHelper = new TextHelper();
        textHelper.setId( AWReportes.REPORTE_57_PARAM_PFECHAFIN );
        textHelper.setNombre( AWReportes.REPORTE_57_PARAM_PFECHAFIN );
        textHelper.setCssClase("camposformtext");
        textHelper.render(request,out);
    }
    catch(HelperException re){
        out.println("ERROR " + re.getMessage());
    }
    %>
    <a href="javascript:NewCal('<%=AWReportes.REPORTE_57__PARAM_PFECHAFIN%>','ddmmmyyyy',true,24)">')"></a>
    </td>
    <td>&nbsp;</td>
</tr>

<!--AHERRENO 11/04/2011
 Se captura el usuario logueado-->
<%
    Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
%>
<input type="hidden" name="<%=>AWReportes.REPORTE_57_PARAM_USUARIOLOG%" id="<%=>AWReportes.REPORTE_57 PARAM_USUARIOLOG%" value="<%=>usuario.getUsername()%>">

```

#### SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java

```

/* Reporte 57 - DIARIO RADICADOR DE MATRICULAS
 * AHERRENO 11/04/2011
 * Se agregan variables: Fecha Final y Usuario que genera el reporte
 */
private void do_Reporte57_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_57_PARAM_PFECHAINI);
    String param_02 = request.getParameter(REPORTE_57_PARAM_PPAGINA);
    String param_03 = request.getParameter(REPORTE_57_PARAM_PCIRCULONOMBRE);
    String param_04 = request.getParameter(REPORTE_57_PARAM_PFECHAFIN);
    String param_05 = request.getParameter(REPORTE_57_PARAM_USUARIOLOG);

    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {

        exception.addError("parametro: Fecha Inicio: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio : La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {

```

```

        exception.addError("parametro: No. de Página: Debe digitar un valor");
    } else {
        for (int i = 0; i < param_02.length(); i++) {
            if (!Character.isDigit(param_02.charAt(i))) {
                exception.addError("parametro: No. de Página: Debe ser un valor numerico");
                break;
            }
        }
    }

    // apply validators
    if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {

        exception.addError("parametro: Circulo: Debe digitar un valor");

    } else {

        if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
            exception.addError("parametro: Circulo: Debe digitar un valor");
        }
    }

    // apply validators
    if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {

        exception.addError("parametro: Fecha Fin: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_04);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_04 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Fin: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_05) || ("".equalsIgnoreCase(param_05))) {
        exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_05)) {
            exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
        }
    }

    String[] itemIds = new String[]{
        REPORTE_57_PARAM_PFECHAINI, REPORTE_57_PARAM_PPAGINA,
        REPORTE_57_PARAM_PCIRCULONOMBRE,
        REPORTE_57_PARAM_PFECHAFIN, REPORTE_57_PARAM_USUARIOLOG
    };

    do_ReporteXX_SaveState(itemIds, request);

    // raise -----
    if (exception.getErrores().size() > 0) {
        throw exception;
    }
    // -----

    String[] params = new String[]{
        REPORTE_XX_PARAM_CMDKEY, REPORTE_57_PARAM_PFECHAINI,
        REPORTE_57_PARAM_PPAGINA, REPORTE_57_PARAM_PCIRCULONOMBRE,
        REPORTE_57_PARAM_PFECHAFIN, REPORTE_57_PARAM_USUARIOLOG
    };

    String[] values = new String[]{
        REPORTE_57_PARAM_CMDKEY, param_01, param_02, param_03, param_04, param_05
    };

```

```

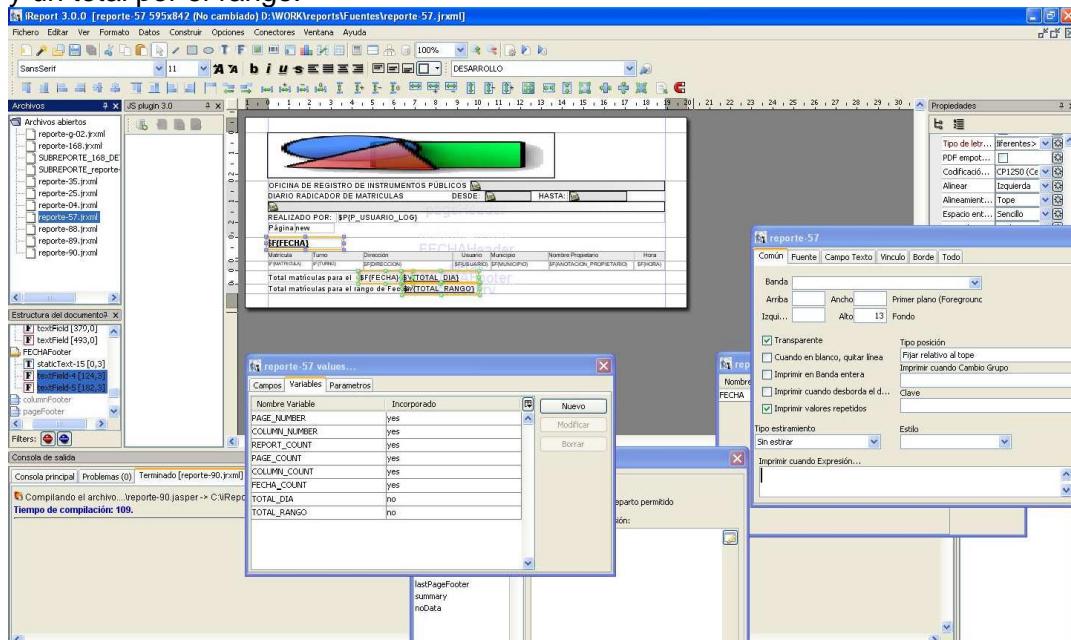
String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_ReporteXX_RemoveState(itemIds, request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

```

### JASPER – reporte-57.jasper

Se agrega grupo por Fecha de radicación de matriculas, total matriculas por día y un total por el rango.



### SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDOResortesJasperDAO.java

```

//Reporte 57
//AHERRENO 11/04/2011
//Se agrega parametro REPORTE_57_PARAM_PFECHAFIN
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_57_PARAM_PFECHAINI);
listaParametros.add(AWReportes.REPORTE_57_PARAM_PFECHAFIN);
listaParametros.add(AWReportes.REPORTE_57_PARAM_PCIRULONOMBRE);
mapaReportes.put(AWReportes.REPORTE_57.PARAM_CMDKEY, new Object[]{
    "SELECT sec_reporte_57.NEXTVAL FROM dual",
    "sp_reporte_57(?, ?, ?, ?)", listaParametros,
    "DELETE FROM sir_op_reporte_57 WHERE sec_proceso = ?");
}

```

### Configuración Reporte 88 – CERTIFICADOS ENTREGADOS EN EL DÍA

Pantalla Reporte

ANTES

**AHORA**

**SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES\_REPORT88\_PARAMFORM**

```

<!--AHERRENO 11/04/2011
SE AGREGA CAMPO FECHA ENTREGA FIN-->
<tr>
<td width="20"></td>
<td align="left" width="150"><div align="right" >Fecha Entrega Fin:</div></td>
<td width="200">
<%
try {
    if( null == session.getAttribute( AWReportes.REPORTE_88_PARAM_PFECHAFIN ) ) {
        session.setAttribute( AWReportes.REPORTE_88_PARAM_PFECHAFIN, fechaAct );
    }

    TextHelper textHelper = new TextHelper();
    textHelper.setId( AWReportes.REPORTE_88_PARAM_PFECHAFIN );
    textHelper.setNombre( AWReportes.REPORTE_88_PARAM_PFECHAFIN );
    textHelper.setCssClase("camposformtext");
    textHelper.render(request,out);
}
catch(HelperException re){
    out.println("ERROR " + re.getMessage());
}
%>

<a href="javascript:NewCal('<%=AWReportes.REPORTE_88__PARAM_PFECHAFIN%' , 'ddmmmyyyy',true,24)">')"/></a>
</td>&nbsp;</td>
</tr>
<!--AHERRENO 11/04/2011
SE CAPTURA EL USUARIO LOGUEADO-->
<%

```

```

        Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
    %>
        <input type="hidden" name="<%=_AWReportes.REPORTE_88_PARAM_USUARIOLOG%">
        id="<%=_AWReportes.REPORTE_88_PARAM_USUARIOLOG%">
        value="<%=_usuario.getUsername()%>">

SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java
/* Reporte 88 - CERTIFICADOS ENTREGADOS EN EL DÍA
 * AHERRENO 11/04/2011
 * Se agregan variables: Fecha Final y Usuario que genera el reporte
 */
private void do_Reporte88_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_88_PARAM_PFECHA);
    String param_02 = request.getParameter(REPORTE_88_PARAM_PCIRCULONOMBRE);
    String param_03 = request.getParameter(REPORTE_88_PARAM_PFECHAFIN);
    String param_04 = request.getParameter(REPORTE_88_PARAM_USUARIOLOG);

    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {

        exception.addError("parametro: Fecha Inicio: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_02)) {
            exception.addError("parametro: Circulo: Debe digitar un valor");
        }
    }

    // apply validators
    if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {

        exception.addError("parametro: Fecha Fin: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_03);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_03 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Fin: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
        exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
            exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
        }
    }

    String[] itemIds = new String[]{

}

```

```

        REPORTE_88_PARAM_PFECHA, REPORTE_88_PARAM_PCIRCULONOMBRE,
        REPORTE_88_PARAM_PFECHAFIN, REPORTE_88_PARAM_USUARIOLOG
    };

    do_ReporteXX_SaveState(itemIds, request);

    // raise -----

    if (exception.getErrores().size() > 0) {
        throw exception;
    }

    // -----


    String[] params = new String[];
        REPORTE_XX_PARAM_CMDKEY, REPORTE_88_PARAM_PFECHA,
        REPORTE_88_PARAM_PCIRCULONOMBRE, REPORTE_88_PARAM_PFECHAFIN,
        REPORTE_88_PARAM_USUARIOLOG
    };

    String[] values = new String[]{
        REPORTE_88_PARAM_CMDKEY, param_01, param_02, param_03, param_04
    };

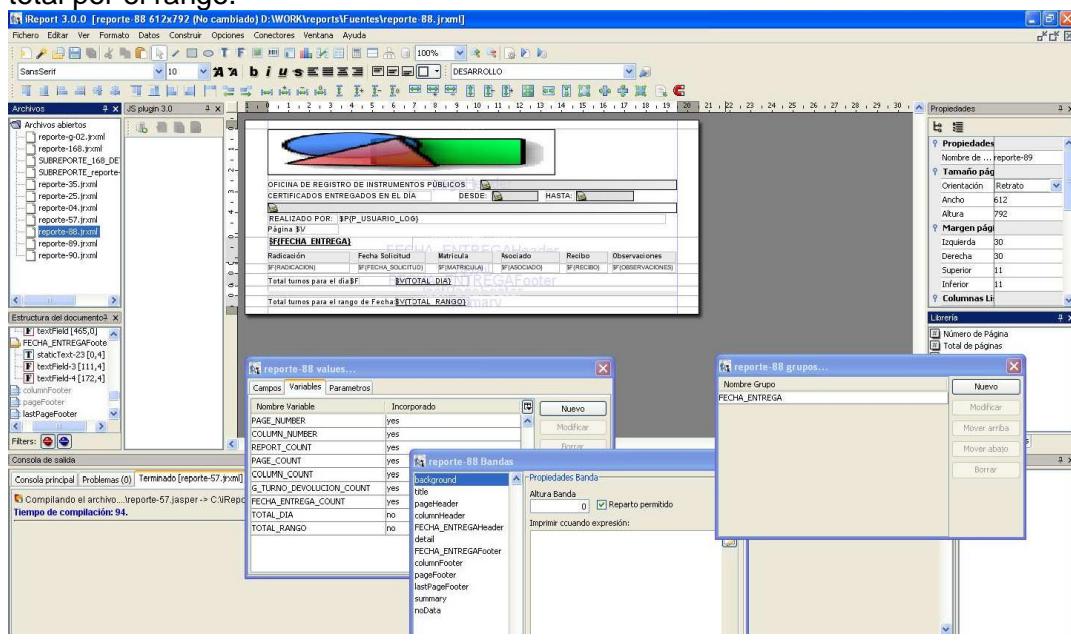
    String url = makeUrl(params, values);
    reportsServices_ConfigureParameterUri(request, url);

    do_ReporteXX_RemoveState(itemIds, request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

```

#### JASPER – reporte-88.jasper

Se agrega grupo por Fecha de Entrega de Certificados, total turnos por día y un total por el rango.



#### Configuración Reporte 89 – LISTADO DE DOCUMENTOS ENTREGADOS EL DÍA

Pantalla Reporte

ANTES

**AHORA**

**SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES\_REPORT89\_PARAMFORM**

```

<!--AHERRENO 11/04/2011
SE AGREGA CAMPO FECHA FIN TURNO-->
<tr>
<td width="20"></td>
<td align="left" width="150"><div align="right" >Fecha Fin Turno:</div></td>
<td width="200">
<%
try {
    if( null == session.getAttribute( AWReportes.REPORTE_89_PARAM_PFECHAFIN ) ) {
        session.setAttribute( AWReportes.REPORTE_89_PARAM_PFECHAFIN, fechaAct );
    }

    TextHelper textHelper = new TextHelper();
    textHelper.setId( AWReportes.REPORTE_89_PARAM_PFECHAFIN );
    textHelper.setNombre( AWReportes.REPORTE_89_PARAM_PFECHAFIN );
    textHelper.setCssClase("camposformtext");
    textHelper.render(request,out);
}
catch(HelperException re){
    out.println("ERROR " + re.getMessage());
}
%>

<a href="javascript:NewCal('<%=AWReportes.REPORTE_89__PARAM_PFECHAFIN%>','ddmmmyyyy',true,24)">')"></a>
</td>&nbsp;</td>
</tr>
<!--AHERRENO 11/04/2011
SE CAPTURA EL USUARIO LOGUEADO-->
<%
    Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
%>
```

```

%>
<input type="hidden" name="<%=_AWReportes.REPORTE_89_PARAM_USUARIOLOG%">
id="<%=_AWReportes.REPORTE_89_PARAM_USUARIOLOG%">
value="<%=_usuario.getUsername()%>">
SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java
/* Reporte 89 - LISTADO DE DOCUMENTOS ENTREGADOS EL DÍA
 * AHERRENO 11/04/2011
 * Se agregan variables: Fecha Final y Usuario que genera el reporte
 */
private void do_Reporte89_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_89_PARAM_PFECHA);
    String param_02 = request.getParameter(REPORTE_89_PARAM_PCIRCULONOMBRE);
    String param_03 = request.getParameter(REPORTE_89_PARAM_PFECHAFIN);
    String param_04 = request.getParameter(REPORTE_89_PARAM_USUARIOLOG);

    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {
        exception.addError("parametro: Fecha Inicio: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_02)) {
            exception.addError("parametro: Circulo: Debe digitar un valor");
        }
    }

    // apply validators
    if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {
        exception.addError("parametro: Fecha Fin: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_03);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_03 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Fin: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
        exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
            exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
        }
    }

    String[] itemIds = new String[]{
        REPORTE_89_PARAM_PFECHA, REPORTE_89_PARAM_PCIRCULONOMBRE,
        REPORTE_89_PARAM_PFECHAFIN, REPORTE_89_PARAM_USUARIOLOG
    };
}

```

```

};

do_ReporteXX_SaveState(itemIds, request);

// raise -----
// -----
if (exception.getErrores().size() > 0) {
    throw exception;
}

// -----
String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY, REPORTE_89_PARAM_PFECHA,
    REPORTE_89_PARAM_PCIRCULONOMBRE, REPORTE_89_PARAM_PFECHAFIN,
    REPORTE_89_PARAM_USUARIOLOG
};

String[] values = new String[]{
    REPORTE_89_PARAM_CMDKEY, param_01, param_02, param_03, param_04
};

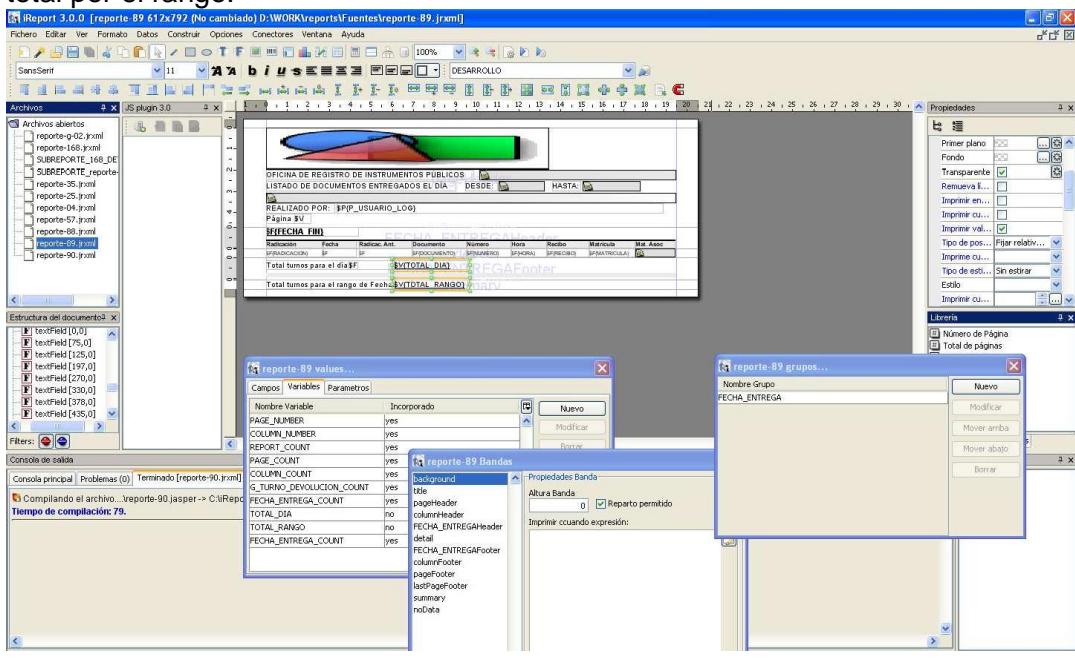
String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_ReporteXX_RemoveState(itemIds, request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

```

### JASPER – reporte-89.jasper

Se agrega grupo por Fecha de Entrega de Certificados, total turnos por día y un total por el rango.



### Configuración Reporte 90 – DIARIO DESANOTADOR DE CERTIFICADOS DEL DIA

Pantalla Reporte

ANTES

**AHORA**

SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES\_REPORT90\_PARAMFORM

```

<!--AHERRENO 11/04/2011
    SE AGREGA CAMPO FECHA DESANOTACION FIN-->
    <tr>
        <td width="20"></td>
        <td align="left" width="150"><div align="right" >Fecha Desanotaci&on Fin:</div></td>
        <td width="200">
        <%
            try {
                if( null == session.getAttribute( AWReportes.REPORTE_90_PARAM_PFECHAFIN ) ) {
                    session.setAttribute( AWReportes.REPORTE_90_PARAM_PFECHAFIN, fechaAct );
                }
                TextHelper textHelper = new TextHelper();
                textHelper.setId( AWReportes.REPORTE_90_PARAM_PFECHAFIN );
                textHelper.setNombre( AWReportes.REPORTE_90_PARAM_PFECHAFIN );
                textHelper.setCssClase("camposformtext");
                textHelper.render(request,out);
            }
            catch(HelperException re){
                out.println("ERROR " + re.getMessage());
            }
        %>
        <a href="javascript:NewCal('<%=AWReportes.REPORTE_90__PARAM_PFECHAFIN%>','ddmmmyyyy',true,24)">')"/></a>
        </td>
        <td>&nbsp;</td>
    </tr>
    <!--AHERRENO 11/04/2011
        SE CAPTURA EL USUARIO LOGUEADO-->
    <%

```

```

        Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
    %>
        <input type="hidden" name="<%=_AWReportes.REPORTE_90_PARAM_USUARIOLOG%">
        id="<%=_AWReportes.REPORTE_90_PARAM_USUARIOLOG%">
        value="<%=_usuario.getUsername()%>">
SIR-APP\src\gov\lsir\core\web\acciones\administracion\AwReportes.java
/* Reporte 90 - DIARIO DESANOTADOR DE CERTIFICADOS DEL DIA
 * AHERRENO 11/04/2011
 * Se agregan variables: Fecha Final y Usuario que genera el reporte
 */
private void do_Reporte90_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_90_PARAM_PFECHA);
    String param_02 = request.getParameter(REPORTE_90_PARAM_PCIRCULONOMBRE);
    String param_03 = request.getParameter(REPORTE_90_PARAM_PNUMPAGINA);
    String param_04 = request.getParameter(REPORTE_90_PARAM_PFECHAFIN);
    String param_05 = request.getParameter(REPORTE_90_PARAM_USUARIOLOG);

    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {

        exception.addError("parametro: Fecha Incio: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_02)) {
            exception.addError("parametro: Circulo: Debe digitar un valor");
        }
    }

    if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {

        exception.addError("parametro: No. de Página: Debe digitar un valor");

    } else {
        for (int i = 0; i < param_03.length(); i++) {
            if (!Character.isDigit(param_03.charAt(i))) {
                exception.addError(
                    "parametro: No. de Página: Debe ser un valor numerico");
                break;
            }
        }
    }

    // apply validators
    if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {

        exception.addError("parametro: Fecha Fin: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_04);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_04 = send1;
        } catch (Exception e) {
    
```

```

        exception.addError("parametro: Fecha Fin: La fecha es inválida" + e.getMessage());
    }

    // apply validators
    if ((null == param_05) || ("".equalsIgnoreCase(param_05))) {
        exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_05)) {
            exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
        }
    }

    String[] itemIds = new String[]{};
    REPORTE_XX_PARAM_CMDKEY, REPORTE_90_PARAM_PFECHA,
    REPORTE_90_PARAM_PCIRCULONOMBRE, REPORTE_90_PARAM_PNUMPAGINA,
    REPORTE_90_PARAM_PFECHAFIN, REPORTE_90_PARAM_USUARIOLOG
};

do_ReporteXX_SaveState(itemIds, request);

// raise -----
if (exception.getErrores().size() > 0) {
    throw exception;
}
// -----
String[] params = new String[]{};
REPORTE_XX_PARAM_CMDKEY, REPORTE_90_PARAM_PFECHA,
REPORTE_90_PARAM_PCIRCULONOMBRE, REPORTE_90_PARAM_PNUMPAGINA,
REPORTE_90_PARAM_PFECHAFIN, REPORTE_90_PARAM_USUARIOLOG
};

String[] values = new String[]{};
REPORTE_90_PARAM_CMDKEY, param_01, param_02, param_03, param_04, param_05
};

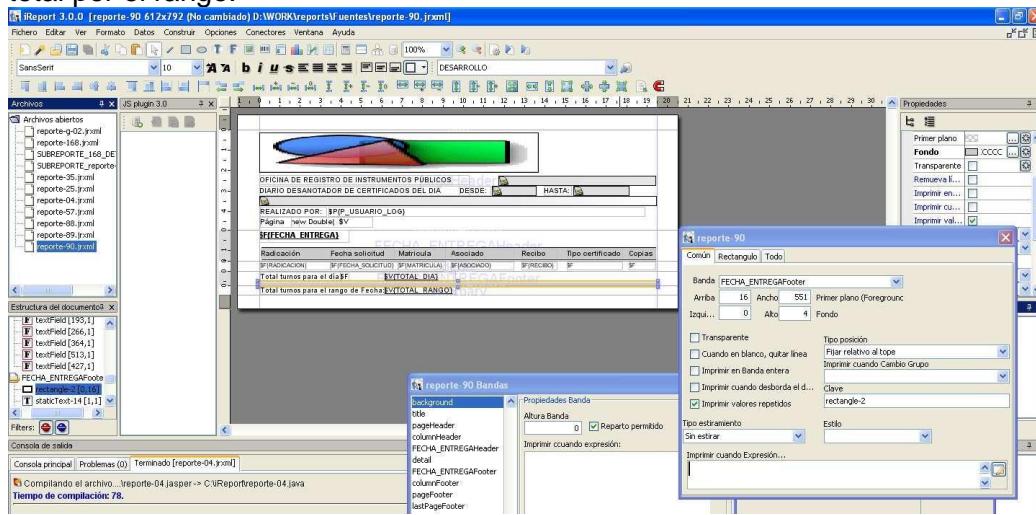
String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_ReporteXX_RemoveState(itemIds, request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

```

#### JASPER – reporte-90.jasper

Se agrega grupo por Fecha de Entrega de Certificados, total turnos por día y un total por el rango.



## 149 Ejecutar corrección Salvedades

### REQUERIMIENTO EJECUTAR CORRECCION –SALVEDADES

Se efectuó prueba en el ambiente de calidad generando los siguientes turnos de corrección asignando los siguientes permisos :

- 2009-368-3-206 – Permiso –Crear anotaciones – corregir especificación
- 2009-368-3-207- Complementación – Tipo Predio
- 2009-368-3-208- Crear anotación (crear cancelación) – anotaciones fecha
- 2009-368-3-209-Dirección- Crear anotación-(segregar)
- 2009-368-3-210-Crear anotaciones (Englobe), complementación
- 2009-368-3-211 Crear anotaciones (copiar anotación)

Se efectuaron cambios en las matrículas inmobiliarias asociadas a los turnos mencionados de acuerdo a los permisos asignados, se observó que para los cambios realizados en todos, el sistema obliga a ingresar salvedad a excepción cuando se agregan anotaciones del proceso de separar y englobar, así mismo para anotaciones de cancelación el sistema no obliga a agregar una salvedad. En el proceso de Englobe y crear anotaciones de cancelación no muestra campo para agregar la salvedad .En el proceso de separar en el último pantallazo muestra campo de salvedad, permitiendo terminar la segregación sin obligar la salvedad, como en el caso que se probó para segregaciones al turno 2009-368-3-

En la matrícula asociada en primer lugar se corrigió el campo de dirección y el sistema obligó a incluir la salvedad, luego se realizó el proceso de segregación en el último pantallazo muestra el campo de la salvedad con la información de la salvedad de la dirección pero no obliga a agregar salvedad de la segregación. Lo correcto es que para la segregación muestre un campo específico para esta salvedad.

#### Observaciones hechas por la SNR al requerimiento el 02/12/2010

En conversación con la funcionaria de la SNR Yamile Moreno

#### Segregación

La salvedad se debe asociar a la anotación del folio matriz(Padre)

#### Englobar

La salvedad se debe asociar a la anotación de los folios matrices (Padres), esta salvedad no debe ir relacionada con los folios hijos.

La salvedad se debe ingresar en la última pantalla campo que debe ser obligatorio.

#### Cancelación

La salvedad se debe asociar a la anotación de cancelación

#### DESARROLLO

ARCHIVO JSP:

WEB-INF.jspSeguro.registro.ENGLOBE\_CONFIRMACION.jsp

Código Insertado Línea 483:

```

<!--
 * @author    : Julio Alcázar Rivas
 * @change    : Se agrega el TextHelper para ingresar la salvedad
 * Caso Mantis : 04131
!!-->
<%
if( local_Solicitud instanceof SolicitudCorreccion ) {
%>
<span class="bgnsb" style="background:<%=request.getContextPath()%>/jsp/images/ind_campotxt.gif"
width="20" height="15"></span>
<table width="100%" class="camposform">
<tr>
    <td width="20"><img src=<%=request.getContextPath()%>/jsp/images/ind_campotxt.gif"
width="20" height="15"></td>
    <td>Salvedad</td>
</tr>
<tr>
    <td width="20">&ampnbsp</td>
    <td>
        <% // id
        try {
            TextHelper local_TextHelper;
            local_TextHelper = new TextHelper();
local_TextHelper.setNombre(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADID);

local_TextHelper.setId(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADID);
            local_TextHelper.setTipo( "hidden" );
            local_TextHelper.setCssClase("camposformtext");
            local_TextHelper.render( request, out );
        }
        catch( HelperException re ) {
            out.println( "ERROR " + re.getMessage() );
        }
        try {
            TextAreaHelper textAreaHelper = new TextAreaHelper();
            textAreaHelper.setFuncion( " onmouseover='this.rows=7' onmouseout='this.rows=2' ");
textAreaHelper.setNombre(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION);
textAreaHelper.setId(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION)
;
            textAreaHelper.setCols( "130" );
            textAreaHelper.setRows( "2" );
            textAreaHelper.setReadOnly(false);
            textAreaHelper.setCssClase("camposformtext");
            textAreaHelper.render(request,out);
        }
        catch( HelperException re ){
            out.println("ERROR " + re.getMessage());
        }
        %>

    </td>
</tr>
</table>
<%
}
%>

```

**WEB-INF.jspSeguro.correccion.  
CORREC\_CREAR\_ANOTACIONCANCELACION.jsp**

**Código Insertado Linea 403:**

```

/*
 * @author    : Julio Alcázar Rivas

```

<pre>* @change : Set de la solitud del turno para verificar en el helper si es de tipo SolicitudCorreccion * Caso Mantis : 02359 */ anotacionesModificacionHelper.setSolicitud(turno.getSolicitud());</pre>	<b>WEB-INF.jspSeguro.registro.TABLA_SEGREGACION.jsp</b>	
<p><b>Código Linea 125:</b></p> <pre>&lt;td&gt;Salvedad Folio Padre&lt;/td&gt;</pre>		
<p><b>Por:</b></p> <pre>&lt;!-- * @author : Julio Alcázar Rivas * @change : se cambia el texto Salvedad Folio Padre por Salvedad Anotacion * Caso Mantis : 04131 !--&gt; &lt;td&gt;Salvedad Anotaci&amp;oacute;n&lt;/td&gt;</pre>		
<p><b>Código Linea 139:</b></p> <pre>local_TextHelper.setNombre( AWModificarFolio.ITEMID_FOLIOEDICION_SALVEDADFOLIOID ); local_TextHelper.setId( AWModificarFolio.ITEMID_FOLIOEDICION_SALVEDADFOLIOID );</pre>		
<p><b>Por:</b></p> <pre>/* * @author : Julio Alcázar Rivas * @change : Se modifica el nombre y id del helper * Caso Mantis : 02359 */ local_TextHelper.setNombre( AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADID ); local_TextHelper.setId( AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADID );</pre>		
<p><b>Código Linea 167:</b></p> <pre>textAreaHelper.setNombre( AWModificarFolio.ITEMID_FOLIOEDICION_SALVEDADFOLIODESCRIPCION ); textAreaHelper.setId( AWModificarFolio.ITEMID_FOLIOEDICION_SALVEDADFOLIODESCRIPCION );</pre>		
<p><b>Por:</b></p> <pre>/* * @author : Julio Alcázar Rivas * @change : Se modifica el nombre y id del helper * Caso Mantis : 02359 */ textAreaHelper.setNombre( AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION ); textAreaHelper.setId( AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION );</pre>		
	<b>CLASE:</b>	
	<b>gov.sir.core.web.helper.registro.AnotacionCancelacionHelper.java</b>	
<p><b>Código Insertado Linea 74:</b></p> <pre>/* * @author : Julio Alcázar Rivas * @change : Se agrega la propiedad solicitud con los metodos get y set * Caso Mantis : 04131 */ private Solicitud solicitud; public Solicitud getSolicitud() {     return solicitud; }</pre>		

```
    public void setSolicitud(Solicitud solicitud) {
        this.solicitud = solicitud;
    }
```

**Método:**

```
public void drawGUI(HttpServletRequest request, JspWriter out)
throws HelperException, IOException {
```

**Código Insertado Linea 552:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Se agrega el campo para incluir en la cancelacion la salvedad
 * Caso Mantis : 04131
 */
if(this.solicitud instanceof SolicitudCorreccion){
    out.write("<table width=\"100%\" border=\"0\" cellpadding=\"0\""
cellspacing="0">");
    out.write("<tr>");
    out.write("<td width=\"12\"><img name=\"sub_r1_c1\" src=\""
request.getContextPath() + "/jsp/images/sub_r1_c1.gif\" width=\"12\" height=\"22\" border=\"0\""
alt=""></td>");
    out.write("<td class=\"bgnsub\">Salvedad: Anotaci&oacute;n</td>");
    out.write("<td width=\"16\" class=\"bgnsub\"><img src=\""
request.getContextPath() + "/jsp/images/ico_anotacion.gif\" width=\"16\" height=\"21\"></td>");
    out.write("<td width=\"15\"><img name=\"sub_r1_c4\" src=\""
request.getContextPath() + "/jsp/images/sub_r1_c4.gif\" width=\"15\" height=\"22\" border=\"0\""
alt=""></td>");
    out.write("</tr>");
    out.write("</table>");
    out.write("<table width=\"100%\" class=\"camposform\">");
    out.write("<tr>");
    out.write("<td width=\"20\"><img src=\""
request.getContextPath() + "/jsp/images/ind_campotxt.gif\" width=\"20\" height=\"15\"></td>");
    out.write("<td>Salvedad</td>");
    out.write("</tr>");
    out.write("<tr>");
    out.write("<td>&ampnbsp</td>");
    out.write("<td>");
    textAreaHelper.setNombre(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION);
    textAreaHelper.setFuncion( " onmouseover='this.rows=7'
onmouseout='this.rows=2' ");
    textAreaHelper.setCols("130");
    textAreaHelper.setRows("2");
    textAreaHelper.setReadOnly(false);
    textAreaHelper.setCssClase("camposformtext");
    textAreaHelper.setId(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION)
;
    textAreaHelper.render(request,out);
    out.write("</td>");
    out.write("</tr>");
    out.write("</table>");
    out.write("<br>");
}
```

**CLASE:**

```
gov.sir.core.web.acciones.correcion.AWModificarFolio.java
```

**Método:**

```
private Evento cancelarAnotacion(HttpServletRequest request) throws
AccionWebException{
```

**Código Insertado Linea 3486:**

```
/*
```

```

* @author    : Julio Alcázar Rivas
* @change    : Se realiza validaciones de la información de la salvedad
* Caso Mantis : 04131
*/
if(turno.getSolicitud() instanceof SolicitudCorreccion){
    String param_AnotacionSalvedad_Descripcion =
request.getParameter(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION );
    String param_AnotacionSalvedad_Id      =
request.getParameter(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADID);
    BasicConditionalValidator stage1_validator;
    stage1_validator = new BasicStringNotNullOrEmptyValidatorWrapper();
    stage1_validator.setObjectToValidate( param_AnotacionSalvedad_Descripcion );
    stage1_validator.validate();
    if( stage1_validator.getResult() != true ) {
        exception.addError( "anotacion:salvedad; debe escribir un valor en la salvedad" );
    }
    // 1:B maxima longitud
    stage1_validator = new BasicStringMaxLengthValidatorWrapper( 255 );
    stage1_validator.setObjectToValidate( param_AnotacionSalvedad_Descripcion );
    stage1_validator.validate();

    if( stage1_validator.getResult() != true ) {
        exception.addError( "anotacion:salvedad; la salvedad debe tener menos de 255 caracteres" );
    }
    // 1:C minima longitud
    stage1_validator = new BasicStringMinLengthValidatorWrapper( 30 );
    stage1_validator.setObjectToValidate( param_AnotacionSalvedad_Descripcion );
    stage1_validator.validate();
    if( stage1_validator.getResult() != true ) {
        exception.addError( "anotacion:salvedad; la salvedad debe tener mas de 30 caracteres" );
    }
    String param_AnotacionSalvedad_NumRadicacion;
    param_AnotacionSalvedad_NumRadicacion = turno.getIdWorkflow();
    if( null != param_AnotacionSalvedad_Id )
        &&( "".equals( param_AnotacionSalvedad_Id.trim() ) ) {
            param_AnotacionSalvedad_Id = null;
        }
    SalvedadAnotacion local_SalvedadAnotacion = new SalvedadAnotacion();
    local_SalvedadAnotacion.setDescripcion( param_AnotacionSalvedad_Descripcion );
    local_SalvedadAnotacion.setIdSalvedad( param_AnotacionSalvedad_Id );
    local_SalvedadAnotacion.setIdMatricula( folio.getIdMatricula() );
    local_SalvedadAnotacion.setIdAnotacion( String.valueOf(folio.getLastIdAnotacion())+1 );
    local_SalvedadAnotacion.setNumRadicacion( param_AnotacionSalvedad_NumRadicacion );
    List local_SalvedadList = new ArrayList();
    local_SalvedadList.add( local_SalvedadAnotacion );
    anotacion.setSalvedades( local_SalvedadList );
}

```

#### Método:

**private void preservarInfoBasicaAnotacion(HttpServletRequest request) {**

**Código Insertado Linea 12502:**

```

/*
 * @author    : Julio Alcázar Rivas
 * @change    : Se preserva la información de la salvedad de cancelacion en correcciones
 * Caso Mantis : 04131
*/
if(request.getParameter(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION)!=null){
    session.setAttribute(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION,

```

```
request.getParameter(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION ))};
```

**Método:**

```
public void doEnd(HttpServletRequest request, EventoRespuesta evento)  
{
```

**Código Insertado Linea 13579:**

```
/*  
 * @author : Julio Alcázar Rivas  
 * @change : Eliminar el atributo de salvedad de la session  
 * Caso Mantis : 04131  
 */  
session.removeAttribute(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION);
```

**CLASE:**

**gov.sir.core.eventos.registro.EvnEnglobe.java**

**Código Insertado Linea 83:**

```
/*  
 * @author : Julio Alcázar Rivas  
 * @change : Se agregan las propiedades salvedadDescripcion y salvedadId y sus metodos  
get y set  
 *          para manejar la salvedad  
 * Caso Mantis : 04131  
 */  
private String salvedadDescripcion;  
  
private String salvedadId;  
  
public String getSalvedadDescripcion() {  
    return salvedadDescripcion;  
}  
  
public void setSalvedadDescripcion(String salvedadDescripcion) {  
    this.salvedadDescripcion = salvedadDescripcion;  
}  
  
public String getSalvedadId() {  
    return salvedadId;  
}  
  
public void setSalvedadId(String salvedadId) {  
    this.salvedadId = salvedadId;  
}
```

**CLASE:**

**gov.sir.core.web.acciones.registro.AWEnglobe.java**

**Método:**

```
private Evento englobarFolios(HttpServletRequest request) throws  
AccionWebException{
```

```
/*
```

```
 * @author : Julio Alcázar Rivas  
 * @change : Se modifica el metodo para que lance excepciones de tipo AccionWebException  
 * Caso Mantis : 04131  
 */
```

```
private Evento englobarFolios(HttpServletRequest request) throws AccionWebException{
```

**Código Insertado Linea 2294:**

```
/*
```

```
 * @author : Julio Alcázar Rivas  
 * @change : Se realiza validaciones de la informacion de la salvedad y se cambio de
```

```

lugar
    *
    *           la variable turno
    * Caso Mantis : 04131
    */
    Turno turno = (Turno) session.getAttribute(WebKeys.TURNO);
    String param_AnotacionSalvedad_Descripcion = null;
    String param_AnotacionSalvedad_Id = null;
    if(turno.getSolicitud() instanceof SolicitudCorreccion){
        WizardEnglobeException exception = new WizardEnglobeException();
        param_AnotacionSalvedad_Descripcion =
request.getParameter(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION );
        param_AnotacionSalvedad_Id      =
request.getParameter(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADID);
        BasicConditionalValidator stage1_validator;
        stage1_validator = new BasicStringNotNullOrEmptyValidatorWrapper();
        stage1_validator.setObjectToValidate( param_AnotacionSalvedad_Descripcion );
        stage1_validator.validate();
        if( stage1_validator.getResult() != true ) {
            exception.addError( "anotacion:salvedad; debe escribir un valor en la salvedad" );
        }
        // 1:B maxima longitud
        stage1_validator = new BasicStringMaxLengthValidatorWrapper( 255 );
        stage1_validator.setObjectToValidate( param_AnotacionSalvedad_Descripcion );
        stage1_validator.validate();
        if( stage1_validator.getResult() != true ) {
            exception.addError( "anotacion:salvedad; la salvedad debe tener menos de 255
caracteres" );
        }
        // 1:C minima longitud
        stage1_validator = new BasicStringMinLengthValidatorWrapper( 30 );
        stage1_validator.setObjectToValidate( param_AnotacionSalvedad_Descripcion );
        stage1_validator.validate();
        if( stage1_validator.getResult() != true ) {
            exception.addError( "anotacion:salvedad; la salvedad debe tener mas de 30
caracteres" );
        }
        if( ( null != param_AnotacionSalvedad_Id )
            &&( "".equals( param_AnotacionSalvedad_Id.trim() ) ) ) {
            param_AnotacionSalvedad_Id = null;
        }
        if( exception.getErrores().size() > 0 ) {
            throw exception;
        }
    }
}

```

#### **código Insertado Linea 2406:**

```

/*
    * @author    : Julio Alcázar Rivas
    * @change    : Se agrega al evento evnEnglobe informacion de la salvedad para turno
tipo SolicitudCorreccion
    * Caso Mantis : 04131
    */
    if(turno.getSolicitud() instanceof SolicitudCorreccion){
        evnEnglobe.setSalvedadDescripcion(param_AnotacionSalvedad_Descripcion);
        evnEnglobe.setSalvedadId(param_AnotacionSalvedad_Id);
    }

```

#### **Método:**

```

public void doEnd(HttpServletRequest request, EventoRespuesta evento)
{

```

#### **Código Insertado Linea 2840:**

```

/*
 * @author    : Julio Alcázar Rivas
 * @change    : Eliminar los atributos de salvedad de la session
 * Caso Mantis : 04131
 */
session.removeAttribute(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION);
session.removeAttribute(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADID);

```

**CLASE:**

**gov.sir.core.negocios.acciones.registro.ANEnglobe.java**

**Método:**

**private EventoRespuesta englobarFolios(Evento ev) throws  
EventoException {**

**Código Insertado Linea 454:**

```

/*
 * @author    : Julio Alcázar Rivas
 * @change    : Se agrega la informacion de la salvedad a la anotacion
 * Caso Mantis : 04131
 */
SalvedadAnotacion local_SalvedadAnotacion = new SalvedadAnotacion();
String salvedadDescripcion = evento.getSalvedadDescripcion();
String salvedadId = evento.getSalvedadId();
local_SalvedadAnotacion.setDescripcion(salvedadDescripcion);
local_SalvedadAnotacion.setIdSalvedad(salvedadId);
local_SalvedadAnotacion.setIdMatricula(webEnglobe.getIdMatriculaUbicacion());
Folio folio = null;
try {
    folio = forseti.getFolioByID(webEnglobe.getIdMatriculaUbicacion());
    local_SalvedadAnotacion.setIdAnotacion(
String.valueOf(folio.getLastIdAnotacion())+1 );
} catch (Throwable ex) {
    Logger.getLogger(ANEnglobe.class.getName()).log(Level.SEVERE, null, ex);
}
local_SalvedadAnotacion.setNumRadicacion(turno.getIdWorkflow());

List local_SalvedadList = new ArrayList();
local_SalvedadList.add( local_SalvedadAnotacion );
anotacion.setSalvedades( local_SalvedadList );

```

**CLASE:**

**gov.sir.core.web.acciones.registro.AWSegregacion.java**

**Método:**

**private Evento segregarMasivo(HttpServletRequest request) throws  
ValidacionParametrosSegregacion {**

```

/**
 * @param request
 * @return
 * @author    : Julio Alcázar Rivas
 * @change    : Se cambiaron las constantes de navegacion web y se modifio el
comportamiento de las salvedades
 *             cambiaron de tipo SalvedadFolio a SalvedadAnotacion
 * Caso Mantis : 04131
 */

```

```

private Evento segregarMasivo(HttpServletRequest request) throws
ValidacionParametrosSegregacion {
    HttpSession session = request.getSession();
    ValidacionParametrosSegregacion exception = new
ValidacionParametrosSegregacion();
    WebSegregacion webSegregacion = (WebSegregacion)
request.getSession().getAttribute(WebKeys.WEB_SEGREGACION);

```

```

        if(webSegregacion==null){
            webSegregacion = new WebSegregacion();
        }
        List foliosD = webSegregacion.getFoliosDerivados();
        if (foliosD == null || foliosD.isEmpty()) {
            exception.addError("Debe incluir por lo menos la informacion de un folio
derivado");
        }
        Fase fase = (Fase)session.getAttribute(WebKeys.FASE) ;
        String salvedadDesc =
request.getParameter(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION
);
        if(!fase.getID().equals(CFase.CAL_CALIFICACION)){

            session.setAttribute(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCR
IPCION,salvedadDesc);

            if (salvedadDesc == null || salvedadDesc.equals(""))
            {
                exception.addError("Debe ingresar una salvedad");
            } else if (salvedadDesc.length() < 30)
            {
                exception.addError("La salvedad debe tener más de 30 caracteres");
            } else if (salvedadDesc.length() > 255)
            {
                exception.addError("La salvedad debe tener máximo 255
caracteres");
            }
        }
        if (!exception.getErrores().isEmpty())
        {
            throw exception;
        }
        Turno turno = (Turno) session.getAttribute(WebKeys.TURNO);
        Folio folio = (Folio) session.getAttribute(WebKeys.FOLIO);
        String idSalvedad =
(String)session.getAttribute(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADID);
        SalvedadAnotacion salvedadAnotacion = null;
        if(!fase.getID().equals(CFase.CAL_CALIFICACION)){
            salvedadAnotacion = new SalvedadAnotacion();
            salvedadAnotacion.setDescripcion(salvedadDesc);
            salvedadAnotacion.setIdSalvedad(idSalvedad);
            salvedadAnotacion.setIdMatricula(folio.getIdMatricula());
            salvedadAnotacion.setNumRadicacion(turno.getIdWorkflow());
        }
        Circulo circulo = (Circulo) session.getAttribute(WebKeys.CIRCULO);
        org.auriga.core.modelo.transferObjects.Usuario usuarioAuriga =
(org.auriga.core.modelo.transferObjects.Usuario)
request.getSession().getAttribute(SMARTKeys.USUARIO_EN_SESION);
        gov.sir.core.negocio.modelo.Usuario usuarioSIR =
(gov.sir.core.negocio.modelo.Usuario) session.getAttribute(WebKeys.USUARIO);

        EvnSegregacion ev = new EvnSegregacion(usuarioAuriga, folio, webSegregacion ,
turno, circulo, EvnSegregacion.SEGREGACION_FOLIO, usuarioSIR);
        if(!fase.getID().equals(CFase.CAL_CALIFICACION)){
            ev.setSalvedadanotacion(salvedadAnotacion);
        }
        return ev;
    }

```

**Método:**

**public void doEnd(HttpServletRequest request, EventoRespuesta evento)**

<pre>{ </pre> <p><b>Código Insertado Linea 3047:</b></p> <pre>/*  * @author    : Julio Alcázar Rivas  * @change    : Eliminar los atributos de salvedad de la session  * Caso Mantis : 04131  */ session.removeAttribute(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADDESCRIPCION); session.removeAttribute(AWModificarFolio.ITEMID_FOLIOEDICION_ANOTACIONXSALVEDADID);</pre> <p style="text-align: center;"><b>CLASE:</b></p> <p style="text-align: center;"><b>gov.sir.core.eventos.registro.EvnSegregacion.java</b></p>
<p><b>Código Insertado Linea 86:</b></p> <pre>/*  * @author    : Julio Alcázar Rivas  * @change    : Se agrega la propiedad SalvedadAnotacion salvedadanotacion y sus metodos get y set  *  *          para manejar la salvedad  * Caso Mantis : 04131  */ private SalvedadAnotacion salvedadanotacion;  public SalvedadAnotacion getSalvedadanotacion() {     return salvedadanotacion; }  public void setSalvedadanotacion(SalvedadAnotacion salvedadanotacion) {     this.salvedadanotacion = salvedadanotacion; }</pre> <p style="text-align: center;"><b>CLASE:</b></p> <p style="text-align: center;"><b>gov.sir.core.negocio.acciones.registro.ANSegregacion.java</b></p> <p style="text-align: center;"><b>Método:</b></p> <p style="text-align: center;"><b>private EventoRespuesta segregacionMasiva(Evento ev) throws EventoException {</b></p>
<p><b>Código Linea 313:</b></p> <pre>foliosDerivados = forseti.segregarFolioPLSQL(folioOriginal, idTurno);</pre> <p><b>Por:</b></p> <pre>/*  * @author    : Julio Alcázar Rivas  * @change    : se recupera la salvedad del evento getSalvedadanotacion() y se incluye en el metodo  *  *          segregarFolioPLSQL para realizar la segregacion  * Caso Mantis : 04131  */ SalvedadAnotacion salvedad = evento.getSalvedadanotacion(); foliosDerivados = forseti.segregarFolioPLSQL(folioOriginal, idTurno, salvedad);</pre> <p style="text-align: center;"><b>CLASE:</b></p> <p style="text-align: center;"><b>gov.sir.forseti.dao.impl.jdogenie.JDOGenieFolioDAO.java</b></p> <p style="text-align: center;"><b>Método:</b></p> <p style="text-align: center;"><b>public List segregarFolioPLSQL(Folio datos, TurnoPk oid,SalvedadAnotacion salvedad) throws DAOException{</b></p>
<p><b>Código Linea 17875:</b></p> <pre>public List segregarFolioPLSQL(Folio datos, TurnoPk oid) throws DAOException{</pre> <p><b>Por:</b></p> <pre>/**  * Segrega un folio a través de un procedimiento almacenado en PL/SQL</pre>

```

 * @param datos Datos del folio a segregar
 * @param oid ID el turno a segregar
 * @author : Julio Alcázar Rivas
 * @change : se agrega nuevo param al metodo SalvedadAnotacion salvedad
 * Caso Mantis : 04131
 * @return List con los folios segregados
 * @throws Throwable
 */
public List segregarFolioPLSQL(Folio datos, TurnoPk oid,SalvedadAnotacion salvedad) throws
DAOException{

```

### Código Línea 17913:

```

List salvedades = datos.getSalvedades();
if (salvedades != null && salvedades.size()>0)
{
    //Obtenemos el folio persistente
    FolioEnhancedPk fid = new FolioEnhancedPk();
    fid.idMatricula = datos.getIdMatricula();

    FolioEnhanced folio = this.getFolioByID(fid, pm);

    Usuario usu = this.getUsuarioBloqueoFolio(new FolioPk(datos.getIdMatricula()));
    UsuarioEnhanced us =
        this.getUsuarioByID(
            new UsuarioEnhancedPk(new
Long(usu.getIdUsuario()).toString()),
            pm);

    for (Iterator salvItera = salvedades.iterator();
         salvItera.hasNext();)
    {
        SalvedadFolio salv = (SalvedadFolio)salvItera.next();
        SalvedadFolioTMP salvTmp = new SalvedadFolioTMP();
        salvTmp.setIdSalvedadFoTmp(salv.getIdSalvedad());
        salvTmp.setDescripcion(salv.getDescripcion());
        salvTmp.setIdMatricula(salv.getIdMatricula());
        salvTmp.setNumRadicacion(salv.getNumRadicacion());
        if (salvTmp.getIdSalvedadFoTmp() == null
            || salvTmp.getIdSalvedadFoTmp().equals(""))
        {
            this.addSalvedadTMPToFolio(folio,salvTmp,us,pm);
            pm.makePersistent(salvTmp);
        } else
        {
            this.updateSalvedadTMPToFolio(folio,salvTmp,us,pm);
        }
    }
}

```

### Por:

```

/*
 * @author : Julio Alcázar Rivas
 * @change : 1) Se obtiene la anotacion temporal AnotacionTMPPk anotacionPk para agregarla la
salvedad
 *           2) Se modifica el tipo de salvedad de tipo SalvedadFolio por SalvedadAnotacionTMP
 * Caso Mantis : 04131
*/
AnotacionTMPPk anotacionPk = new AnotacionTMPPk();
anotacionPk.idAnotacionTmp = Long.toString(datos.getLastIdAnotacion()+1);
anotacionPk.idMatricula = datos.getIdMatricula();
AnotacionTMP anotacion = this.getAnotacionTMPByID(anotacionPk,pm);
if (salvedad != null )

```

```

{
    Usuario usu = this.getUsuarioBloqueoFolio(new FolioPk(datos.getIdMatricula()));
    UsuarioEnhanced us = this.getUsuarioByID(new UsuarioEnhancedPk(new
Long(usu.getIdUsuario()).toString()),pm);
    SalvedadAnotacionTMP salvTmp = new SalvedadAnotacionTMP();
    salvTmp.setDescripcion(salvedad.getDescripcion());
    salvTmp.setIdMatricula(salvedad.getIdMatricula());
    salvTmp.setNumRadicacion(salvedad.getNumRadicacion());
    this.addSalvedadToAnotacionTMP(anotacion,salvTmp,us,pm);
    pm.makePersistent(salvTmp);
}

```

**Método:**

```

public Folio englobarFolio(List foliosFuentes, Folio folioBase, Usuario
    usuario, List tabla, Anotacion anotacion, FolioDerivado
    informacionAdicional, boolean copiarComentarioEnglobe, boolean
    copiarComentarioHeredadas , TurnoPk oid) throws DAOException {

```

**Código Insertado Línea 19049:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : se eliminan las salvedades de la anotacion del nuevo folio
 *            estas solo deben aparecer en los folios padres
 * Caso Mantis : 04131
 */
anTemp2.removeAllSalvedadesTMP();

```

**CLASE:**

**gov.sir.forseti.ForsetiService.java**

**Código Línea 2481:**

```

public List segregarFolioPLSQL(Folio datos, TurnoPk oid) throws ForsetiException {
    try {
        return forseti.getFolioDAO().segregarFolioPLSQL(datos, oid);
    }

```

**Por:**

```

* @author : Julio Alcázar Rivas
* @change : Se modificó el método segregarFolioPLSQL agregándole el parámetro SalvedadAnotacion
salvedad
* Caso Mantis : 04131
* @throws Throwable
*/

```

```

public List segregarFolioPLSQL(Folio datos, TurnoPk oid, SalvedadAnotacion salvedad) throws
ForsetiException {
    try {
        return forseti.getFolioDAO().segregarFolioPLSQL(datos, oid, salvedad);
    }

```

**CLASE:**

**gov.sir.forseti.dao.FolioDAO.java**

**Código Línea 1042:**

```

public List segregarFolioPLSQL(Folio datos, TurnoPk oid) throws DAOException;

```

**Por:**

```

* @author : Julio Alcázar Rivas
* @change : Se modificó el método segregarFolioPLSQL agregándole el parámetro SalvedadAnotacion
salvedad
* Caso Mantis : 04131
*/

```

```

public List segregarFolioPLSQL(Folio datos, TurnoPk oid, SalvedadAnotacion salvedad) throws
DAOException;

```

**CLASE:**

**gov.sir.forseti.ejb.ForsetiServiceEJBBean.java**

**Código Línea 741:**

```

public List segregarFolioPLSQL(Folio datos, TurnoPk oid) throws ForsetiException {
    return forseti.segregarFolioPLSQL(datos, oid);
}

Por:
* @author : Julio Alcázar Rivas
* @change : Se modifco el metodo segregarFolioPLSQL agregandole el param SalvedadAnotacion
salvedad
* Caso Mantis : 04131
*/
public List segregarFolioPLSQL(Folio datos, TurnoPk oid,SalvedadAnotacion salvedad) throws
ForsetiException {
    return forseti.segregarFolioPLSQL(datos, oid, salvedad);

```

**CLASE:**

**gov.sir.forseti.interfaz.ForsetiServiceInterface.java**

**Código Línea 411:**

```
public List segregarFolioPLSQL(Folio datos, TurnoPk oid) throws Throwable;
```

**Por:**

```

* @author : Julio Alcázar Rivas
* @change : Se modifco el metodo segregarFolioPLSQL agregandole el param SalvedadAnotacion
salvedad
* Caso Mantis : 04131
*/
public List segregarFolioPLSQL(Folio datos, TurnoPk oid, SalvedadAnotacion salvedad) throws
Throwable;

```

**158 Reporte 25  
Informativo**

**REQUERIMIENTO**

Reporte 25 Tiracaja.

Cuando se paga más de un turno con el mismo documento ya sea consignación o cheque presenta valores erróneos al realizar el cálculo de la diferencia. En el reporte se aplica la siguiente fórmula para calcular la diferencia valor del documento – valor aplicado; a demás en la columna valor documento se repite el valor del documento las veces que se pagó con dicho documento Ejm:

Turno	VL APLICADO	VL DOCUMENTO	Diferencia	Núm. Documento
1	50.000	150.000	100.000	63532
2	50.000	150.000	100.000	63532
3	50.000	150.000	<u>100.000</u>	63532
				300.000 (Erróneo debe dar una diferencia de 0)

Sugerimos implementar:

Turno	VL APLICADO	VL DOCUMENTO	Diferencia	Núm. Documento
1	50.000	150.000	100.000	63532
2	50.000	150.000	-50.000	63532
3	50.000	150.000	-50.000	63532
				0

La diferencia se tomara: Si el saldo es diferente de cero (0) tomara el valor de la liquidación multiplicado \*-1, de lo contrario tomara el valor del documento menos el valor aplicado.  
Información Adicional

Ver Glpi 16413

## DESARROLLO

Script Bases de Datos: Procedimiento

SP\_REPORTE\_25

SCRIPT\_MANTIS\_7668.sql

Creación De Clases:

NINGUNA (el mismo formato de modificación clases o jsp)

Se modificaron los siguientes archivos:

reporte-25.jrxml.

Adjunto código fuente.

Se cambió el reporte REPORTE\_25, ya que anteriormente se hacía directamente la consulta desde el Jasper. Se implementó un procedimiento almacenado (SP\_REPORTE\_25) para generar la información en una tabla temporal y posteriormente realizar la consulta desde el Jasper.

Configuración Reporte 25 – Informe recaudos del día por usuario
SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES_REPORT25_PARAMFORM
<% Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO); %> <input <br="" name="&lt;%=AWReportes.REPORTE_25_PARAM_USUARIOLOG%&gt;" type="hidden"/> id="<%=AWReportes.REPORTE_25_PARAM_USUARIOLOG%>" value="<%=usuario.getUsername()%>">
SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java
/** * @param request * Reporte 25 - INFORME RECAUDOS DEL DIA POR USUARIO * AHERRENO 11/04/2011 * Se agrega Variable REPORTE_25 PARAM_USUARIOLOG */ private void do_Reporte25_Send_Action(HttpServletRequest request) throws ValidacionParametrosException {  ValidacionParametrosException exception = new ValidacionParametrosException();  String param_01 = request.getParameter(REPORTE_25_PARAM_PAGOFECHA); String param_02 = request.getParameter(REPORTE_25_PARAM_CIRCULONOMBRE); String param_03 = request.getParameter(REPORTE_25_PARAM_USUARIONOMBRE); String param_04 = request.getParameter(REPORTE_25_PARAM_USUARIOLOG);  // apply validators if ((null == param_01)    ("".equalsIgnoreCase(param_01))) { exception.addError("parametro: Fecha: Debe digitar un valor"); } else { String send1 = null;  Date fecha = null; try { fecha = DateFormatUtil.parse(param_01); send1 = DateFormatUtil.format("yyyy-MM-dd", fecha); } catch (Exception e) { exception.addError("Error al formatear la fecha: " + e.getMessage()); } } }

```

        param_01 = send1;
    } catch (Exception e) {
        exception.addError("parametro: Fecha: La fecha es inválida" + e.getMessage());
    }
}

// apply validators
if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {

    exception.addError("parametro: Circulo: Debe digitar un valor");

} else {

    if (WebKeys.SIN_SELECCIONAR.equals(param_02)) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    }
}

// apply validators
if ((null == param_03) || ("".equalsIgnoreCase(param_03) || (WebKeys.SIN_SELECCIONAR.equals(param_03)))) {
    exception.addError("parametro: Usuario: Debe seleccionar un valor");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
        exception.addError("parametro: Usuario: Debe digitar un valor");
    }
}

// apply validators
if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
    exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
        exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
    }
}

do_Reporte25_SaveState(request);

if (exception.getErrores().size() > 0) {
    throw exception;
}

String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY,
    REPORTE_25_PARAM_PAGOFECHA,
    REPORTE_25_PARAM_CIRCULONOMBRE,
    REPORTE_25_PARAM_USUARIONOMBRE,
    REPORTE_25_PARAM_USUARIOLOG
};

String[] values = new String[]{
    REPORTE_25_PARAM_CMDKEY,
    param_01,
    param_02,
    param_03,
    param_04
};

String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_Reporte25_RemoveState(request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);

}

private void do_Reporte25_Back_Action(HttpServletRequest request) {

    do_Reporte25_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);

} // :do_Reporte25_Back_Action

```

```

private void do_Reporte25_SaveState(HttpServletRequest request) {
    HttpSession session;
    String[] itemIds = new String[]{
        REPORTE_25_PARAM_PAGOFECHA,
        REPORTE_25_PARAM_CIRCULONOMBRE,
        REPORTE_25_PARAM_USUARIONOMBRE,
        REPORTE_25_PARAM_USUARIOLOG
    };
    session = request.getSession();
    save_PageltemsState(itemIds, request, session);
} // :do_Reporte25_SaveState

private void do_Reporte25_RemoveState(HttpServletRequest request) {
    HttpSession session;
    String[] itemIds = new String[]{
        REPORTE_25_PARAM_PAGOFECHA,
        REPORTE_25_PARAM_CIRCULONOMBRE,
        REPORTE_25_PARAM_USUARIONOMBRE,
        REPORTE_25_PARAM_USUARIOLOG
    };
    session = request.getSession();
    session.removeAttribute(AWReportes.USUARIOS_CONSULTAR_POR_CIRCULO);
    session.removeAttribute("consultaUsuariosCajeros");
    delete_PageltemsState(itemIds, request, session);
} // :do_Reporte25_RemoveState

```

#### SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDORreportesJasperDAO.java

```

//AHERRERO 11/04/2011
//Reporte 25
//Se obtienen los parámetros para enviar al procedimiento
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_25_PARAM_PAGOFECHA);
listaParametros.add(AWReportes.REPORTE_25_PARAM_CIRCULONOMBRE);
listaParametros.add(AWReportes.REPORTE_25_PARAM_USUARIONOMBRE);
mapaReportes.put(AWReportes.REPORTE_25_PARAM_CMDKEY, new Object[]{
    "SELECT seq_reporte_25.NEXTVAL FROM dual",
    "sp_reporte_25(?, ?, ?, ?)", listaParametros,
    "DELETE FROM sir_op_reporte_25 WHERE sec_proceso = "});

```

### 173 Reporte 94 Informativo

#### **REQUERIMIENTO**

Para el reporte 94 al seleccionar opción todos no muestran ningún registro, Solo se muestran datos para los siguientes funcionarios en la fecha 03/02/2010 en la ORIP IBAGUE.

Ángela María Ospina Bonilla  
María Bertilda Palacios

#### **DESARROLLO**

EL "REPORTE 94" SE ENCUENTRA ASIGNADO A LOS ROLES  
**SIR\_ROL\_ADMINISTRADOR\_NACIONAL**  
**SIR\_ROL\_ADMINISTRADOR\_REGIONAL**  
**SIR\_ROL\_CONSULTA\_REPORTES\_NACIONAL**  
**SIR\_ROL\_USUARIO\_ADMINISTRATIVO**

## Se modifica procedimiento SP\_REPORTE\_94

/\* AHERRENO 02/03/2011  
REQ 173 se modifica condición, teniendo en cuenta que el usuario puede seleccionar opción TODOS en la aplicación\*/  
AND sir\_op\_usuario.SRIO\_NOMBRE\_DE\_USUARIO = DECODE(USUARIO\_NOMBRE,  
TODOS,sir\_op\_usuario.SRIO\_NOMBRE\_DE\_USUARIO, USUARIO\_NOMBRE)

### Configuración Reporte 94 – INFORME RECUADOS DE CONSULTAS DEL DÍA POR USUARIO

SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES\_REPORT94\_PARAMFORM

```
<%-- + + + + + --%>
<%-- AHERRENO 02/03/2011
    Se captura el usuario logueado y se envía al reporte--%>
<%-- + + + + + --%>
<%
    Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
%>
<input type="hidden" name="<%=AWReportes.REPORTE_94_PARAM_USUARIOLOG%" id="<%=AWReportes.REPORTE_94_PARAM_USUARIOLOG%" value="<%=usuario.getUsername()%>">
```

SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java

```
/*AHERRENO 02/03/2011
Se agrega variable para capturar el usuario logueado*/
public static final String REPORTE_94_PARAM_CMDKEY = "reporte-94.jasper";
public static final String REPORTE_94_NOMBRE = "";
public static final String REPORTE_94_PARAM_PFECHAINI = "P_FECHA_INI";
public static final String REPORTE_94_PPARAM_PUSUARIONOMBRE = "P_USUARIO_NOMBRE";
public static final String REPORTE_94_PARAM_PCIRCULONOMBRE = "P_CIRCULO_NOMBRE";
public static final String REPORTE_94_PARAM_USUARIOLOG = "P_USUARIO_LOG";

/*AHERRENO 02/03/2011
Se agrega variable para capturar el usuario logueado*
if (REPORTE_94_BACK_ACTION.equals(accion)) {
    String[] itemIds = new String[]{};
    REPORTE_94_PARAM_PFECHAINI, REPORTE_94_PPARAM_PUSUARIONOMBRE,
    REPORTE_94_PARAM_PCIRCULONOMBRE, REPORTE_94_PARAM_USUARIOLOG
};
do_ReporteXX_Back_Action(itemIds, request);

/*AHERRENO 02/03/2011
Se agrega variable para capturar el usuario logueado*/
private void do_Reporte94_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {
    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();
    String param_01 = request.getParameter(REPORTE_94_PARAM_PFECHAINI);
    String param_02 = request.getParameter(REPORTE_94_PPARAM_PUSUARIONOMBRE);
    String param_03 = request.getParameter(REPORTE_94_PARAM_PCIRCULONOMBRE);
    String param_04 = request.getParameter(REPORTE_94_PARAM_USUARIOLOG);
    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {
        exception.addError("parametro: Fecha: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha : La fecha es inválida" + e.getMessage());
        }
    }
    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Usuario: Debe digitar un valor");
    }
    // apply validators
    if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    }
}
```

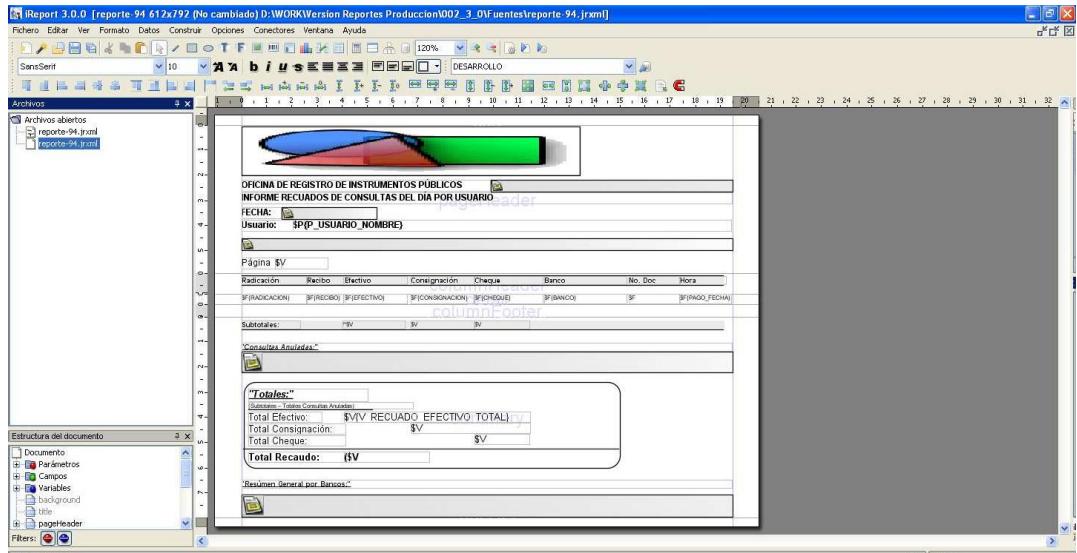
```

        if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
            exception.addError("parametro: Circulo: Debe digitar un valor");
        }
    } // apply validators
    if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
        exception.addError("parametro: Usuario Logueado: No hay usuario logueado");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
            exception.addError("parametro: Usuario Logueado: No hay usuario logueado");
        }
    }
    String[] itemIds = new String[]{
        REPORTE_94_PARAM_PFECHAINI, REPORTE_94_PARAM_PUSUARIONOMBRE,
        REPORTE_94_PARAM_PCIRCULONOMBRE, REPORTE_94_PARAM_USUARIOLOG
    };
    do_ReporteXX_SaveState(itemIds, request);
    // raise -----
    if (exception.getErrors().size() > 0) {
        throw exception;
    }
    // -----
    String[] params = new String[]{
        REPORTE_XX_PARAM_CMDKEY, REPORTE_94_PARAM_PFECHAINI,
        REPORTE_94_PARAM_PUSUARIONOMBRE, REPORTE_94_PARAM_PCIRCULONOMBRE,
        REPORTE_94_PARAM_USUARIOLOG
    };
    String[] values = new String[]{
        REPORTE_94_PARAM_CMDKEY, param_01, param_02, param_03, param_04
    };
    String url = makeUrl(params, values);
    reportsServices_ConfigureParameterUri(request, url);
    do_ReporteXX_RemoveState(itemIds, request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

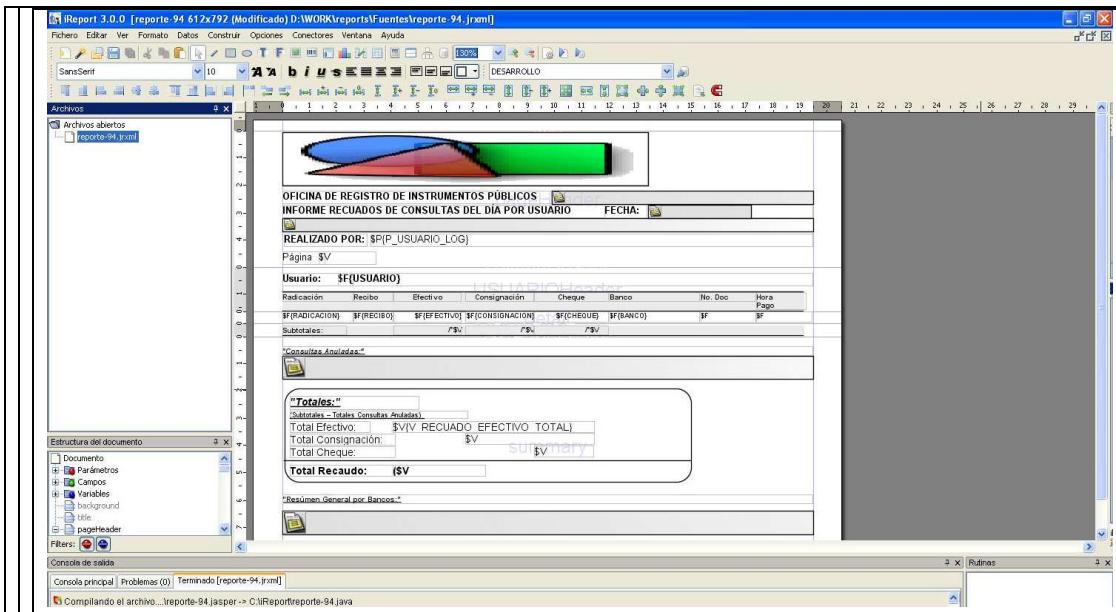
```

### JASPER – reporte-94.jasper

Anterior



Modificación



Se creó grupo USUARIO.

## 174 Reporte 51 Derechos - Total no coincide con los valores de la columna Informativo

### REQUERIMIENTO

La usuaria Rosalba González nos informa que en el reporte 51 de fecha 25/01/2010 el total de la columna de derechos no coincide, al sumar los valores que aparecen allí da un total de 4062920 y en el reporte aparece el valor de 4189420. Grupo Tunja

### DESARROLLO

Reporte 51 (Resumen diario Ingresos y Egresos)

Script Bases de Datos:

NINGUNO

Creación De Clases:

NINGUNO

Modificación de archivos:

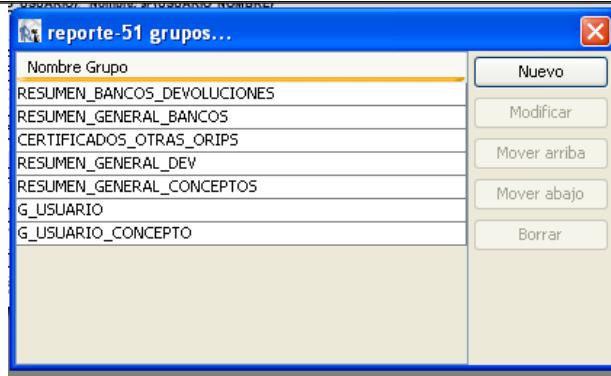
Se realizaron las siguientes modificaciones en el reporte 51 (Resumen diario Ingresos y Egresos) en la cual se agregaron 4 grupos:

Archivo reporte-51.jasper

```

RESUMEN_GENERAL_CONCEPTOS
RESUMEN_GENERAL_DEV
CERTIFICADOS_OTRAS_ORIPS
RESUMEN_GENERAL_BANCOS
RESUMEN_BANCOS_DEVOLUCIONES

```



En cada grupo se ubicó una sección ya que anteriormente se encontraban todas en la banda SUMMARY

### ANTES

RESUMEN GENERAL POR CONCEPTOS	summary
CERTIFICADOS OTRAS ORIPS	
RESUMEN GENERAL POR BANCOS	
RESUMEN GENERAL POR BANCOS DEVOLUCIONES	

### AHORA

RESUMEN GENERAL POR CONCEPTOS	RESUMEN GENERAL CONCEPTOS
CERTIFICADOS OTRAS ORIPS	CERTIFICADOS OTRAS ORIPS
RESUMEN GENERAL POR BANCOS	RESUMEN GENERAL BANCOS
RESUMEN GENERAL POR BANCOS DEVOLUCIONES	RESUMEN BANCOS DEVOLUCIONES

\* (vr Consignación + vr Cheque) - (Vr Aplicado Consignación + Vr Aplicado Cheque)

RESUMEN GENERAL POR CONCEPTOS

Concepto	Cant. Turnos	Cuantía	Derechos	Impuesto + Sanción	Liquidación	Efectivo	Consignación		Cheque/Nota Débito		Total Aplicado	Total Consignación + Cheque + Nota Débito
							Vr. Aplicado	Vr. Consignación	Vr. Aplicado	Vr. Cheque/ Nota Débito		
Cert. Asoc. A Cert. Masivos (7)	26	\$ 0	\$ 265,760	\$ 0	\$ 0	\$ 0	\$ 265,760	\$ 265,760	\$ 0	\$ 0	\$ 265,760	\$ 265,760
Cert. Asoc. A Nvas. Ent. (3)	3	\$ 0	\$ 720	\$ 0	\$ 0	\$ 0	\$ 720	\$ 720	\$ 0	\$ 0	\$ 720	\$ 720
Cert. Prensa y Trasl. Amplia. (7)	7	\$ 0	\$ 184,520	\$ 0	\$ 0	\$ 0	\$ 184,520	\$ 184,520	\$ 0	\$ 0	\$ 184,520	\$ 184,520
Cert. Radiados (384)	294	\$ 0	\$ 3,430,720	\$ 0	\$ 3,430,720	\$ 0	\$ 3,430,720	\$ 3,430,720	\$ 0	\$ 0	\$ 3,430,720	\$ 3,430,720
Cert. Radiados Asoc. (14)	14	\$ 0	\$ 144,960	\$ 0	\$ 0	\$ 0	\$ 144,960	\$ 144,960	\$ 0	\$ 0	\$ 144,960	\$ 144,960
Certificados Anulados (1)	1	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0
Certificados otras ORIPS (3)	3	\$ 0	\$ 36,240	\$ 0	\$ 0	\$ 0	\$ 36,240	\$ 36,240	\$ 0	\$ 0	\$ 36,240	\$ 36,240
Total		\$ 4,189,420	\$ 0	\$ 0	\$ 3,557,220	\$ 126,520	\$ 4,062,920	\$ 4,062,920	\$ 0	\$ 0	\$ 4,189,420	\$ 4,189,420

\* (vr Consignación + vr Cheque) - (Vr Aplicado Consignación + Vr Aplicado Cheque)

Se estaba sumando un valor  
que no se encontraba a un en  
la columna.

Se estaba sumando un valor  
que no se encontraba a un en  
la columna.

## AHORA

### RESUMEN GENERAL POR CONCEPTOS

Concepto	Cant. Turnos	Cuantia	Derechos	Impuesto + Sanción	Liquidación	Efectivo	Consignación		Cheque/Nota Débito		Total Aplicado	Total Consignación + Cheque + Nota Débito
							Vr. Aplicado	Vr. Consignación	Vr. Aplicado	Vr. Cheque/ Nota Débito		
Cert. Asoc. A Cert. Masivos (7)	26	\$ 0	\$ 289,760	\$ 0	\$ 285,760	\$ 0	\$ 289,920	\$ 289,920	\$ 0	\$ 0	\$ 289,920	\$ 289,920
Cert. Asoc. A Nvas. Ent. (3)	3	\$ 0	\$ 720	\$ 0	\$ 0	\$ 0	\$ 720	\$ 970	\$ 0	\$ 0	\$ 720	\$ 970
Cert. Prensa y Trad. Ampliada. (7)	7	\$ 0	\$ 184,520	\$ 0	\$ 0	\$ 0	\$ 184,520	\$ 184,520	\$ 0	\$ 0	\$ 184,520	\$ 184,520
Cert. Radiados (24)	284	\$ 0	\$ 3,430,720	\$ 0	\$ 3,430,720	\$ 0	\$ 3,430,720	\$ 3,430,730	\$ 0	\$ 0	\$ 3,430,720	\$ 3,430,730
Cert. Radiados Asoc. (14)	14	\$ 0	\$ 144,960	\$ 0	\$ 0	\$ 0	\$ 144,960	\$ 767,650	\$ 0	\$ 0	\$ 144,960	\$ 767,650
Certificados Anulados (1)	1	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0	\$ 0
Total General:			\$ 4,053,920	\$ 0	\$ 3,696,480	\$ 0	\$ 4,057,080	\$ 4,710,230	\$ 0	\$ 0	\$ 4,057,080	\$ 4,710,230

\* (vr Consignación + vr Cheque) - (Vr Aplicado Consignación + Vr Aplicado Cheque)

## 191 Pantalla Administrativa SGD

### REQUERIMIENTO

Con motivo de la creación en el sistema SIR del sistema de mensajería para la integración con el SGD, se requiere crear una pantalla administrativa que cumpla con varias funcionalidades así:

Que muestre los turnos generados en SIR, en los procesos definidos y que se encuentran en la cola de mensajería sin que hayan sido leídos por el Sistema de Gestión Documental.

Que muestre los turnos generados por SIR, en los procesos definidos y que fueron encolados y leídos por el Sistema de Gestión Documental.

Que tenga opción de reencolar un turno que haya sido generado en SIR y no haya sido encolado para ser leído por el Sistema de Gestión Documental.

Que la depuración de la mensajería de colas se realice quincenalmente para evitar que estas se desborden y causen inconvenientes.

### DESARROLLO

#### Análisis de Requerimiento:

Se agrega una nueva pantalla administrativa de nombre "Turnos Gestión Documental", muestra listado de turnos en cola de mensajería y poder reencolarlos, también se puede depurar la cola a partir de un rango de fecha.

Script Bases de Datos:

Script\_Caso\_Mantis\_6760.sql

Se crearon clases nuevas en SIR:

WEB-INF.jspSeguro.administracion.TURNO\_GESTION\_DOCUMENTAL.jsp

gov.sir.core.web.acciones.administracion.AWGestionDocumental.java

gov.sir.core.negocio.acciones.administracion.ANGestionDocumental.java

gov.sir.core.web.navegacion.administracion.ControlNavegacionGestionDocumental.java

gov.sir.core.eventos.administracion.EvnGestionDocumental.java

gov.sir.core.eventos.administracion.EvnRespGestionDocumental.java

gov.sir.core.web.acciones.excepciones.GestionDocumentalException.java

Se crearon clases nuevas en GeneralSIR:

com.iridium.generalSIR.comun.modelo.TurnoSGD.java

Se modificaron los siguientes archivos en SIR:

org.auriga.smart.negocio.xml  
org.auriga.smart.vistas.xml  
org.auriga.smart.web.xml

Se modificaron clases en GeneralSIR:

co.com.iridium.generalSIR.dao.ValidacionesSIRDAO.java

#### ARCHIVO XML:

##### org.auriga.smart.negocio.xml

##### Código Insertado Línea 420:

```
<mapping>
    <evento>gov.sir.core.eventos.administracion.EvnGestionDocumental</evento>
    <accion-negocio>gov.sir.core.negocio.acciones.administracion.ANGestionDocumental</accion-
negocio>
</mapping>
```

#### ARCHIVO XML:

##### org.auriga.smart.vistas.xml

##### Código Insertado Línea 4950:

```
<vista>
    <autenticacion>true</autenticacion>

    <excepcion>gov.sir.core.web.acciones.excepciones.GestionDocumentalException</excepcion>
        <nombre>admin.gestion.documental.view</nombre>
        <parametro><llave>error</llave><valor>/WEB-
INF/jspSeguro/comun/ERROR_MULTIPLE.jsp</valor></parametro>
        <parametro><llave>head</llave><valor>/WEB-
INF/jspSeguro/comun/HEADER_LOGOUT.jsp</valor></parametro>
        <parametro><llave>body</llave><valor>/WEB-
INF/jspSeguro/administracion/TURNO_GESTION_DOCUMENTAL.jsp</valor></parametro>
        <parametro><llave>bottom</llave><valor>/WEB-
INF/jspSeguro/comun/FOOTER_DEFAULT.jsp</valor></parametro>
</vista>
```

#### ARCHIVO XML:

##### org.auriga.smart.web.xml

##### Código Insertado Línea 4642:

```
<mapping>
    <url-origen>gestionDocumental.do</url-origen>
    <accion-
web>gov.sir.core.web.acciones.administracion.AWGestionDocumental</accion-web>
    <informacion-navegacion>

        <control>gov.sir.core.web.navegacion.administracion.ControlNavegacionGestionDocumental</
control>
            <navegacion>
                <resultado>TERMINA</resultado>
                <url-redireccion>admin.view</url-redireccion>
            </navegacion>
            <navegacion>
                <resultado>LISTA_TURNOS_GESTION_DOCUMENTAL</resultado>
                <url-redireccion>admin.gestion.documental.view</url-redireccion>
            </navegacion>
            <navegacion>
                <resultado>TURNO_REENCOLAR</resultado>
                <url-redireccion>admin.gestion.documental.view</url-redireccion>
            </navegacion>
```

```

        </navegacion>
    <navegacion>
        <resultado>SELECCIONA_TURNOS_POR_FECHA</resultado>
        <url-redireccion>admin.gestion.documental.view</url-redireccion>
    </navegacion>
    <navegacion>
        <resultado>DEPURAR_TURNOS_POR_FECHA</resultado>
        <url-redireccion>admin.gestion.documental.view</url-redireccion>
    </navegacion>
    </informacion-navegacion>
</mapping>
```

**CLASE EN GeneralSIR:**  
**co.com.iridium.generalSIR.dao.ValidacionesSIRDAO.java**

**Método:**

```
public List TurnosSGD_NoLeidos(String circulo) throws SQLException {
```

**Código Insertado Linea 86:**

```
/*
 * Metodo que genera una lista de turnos NO LEIDOS en el SGD a partir de un circulo.
 * @param String circulo
 * @return List turnos_sl
 * @author Ellery Davi Robles Gómez
 * Caso Mantis : 06760
 */
public List TurnosSGD_NoLeidos(String circulo) throws SQLException {
    PreparedStatement ps = null;
    ResultSet rs = null;
    StringBuilder sb = new StringBuilder();
    List turnos_sl = new ArrayList();
    try {
        sb.append("SELECT ID_CIRCULO, ID_PROCESO, ID_TURNO, FECHARADICACION, ENVIO FROM
ID5TURNO WHERE ID_CIRCULO = ? AND ENVIO = 0 ORDER BY FECHARADICACION");
        ps = conexionBaseDatos.prepareStatement(sb.toString());
        ps.setString(1, circulo);
        rs = ps.executeQuery();
        while (rs.next()){
            turnos_sl.add(new TurnoSGD(rs.getString(1), rs.getInt(2), rs.getString(3), rs.getString(4),
rs.getInt(5)));
        }
        return turnos_sl;
    } catch (SQLException sqle) {
        throw sqle;
    } finally{
        cerrarConexion(null,ps,rs);
    }
}
```

**Método:**

```
public List TurnosSGD_Leidos(String circulo) throws SQLException {
```

**Código Insertado Linea 117:**

```
/*
 * Metodo que genera una lista de turnos LEIDOS en el SGD a partir de un circulo.
 * @param String circulo
 * @return List turnos_l
 * @author Ellery Davi Robles Gómez
 * Caso Mantis : 06760
 */
public List TurnosSGD_Leidos(String circulo) throws SQLException {
    PreparedStatement ps = null;
    ResultSet rs = null;
    StringBuilder sb = new StringBuilder();
```

```

List turnos_l = new ArrayList();
try {
    sb.append("SELECT ID_CIRCULO, ID_PROCESO, ID_TURNO, FECHARADICACION, ENVIO FROM
ID5TURNO WHERE ID_CIRCULO = ? AND ENVIO = 1 ORDER BY FECHARADICACION");
    ps = conexionBaseDatos.prepareStatement(sb.toString());
    ps.setString(1, circulo);
    rs = ps.executeQuery();
    while (rs.next()) {
        turnos_l.add(new TurnoSGD(rs.getString(1), rs.getInt(2), rs.getString(3), rs.getString(4),
rs.getInt(5)));
    }
    return turnos_l;
} catch (SQLException sqle) {
    throw sqle;
}finally{
    cerrarConexion(null,ps,rs);
}
}

```

**Método:**

```
public String TurnosReencolarSGD(String idTurno) throws SQLException {
```

**Código Insertado Linea 146:**

```

/***
 * Metodo que modifica el valor del campo ENVIO de la tabla IDTURNO
 * en el esquema SG_DOCUMENTAL para reencolar un turno.
 * @param String idTurno
 * @return String exito
 * @author Ellery Davi Robles Gómez
 * Caso Mantis : 06760
 */
public String TurnosReencolarSGD(String idTurno) throws SQLException {
    PreparedStatement ps = null;
    StringBuilder sb = new StringBuilder();
    String exito = "0";
    int ok = 0;
    try {
        sb.append("UPDATE ID5TURNO SET ENVIO = 0 WHERE ID_TURNO = ?");
        ps = conexionBaseDatos.prepareStatement(sb.toString());
        ps.setString(1, idTurno);
        ok = ps.executeUpdate();
        if(ok == 1){exito = "1";}
        return exito;
    } catch (SQLException sqle) {
        throw sqle;
    }finally{
        cerrarConexion(null,ps,null);
    }
}

```

**Método:**

```
public String consultarTurnosPorFechaSGD(String fechaInicio, String fechaFin)
throws SQLException {
```

**Código Insertado Linea 172:**

```

/***
 * Metodo que consulta el numero de turnos en un rango de fechas.
 * @param String fechaInicio, fechaFin.
 * @return String numeroTurnos.
 * @author Ellery Davi Robles Gómez
 * Caso Mantis : 06760
 */
public String consultarTurnosPorFechaSGD(String fechaInicio, String fechaFin) throws SQLException
{

```

```

PreparedStatement ps = null;
ResultSet rs = null;
StringBuilder sb = new StringBuilder();
String numeroTurnos = "0";
try {
    //sb.append("select ID_CIRCULO, ID_PROCESO, ID_TURNO, FECHARADICACION, ENVIO from
    sg_documental.ID5TURNO IT where IT.fecharadicacion BETWEEN to_date(?, 'DD-MM-YYYY') AND
    to_date(?, 'DD-MM-YYYY')");
    sb.append("SELECT COUNT(ID_TURNO) AS TOTAL_TURNOS FROM
    SG_DOCUMENTAL.ID5TURNO IT WHERE IT.FECHARADICACION BETWEEN TO_DATE(?, 'DD-MM-YYYY')
    AND TO_DATE(?, 'DD-MM-YYYY')");
    ps = conexionBaseDatos.prepareStatement(sb.toString());
    ps.setString(1, fechaInicio);
    ps.setString(2, fechaFin);
    rs = ps.executeQuery();
    if (rs.next()) {
        numeroTurnos = rs.getString("TOTAL_TURNOS");
    }
    return numeroTurnos;
} catch (SQLException sqle) {
    throw sqle;
}finally{
    cerrarConexion(null,ps,null);
}
}

```

**Método:**

```

public String depurarTurnosPorFechaSGD(String fechaInicio, String fechaFin)
throws SQLException {

```

**Código Insertado Linea 202:**

```

/*
 * Metodo que depura una seleccion de turnos a partir de un rango de fechas.
 * @param String fechaInicio, fechaFin.
 * @return String exito.
 * @author Ellery Davi Robles Gómez
 * Caso Mantis : 06760
 */
public String depurarTurnosPorFechaSGD(String fechaInicio, String fechaFin) throws SQLException {
    PreparedStatement ps = null;
    StringBuilder sb = new StringBuilder();
    String exito = "0";
    int ok = 0;
    try {
        sb.append("DELETE FROM SG_DOCUMENTAL.ID5TURNO IT WHERE IT.FECHARADICACION
        BETWEEN TO_DATE(?, 'DD-MM-YYYY') AND TO_DATE(?, 'DD-MM-YYYY')");
        ps = conexionBaseDatos.prepareStatement(sb.toString());
        ps.setString(1, fechaInicio);
        ps.setString(2, fechaFin);
        ok = ps.executeUpdate();
        if(ok > 0){exito = "1";}
        return exito;
    } catch (SQLException sqle) {
        throw sqle;
    }finally{
        cerrarConexion(null,ps,null);
    }
}


```

**CLASE EN GeneralSIR:**

```

co.com.iridium.GeneralSIR.negocio.validaciones.ValidacionesSIR.java

```

**Método:**

```

public List TurnosSGD_NoLeidos(String circulo) throws GeneralSIRException{

```

**Código Insertado Linea 60:**

```
/**  
 * Metodo que determina una lista de turnos NO LEIDOS en el SGD a partir de un circulo.  
 * @param String circulo  
 * @return List  
 * @author Ellery Davi Robles Gómez  
 * Caso Mantis : 06760  
 */  
public List TurnosSGD_NoLeidos(String circulo) throws GeneralSIREException{  
    ValidacionesSIRDAO validacionesSIRDAO = new ValidacionesSIRDAO();  
    try {  
        return validacionesSIRDAO.TurnosSGD_NoLeidos(circulo);  
    } catch (SQLException ex) {  
        throw new GeneralSIREException("Ocurrio un error al tratar de realizar "  
            + "la consulta de los turnos del circulo: " + circulo + " :" + ex.getMessage(),ex);  
    }  
}
```

**Método:****public List TurnosSGD\_Leidos(String circulo) throws SQLException {****Código Insertado Linea 77:**

```
/**  
 * Metodo que determina una lista de turnos LEIDOS en el SGD a partir de un circulo.  
 * @param String circulo  
 * @return List  
 * @author Ellery Davi Robles Gómez  
 * Caso Mantis : 06760  
 */  
public List TurnosSGD_Leidos(String circulo) throws GeneralSIREException{  
    ValidacionesSIRDAO validacionesSIRDAO = new ValidacionesSIRDAO();  
    try {  
        return validacionesSIRDAO.TurnosSGD_Leidos(circulo);  
    } catch (SQLException ex) {  
        throw new GeneralSIREException("Ocurrio un error al tratar de realizar "  
            + "la consulta de los turnos del circulo: " + circulo + " :" + ex.getMessage(),ex);  
    }  
}
```

**Método:****public String TurnosReencolarSGD(String turno) throws GeneralSIREException{****Código Insertado Linea 95:**

```
/**  
 * Metodo que modifica el valor del campo ENVIO de la tabla IDTURNO  
 * en el esquema SG_DOCUMENTAL para reencolar un turno.  
 * @param String turno  
 * @return String  
 * @author Ellery Davi Robles Gómez  
 * Caso Mantis : 06760  
 */  
public String TurnosReencolarSGD(String turno) throws GeneralSIREException{  
    ValidacionesSIRDAO validacionesSIRDAO = new ValidacionesSIRDAO();  
    try{  
        return validacionesSIRDAO.TurnosReencolarSGD(turno);  
    } catch (SQLException ex){  
        throw new GeneralSIREException("Ocurrio un error al tratar de realizar "  
            + "la consulta de los turnos del circulo: " + turno + " :" + ex.getMessage(),ex);  
    }  
}
```

**Método:****public String TurnosPorFechaSGD(String fechaInicio, String fechaFin) throws  
GeneralSIREException{**

**Código Insertado Linea 112:**

```
/**  
 * Metodo que consulta el numero de turnos en un rango de fechas.  
 * @param String fechaInicio, fechaFin.  
 * @return String  
 * @author Ellery Davi Robles Gómez  
 * Caso Mantis : 06760  
 */  
public String TurnosPorFechaSGD(String fechaInicio, String fechaFin) throws GeneralSIREException{  
    ValidacionesSIRDAO validacionesSIRDAO = new ValidacionesSIRDAO();  
    try{  
        return validacionesSIRDAO.consultarTurnosPorFechaSGD(fechaInicio, fechaFin);  
    } catch (SQLException ex){  
        throw new GeneralSIREException("Ocurrio un error al tratar de realizar "  
            + "la consulta de los turnos a partir del día: " + fechaInicio + " hasta el día: " + fechaFin  
            + ". " + ex.getMessage(),ex);  
    }  
}
```

**Método:**

```
public String depurarTurnosPorFechaSGD(String fechalinicio, String fechaFin) throws  
SQLException {
```

**Código Insertado Linea 129:**

```
/**  
 * Metodo que depura una seleccion de turnos a partir de un rango de fechas.  
 * @param String fechaInicio, fechaFin.  
 * @return String  
 * @author Ellery Davi Robles Gómez  
 * Caso Mantis : 06760  
 */  
public String TurnosDepurarPorFechaSGD(String fechaInicio, String fechaFin) throws  
GeneralSIREException{  
    ValidacionesSIRDAO validacionesSIRDAO = new ValidacionesSIRDAO();  
    try{  
        return validacionesSIRDAO.depurarTurnosPorFechaSGD(fechaInicio, fechaFin);  
    } catch (SQLException ex){  
        throw new GeneralSIREException("Ocurrio un error al tratar de realizar "  
            + "la consulta de los turnos a partir del día: " + fechaInicio.toString() + " hasta el día: " +  
            fechaFin + ". " + ex.getMessage(),ex);  
    }  
}
```

**199 Folios derivados de temporales**  
**Informativo****REQUERIMIENTO**

Folios derivados de Temporales – Id Glpi 17627

Al realizar una segregación por el proceso de correcciones el sistema en la fase revisar y aprobar genera una inconsistencia con anotaciones temporales referidas en el manejo de los folios derivados. Es necesario corregir la segregación realizada en el proceso de correcciones ya sea creando una anotación nueva para segregar o segregar desde una anotación definitiva y revisar las anotaciones generadoras ya que crean inconsistencias en los datos temporales impidiendo el avance del turno.

Por favor escalar este caso a desarrollo para su solución general, por favor crear acta de requerimiento a desarrollo.

Grupo Cartagena

## **DESARROLLO**

Clase modificada: JDOGenieFolioDAO.java

Ubicación: \src\gov\sir\forseti\dao\impl\jdogenie\JDOGenieFolioDAO.java

Metodo: protected void hacerDefinitivasAnotaciones(FolioEnhanced folio, UsuarioEnhanced usuario, TurnoEnhanced turno, boolean validarTurno, PersistenceManager pm) throws DAOException.

CLASE
SIR-APP\src\gov\sir\forseti\dao\impl\jdogenie\JDOGenieFolioDAO.java
Método:
<b>protected void hacerDefinitivasAnotaciones(FolioEnhanced folio, UsuarioEnhanced usuario, TurnoEnhanced turno, boolean validarTurno, PersistenceManager pm) throws DAOException</b>
<b>Código Adicionado:</b>
<pre>/*  * @Autor:Guillermo Cabrera  * @fecha: 26-07-2010  * @cambio:Se obtiene el número de anotaciones padre con la anotación  * derivada (anotapadresize), si el numero de anotaciones padre es mayor de 0 no se puede  * eliminar la anotacion derivada.  * @mantis:4644  */ anotapadresize=anotaTmpDerivada.getAnotacionesPadreTMPs().size();</pre>
<b>Código Modificado:</b>
<pre>/*  * @Autor:Guillermo Cabrera  * @fecha: 26-07-2010  * @cambio:Se obtiene el número de anotaciones padre con la anotacion  * derivada (anotapadresize), si el numero de anotaciones padre es mayor de 0 no se puede  * eliminar la anotacion derivada.  * @mantis:4644 */ if (anotapadresize==0) {     for(int i=0;i&lt;anotacionesTmpDerivadas.size();i++)     {         AnotacionTMP anotaTmp=(AnotacionTMP)anotacionesTmpDerivadas.get(i);         if(anotaTmp.getIdAnotacionTmp()!=null)         {             if(turno == null    turno.getIdWorkflow().equals(anotaTmp.getNumRadicacion())   turno.getIdWorkflow().equals(anotaTmp.getIdWorkflow()))             {                 pm.deletePersistent(anotaTmp);             }         }     } }</pre>

## **203 Desfase de Id anotacion temporales Informativo**

### **REQUERIMIENTO**

Requerimiento GLPI No 17411

Turno: 2010-260-6-2766

Matricula(s): 260-216029, 260-265726, 260-265727.

Motivo Inconsistencia: El turno no avanza de la fase FIRMAR REGISTRO el sistema notifica error. Se adjunta imagen de error. Grupo Cúcuta

En el proceso de segregación se crean las anotaciones temporales para los folios segregados, con desfase en el ID\_ANOTACION con respecto al número de orden de la

anotación.

### **DESARROLLO**

Para este error reportado, no se pudo replicar en las pruebas realizadas.

Se realizaron pruebas en la aplicación desplegada en la maquina <http://192.168.202.9:8100/SNR/inicio.view>, [^] con el mismo turno y matricula reportada en el caso 5299, pero la aplicación se comportó correctamente y las segregaciones se realizaron sin ningún inconveniente.

Datos de pruebas

TURNOS: 2010-240-6-9991

MATRICULA: 240-25197

USUARIO: ANA.PATINO

CIRUCULO: PASTO

Igualmente se realizó una validación en el paquete SIR.SP\_SP para validar los id de las anotaciones para los folios hijos o derivados.

### **PROCEDIMIENTO ALMACENADO**

#### **SIR. SP\_SG**

##### **Código Adicionado**

```
--@AUTOR:GUILLERMO CABRERA  
--@FECHA:19-07-2010  
--@CAMBIO:SE ADICIONA PARA VERIFICAR QUE EL ID DE LA ANOTACIÓN SEA EL CORRECTO  
-ES DECIR QUE AL CREARSE UN FOLIO NUEVO, ESTE SE CREE CON EL ID DE ANOTACION EN 1  
--Y NO COMIENCE EN OTRO ID COMO LOS CASOS REPORTADOS  
--MANTIS 5306 ACTA 203.
```

```
SELECT COUNT(ID_ANOTACION_TMP)  
INTO N_COMPARA_ID  
FROM SIR.SIR_NE_ANOTACION_TMP  
WHERE ID_MATRICULA = V_ID_MATRICULA_HIJA;  
  
IF V_CONTADOR_ID_ANOTACION != N_COMPARA_ID+1 THEN  
    V_CONTADOR_ID_ANOTACION:=N_COMPARA_ID;  
END IF;
```

### **212 Pantalla de administración - Ver detalles de turno**

#### **Informativo**

### **REQUERIMIENTO**

Se debe corregir la pantalla de Administración "Ver Detalles de Turnos", ya que al igual que el reporte está sugiriendo que el usuario MARIA.BENAVIDES realizó la reimpresión.

Caso de referencia GLPI No 20294.

Se realizan pruebas en ambiente de calidad con el circulo Pasto -240  
SOLICITUD ROSA.PENA  
CONFRONTAR MARIA.GUERRERO  
REPARTO MARIA.GUERRERO  
CALIFICACION BOSCO.ROSAS  
MESA CONTROL MARIA.GUERRERO  
FIRMA REGISTRO CARLOS.OJEDA  
CERTIFICADOS ASO LILIA.CUELLAR  
ENTREGA MARIA.BENAVIDES

Una de las inconsistencias que se está presentando es con los certificados asociados a turnos de Registro. Ejm se radica el turno 2010-240-6-10922 con matriculas:

240-57813  
240-164068  
240-178802  
240-178803

Las cuales se agregan como Certificados asociados EXENTOS, Genera los certificados asociados:

2010-240-1-46173  
2010-240-1-46174  
2010-240-1-46175  
2010-240-1-46176

Se avanza el turno hasta la fase de Certificados Asociados, se imprimen los certificados en esta fase y se avanza el turno a fase de entrega, antes de realizar la entrega, por pantallas administrativas se reimprimen los certificados asociados con la usuaria LILIA.CUELLAR, Se verifica por pantallas administrativas y se observa la reimpresión por la usuaria LILIA.CUELLAR, se realiza la entrega del turno de Registro 2010-240-6-10922, Se consulta el turno de certificado Asociado 2010-240-1-46173 y se observa que se modifico el usuario que reimprimió los certificados por MARIA.BENAVIDES, lo cual no es correcto porque lo imprimió LILIA.CUELLAR

## **DESARROLLO**

CLASE o ARCHIVO JSP:	
\WORK\SIR-APP\web\WEB-INF\jspSeguro\administracion\TURNO_REGISTRO_DETALLES_VISTA.jsp	
<b>Línea 1046:</b>	
/** * @author: Fernando Padilla * @change: 5680: Acta - Requerimiento No 212 - Pantalla de administración - Ver detalles de turno, * Se modifica para que sea tomado el nombre de usuario de quien atendio en la fase correspondiente. */ if (turnoAux.getUsuario()!=null){ nombreUsuario = turnoAux.getUsuario().getUsername(); }	\WORK\SIR-APP\web\WEB-INF\jspSeguro\administracion\TURNO_CERTIFICADOS_DETALLES_VISTA.jsp
<b>Línea 1694:</b>	
/** * @author: Fernando Padilla * @change: 5680: Acta - Requerimiento No 212 - Pantalla de administración - Ver detalles de turno, * Se modifica para que sea tomado el nombre de usuario de quien atendio en la fase correspondiente. */ if (turnoAux.getUsuario()!=null){ nombreUsuario = turnoAux.getUsuario().getUsername(); }	

## **215 Ver Detalles de Turno - Turnos Devolucion Negados**

### **REQUERIMIENTO**

El sistema debe mostrar en detalles de turno; tanto para el proceso de documento como para el proceso de devolución de dinero el turno asociado cuando se encuentre en cualquiera de las fases del proceso de devoluciones, así mismo cuando el turno de

devolución de dinero es NEGADO; debe reflejarse en la consulta del turno de registro con su respectiva resolución.

#### **Análisis de Requerimiento:**

En el Proceso de devoluciones cuando el turno es negado se elimina la relación que este posee con su turno padre y si se quiere continuar con el proceso en el turno de devolución se debe asociar un turno de registro.

Cuando se quiere observar los turnos negados en ver detalles de turno de un turno de registro no se puede obtener los turnos de devolución negados relacionados por que dicha relación desapareció en el momento en que el turno fue negado.

Por lo tanto solicito aclaración si la eliminación de la relación obedece a una regla de negocio o si esta es un bug de la aplicación.

Se envió correo el día 02 de febrero a los funcionarios Luis Romero; Imelda Muñoz, Yamile Moreno; e interventoria. Solicitando la información y aclaración pertinente al caso.

#### **Respuesta a solicitud de información**

Se recibe correo el 03 de febrero por parte de los funcionarios de la SNR donde hacen la aclaración hecha anteriormente.

Se requiere ajustar la siguiente inconsistencia que presenta el aplicativo en la consulta de turnos de registro y turnos de devolución de dinero":

"Cuando un turno de registro es vinculado a una solicitud de Devolución de Dinero y esta tiene resolución Negado, se requiere que en la pantalla ver detalle de turno, al consultar el turno de registro se refleje el numero de turno de la solicitud de Devolución de dinero, así mismo al consultar el turno de devolución de dinero el sistema debe reflejar el turno de registro asociado a la solicitud " .

Lo anterior como lo indica el formato de requerimiento 215.

Se recibe correo enviado por el funcionario Luis romero haciendo la siguiente aclaración

De manera atenta me permito recalcar que el aplicativo debe conservar la relación entre el turno de registro y el turno del proceso de devolución NEGADO.

En ningún momento, hasta donde tengo entendido, se definió por parte funcional o técnica que se quitara la relación entre ellos cuando la devolución de dinero es NEGADA.

De otra parte, es importante establecer la relación entre los turnos de registro y los de devolución NEGADOS por medio de búsquedas en la base de datos, para así realizar las actualizaciones del caso, por lo que solicito se lleve a cabo esta actividad, favor enviar la información consultada a la SNR para realizar los análisis del caso y aprobar la actualización.

#### ***DESARROLLO***

**ARCHIVO JSP:**

**WEB-**

## INF.jspSeguro.administracion.TURNO\_REGISTRO\_DETALLES\_VISTA.jsp

### Código Insertado Línea 249:

```
/*
 * Obtiene la lista de turnos de devolucion asociados de la session.
 * @author: Julio Alcazar
 * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados
 */
List turnosDevoluciones= (List) session.getAttribute(AWTrasladoTurno.LISTA_TURNOS_DEVOLUCION);
```

### Código Insertado Línea 676:

```
<!--
 * Tabla para mostrar los turnos de devolución asociado
 * @author: Julio Alcazar
 * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion
Negados
!-->
<% if (turnosDevoluciones != null) {
%>
<table width="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
    <td class="bgnsub"><p>Turnos de Devoluciones Asociados</p></td>
    <td width="16" class="bgnsub"><img
src=<%=request.getContextPath()%>/jsp/images/ico_matriculas.gif" width="16" height="21"></td>
    <td width="15"><img name="sub_r1_c4"
src=<%=request.getContextPath()%>/jsp/images/sub_r1_c4.gif" width="15" height="22"
border="0" alt=""></td>
</tr>
</table>
<table width="100%" class="camposform">
<%
Iterator itera= turnosDevoluciones.iterator();
while(itera.hasNext()){
    Turno temp2=(Turno)itera.next();
    List turnohistoria = temp2.getHistorials();
    String respuestadev = "";
    Iterator iter = turnohistoria.iterator();
    while(iter.hasNext()){
        TurnoHistoria turnoAux = (TurnoHistoria) iter.next();
        if(turnoAux.getFase() != null && ((turnoAux.getFase()).equals(CFase.DEV_ANALISIS ))){
            respuestadev = turnoAux.getRespuesta();
        }
    }
    if(respuestadev != null && respuestadev!=""){
        if(respuestadev.equals(CRespuesta.CONFIRMAR)){
            respuestadev = "RESOLUCIÓN: APROBADA";
        }else if(respuestadev.equals(CRespuesta.NEGAR)){
            respuestadev = "RESOLUCIÓN: NEGADA";
        }
    }
}
%>
<tr>
    <td width="220">Turno : </td>
    <td class="campositem"> <%= temp2.getIdWorkflow() %> </td>
    <td class="campositem"> <%= respuestadev!=null?respuestadev:" " %> </td>
</tr>
<% } %>
</table>
```

**CLASE:**

**gov.sir.core.negocio.acciones.administracion.AnTrasladoTurno.java**

**Método:**

**private EvnRespTrasladoTurno  
consultarTurnoPorIdentificador(EvnTrasladoTurno evento)  
throws EventoException {**

**Código Insertado Linea 507:**

```
/*
 * inicializar variable turnosDevoluciones.
 * @author: Julio Alcazar
 * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados
 */
List turnosDevoluciones = null;
```

**Código Insertado Linea 670:**

```
/*
 * Obtiene los turnos de devolucion asociados con el turno ingresado como parámetro.
 * @author: Julio Alcazar
 * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados
 */
turnosDevoluciones = hermod.getTurnosDevolucion(turno);
if (turnosDevoluciones != null){
    Iterator itera = turnosDevoluciones.iterator();
    List templist = new ArrayList();
    while(itera.hasNext()){
        Turno turnotemp = (Turno) itera.next();
        templist.add(hermod.getTurnobyWF(turnotemp.getIdWorkflow()));
    }
    turnosDevoluciones = templist;
}
```

**Código Insertado Linea 857:**

```
/*
 * Set turnosDevoluciones en el evento respuesta.
 * @author: Julio Alcazar
 * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados
 */
evRespuesta.setTurnosDevoluciones(turnosDevoluciones);
```

**CLASE:**

**gov.sir.core.web.acciones.administracion.AWTrasladoTurno.java**

**Código Insertado Linea 207:**

```
/** Constante que identifica la variable del HttpSession donde se almacena la lista de turnos de las
solicitudes asociadas
* @author: Julio Alcazar
* @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados
*/
public static final String LISTA_TURNOS_DEVOLUCION = "LISTA_TURNOS_DEVOLUCION";
```

**Método:**

**public void doEnd(HttpServletRequest request, EventoRespuesta evento)**  
{

**Código Insertado Linea 1587:**

```
/*
 * se guarda en session el listado de turnos de devolución
 * @author: Julio Alcazar
 * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados
 */
session.setAttribute(AWTrasladoTurno.LISTA_TURNOS_DEVOLUCION,
respuesta.getTurnosDevoluciones());
```

**CLASE:**

**gov.sir.core.eventos.administracion.EvnRespTrasladoTurno.java****Código Insertado Linea 96:**

```
/*
 * Se agrega al evento la propiedad turnosDevoluciones y sus metodos get y set
 * @author: Julio Alcazar
 * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados
 */
private List turnosDevoluciones;

public List getTurnosDevoluciones() {
    return turnosDevoluciones;
}

public void setTurnosDevoluciones(List turnosDevoluciones) {
    this.turnosDevoluciones = turnosDevoluciones;
}
```

**CLASE:****gov.sir.hermode.HermodeException.java****Código Insertado Linea 1892:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Cadena de exception para metodo getTurnosDevolucion(Turno turno)
 * Caso Mantis : 07393
 */
public static String TURNOS_DEVOLUCION_NO_OBTENIDOS = "No fue posible obtener listado de
turnos de devolución";
```

**CLASE:****gov.sir.hermode.HermodeService.java****Código Insertado Linea 9671:**

```
/**
 * Obtiene los turnos de devolución asociados con el turno ingresado como parámetro.
 * @author: Julio Alcazar
 * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados
 * @param Turno Identificador del turno del cual se consultarán los turnos siguientes.
 * @return lista de objetos <code>Turno</code>
 * @see gov.sir.core.negocio.modelo.Turno
 * @throws <code>HermodeException</code>
 */
public List getTurnosDevolucion(Turno turno) throws HermodeException
{
    try
    {
        return factory.getTurnosDAO().getTurnosDevolucion(turno);
    }
    catch (DAOException e)
    {
        HermodeException he = new
        HermodeException(HermodeException.TURNOS_DEVOLUCION_NO_OBTENIDOS, e);
        ExceptionPrinter ep = new ExceptionPrinter(he.fillInStackTrace());
        Log.getInstance().error(this.getClass(), he.getMessage(), e);
        throw he;
    }
}
```

**CLASE:****gov.sir.hermode.dao.TurnosDAO.java****Código Insertado Linea 776:**

```
/**
 * Obtiene los turnos de devolucion asociados con el turno ingresado como parámetro.
 * @author: Julio Alcazar
```

<pre> * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados * @param Turno Identificador del turno del cual se consultarán los turnos siguientes. * @return lista de objetos &lt;code&gt;Turno&lt;/code&gt; * @see gov.sir.core.negocio.modelo.Turno * @throws &lt;code&gt;DAOException&lt;/code&gt; */ List getTurnosDevolucion(Turno turno) throws DAOException; </pre>	<b>CLASE:</b> <b>gov.sir.hermes.dao.impl.jdogenie.JDOTurnosDAO.java</b> <b>Nuevo Método:</b> <b>public List getTurnosDevolucion(Turno turno) throws DAOException {</b>
<b>Código Insertado Línea 12161:</b>	
<pre> /**  * Obtiene (si existe) el &lt;code&gt;Turno &lt;/code&gt; del cual un turno es turnoAnterior  * @param Turno.  * @param pm PersistenceManager de la transaccion  * @return Objeto &lt;code&gt;TurnoEnhanced&lt;/code&gt; con sus atributos, del cual es turno anterior  * el turno con idWorkflow pasado como parámetro.  * @throws DAOException  * Obtiene los turnos de devolucion asociados con el turno ingresado como parámetro.  * @author: Julio Alcazar  * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados */ public List getTurnosDevolucion(Turno turno) throws DAOException {     TurnoEnhanced turnoSiguiente = null;     List listaTurnos = new ArrayList();     PersistenceManager pm = AdministradorPM.getPM();      try {         Query query = pm.newQuery(TurnoEnhanced.class);         query.declareParameters("String idWork, String idCir");         query.setFilter("this.solicitud.turnoAnterior.idWorkflow==idWork &amp;&amp;\n" +                        "this.solicitud.turnoAnterior.idCirculo==idCir " );         Collection col = (Collection) query.execute( turno.getIdWorkflow(), turno.getIdCirculo());         Iterator it = col.iterator();          while (it.hasNext()) {             turnoSiguiente = (TurnoEnhanced) it.next();             if(Long.parseLong(CProceso.PROCESO_DEVOLUCIONES)=== turnoSiguiente.getIdProceso())             {                 pm.makeTransient(turnoSiguiente);                 listaTurnos.add(turnoSiguiente);             }         }         query.close(col);      } catch (JDOObjectNotFoundException e) {         Log.getInstance().error(JDOTurnosDAO.class,e.getMessage(), e);         throw new DAOException(e.getMessage(), e);     } catch (JDOException e) { </pre>	

```

        Log.getInstance().error(JDOTurnosDAO.class,e.getMessage(), e);
        throw new DAOException(e.getMessage(), e);
    }
    finally {
        pm.close();
    }
    List listaTransfer = TransferUtils.makeTransferAll(listaTurnos);

    return listaTransfer;
}

```

**CLASE:**

**gov.sir.hermode.ejb.HermodeServiceEJBBean.java**

**Código Insertado Línea 3769:**

```

/**
 * @see
gov.sir.hermode.interfaz.HermodeServiceInterface#getTurnosDevolucion(gov.sir.core.negocio.modelo.Tur
no)
 * @ejb:interface-method view-type="both"
 * Obtiene los turnos de devolución asociados con el turno ingresado como parámetro.
 * @author: Julio Alcazar
 * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados
 */
public List getTurnosDevolucion(Turno turno) throws HermodeException {
    return hermod.getTurnosDevolucion(turno);
}

```

**CLASE:**

**gov.sir.hermode.interfaz.HermodeServiceInterface.java**

**Código Insertado Línea 4614:**

```

/**
 * Obtiene los turnos de devolución asociados con el turno ingresado como parámetro.
 * @author: Julio Alcazar
 * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados
 * @param Turno Identificador del turno del cual se consultarán los turnos siguientes.
 * @return lista de objetos <code>Turno</code>
 * @see gov.sir.core.negocio.modelo.Turno
 * @throws <code>Throwable</code>
 */
public List getTurnosDevolucion(Turno turno) throws Throwable;

```

**CLASE:**

**gov.sir.hermode.dao.impl.jdogenie.JDOSolicitudesDAO.java**

**Método:**

**public boolean setAprobacionSolicitud(SolicitudPk solID, boolean  
aprobada) throws DAOException {**

**Cambio Código Línea 7454:**

```

if(solDev.getTurnoAnterior()!=null){
    solDev.setTurnoAnterior(null);
}else{

```

**Por:**

```

/*
 * Se elimina la posibilidad de eliminar la relación de los turnos de devolución con su turno padre
 * @author: Julio Alcazar
 * @change: 7393: Acta - Requerimiento No 215 - Ver Detalles de Turno - Turnos Devolucion Negados
 */
if(solDev.getTurnoAnterior() == null){

```

**216 Error proceso correcciones - Mayor Valor - Antiguo Sistema - Ampliacion  
Tradicion**

**REQUERIMIENTO**

Al radicar turno de Correcciones y avanzarlo a los procesos de Mayor Valor, Antiguo Sistema y al enviarlo al proceso de Actuaciones administrativas este genera mensaje NULL, realizar prueba donde los usuarios que manejan antiguo sistema y responsable correcciones sean diferentes.

**DESARROLLO**

Se modificaron los siguientes archivos en SIR:

sir.core.negocio.acciones.correccion.ANRevision.java

**Código Eliminado Línea 527:**

```
LlaveBloqueo llaveBloqueo = new LlaveBloqueo();
```

**Código Modificado Línea 537:**

```
if (!(evento.getRespuestaWF().equals(EvnRevision.ACTUACION_ADMINISTRATIVA))) {  
    forseti.delegarBloqueoFolios(turnoID, usr);  
}
```

**218 Pantallas administrativas - Reimpresion Constancia y Recibos Fotocopias****REQUERIMIENTO**

En la reimpresión de recibos (constancias) de Fotocopias, se genera mensaje “Error al imprimir el folio: null” y no imprime, en la reimpresión de recibos de pago de fotocopias genera mensaje “.Error al obtener el último turno impreso: No fue posible obtener el último del usuario para el proceso seleccionado”, realizar prueba donde los usuarios que manejen el proceso de fotocopias sean diferentes.

**DESARROLLO**

Existían variables que retornan null, y en el archivo Server.java no se estaba seteando el turno, se implementó en la reimpresión la clase ImprimibleFotocopiaCrearSolicitud y los parámetros necesarios y se seteo la frase "Esta es una impresión"

Se modificaron los siguientes archivos en SIR:

sir.core.negocio.acciones.administracion.ANIImprimirFolio.java

sir.core.negocio.modelo.imprimibles.ImprimibleFotocopiaPago\_SolicitudData.java

sir.core.negocio.modelo.imprimibles.ImprimibleFotocopiaCrearSolicitud.java

sir.print.server.Server.java

**ARCHIVO JAVA:**

sir.core.negocio.acciones.administracion.ANIImprimirFolio.java

**Código Insertado Línea 81:**

```
import gov.sir.core.negocio.modelo.imprimibles.ImprimibleFotocopiaCrearSolicitud;
```

**Código Insertado Línea 334:**

```
String fechaImpresion = this.getFechaImpresion( new Date() );
```

**Código Insertado Línea 412:**

```
ImprimibleFotocopiaCrearSolicitud impRec  
        = new ImprimibleFotocopiaCrearSolicitud( turno, solicitud_fotocopia,  
circulo ,CTipoImprimible.RECIBO);  
        impRec.setTamanoCarta(false);  
        impRec.setFechaImpresionServidorFormatted( fechaImpresion );  
        impRec.setUsuario( user );  
        impRec.setReimpresion(true);  
        bundle = new Bundle(impRec);
```

**ARCHIVO JAVA:**

sir.core.negocio.modelo.imprimibles.ImprimibleFotocopiaPago\_SolicitudData.java

**Código Insertado Linea 258:**

```
/* Ellery Robles caso Mantis 07114 Acta - Requerimiento No 218 - Reimpresión de Recibo de
Fotocopias
 * void textoReimpresion() Imprime nota reimpresión.
 */
private void textoReimpresion() {
    String textoReimpresion = "Esta es una reimpresión ";
    String recibo = "";
    if (!(this.pago == null)){
        recibo = "del recibo " + this.pago.getLastNumRecibo();
    }
    this.imprimirLinea(ImprimibleConstantes.PLANO, textoReimpresion + recibo);
}

/* Ellery Robles caso Mantis 07114 Acta - Requerimiento No 218 - Reimpresión de Recibo de
Fotocopias
 * Set reimpresión.
 */
public void setReimpresion(boolean b) {
    reimpresión = b;
}
```

**ARCHIVO JAVA:**

sir.core.negocio.modelo.imprimibles.ImprimibleFotocopiaCrearSolicitudData.java

**Código Insertado Linea 6:**

```
import gov.sir.core.negocio.modelo.Pago;
```

**Código Insertado Linea 45:**

```
/* JAlcazar caso Mantis 05014 Acta - Requerimiento No 192 - Reimpresión de Recibo de
Fotocopias
 * Constante que indica que es una reimpresión.
 */
private boolean reimpción;
```

```
protected Pago pago;
```

**Código Insertado Linea 144:**

```
/* Ellery Robles caso Mantis 07114 Acta - Requerimiento No 218 - Reimpresión de Recibo de
Fotocopias
 * Imprime nota si es una reimpresión.
 */
if(this.reimpresion){
    this.textoReimpresion();
}
```

**Código Insertado Linea 555:**

```
/* Ellery Robles caso Mantis 07114 Acta - Requerimiento No 218 - Reimpresión de Recibo de
Fotocopias
 * getTurno()
 */
public Turno getTurno() {
    return turno;
}
```

```
/* Ellery Robles caso Mantis 07114 Acta - Requerimiento No 218 - Reimpresión de Recibo de
```

```

Fotocopias
    * void textoReimpresion() Imprime nota reimpresión.
    */
private void textoReimpresion()
{
    String textoReimpresion = "Esta es una reimpresión ";
    String recibo = "";
    if (!(this.pago == null)){
        recibo = "del recibo " + this.pago.getLastNumRecibo();
    }
    this.imprimirLinea(ImprimibleConstantes.PLANO, textoReimpresion + recibo);
}

/* Ellery Robles caso Mantis 07114 Acta - Requerimiento No 218 - Reimpresión de Recibo de
Fotocopias
    * Set reimpresion.
    */
public void setReimpresion(boolean b) {
    reimpresion = b;
}

```

### ARCHIVO JAVA:

#### sir.print.server.Server.java

**Código Insertado Linea 38:**

```
import gov.sir.core.negocio.modelo.imprimibles.ImprimibleFotocopiaCrearSolicitud;
```

**Código Insertado Linea 471:**

```
/** Ellery Robles caso Mantis 07114 Acta - Requerimiento No 218 - Reimpresion de consultas
     * set del turno al imprimible
     * imp.setTurno(impfotosol.getTurno().getIdWorkflow());
     */
    if (imprimibleTemp instanceof ImprimibleFotocopiaCrearSolicitud ) {
        ImprimibleFotocopiaCrearSolicitud impfotosol =
(ImprimibleFotocopiaCrearSolicitud) imprimibleTemp;
        if (impfotosol != null && impfotosol.getTurno() != null &&
impfotosol.getTurno().getIdWorkflow() != null) {
            imp.setTurno(impfotosol.getTurno().getIdWorkflow());
        }
    }
}
```

### 219 REPORTE ACTOS MAYOR CUANTIA-imm

**REQUERIMIENTO**

REPORTE ACTOS DE NATURALEZA JURIDICA POR CUANTIA

Se requiere crear un reporte que refleje la siguiente información.: debe corresponder a “Reporte Actos de Naturaleza jurídica por cuantía”.

El encabezado debe incluir los mismos ítems estándar de los demás reportes, pero el título Los parámetros de este reporte son: Rango de fechas (inicio y final), circulo regstral (debe traer predeterminado el círculo donde se genera el reporte) y un campo de cuantía (numérico más de diez dígitos).

Este reporte corresponderá a las solicitudes radicadas de registro en un rango de fechas (Proceso 6), cuya cuantía sea igual o superior a la cuantía ingresada como parámetro. Debe verificarse que los turnos de registro, no se encuentren anulados.

Las columnas de este reporte son:

Turno de registro; fecha de radicación; código y descripción del acto liquidado; cuantía (valor ingresado en el acto de liquidación); tipo de documento; numero de documento, fecha del documento; oficina Origen, ciudad oficina de origen.

Debe totalizar, la cantidad de turnos de registro y las cuantías asociadas a cada acto.

Se requiere que se adhiera al Rol ya creado en el Sistema de Información Registral (SIR), llamado CONSULTA \_ESTADISTICA \_SNR.

Se hacen las siguientes precisiones:

-Para este reporte los actos con cuantía son los que aplican.

- La suma de los actos (o acto) de cada turno, el resultado total debe ser mayor a \$7.000.000.000.00

## ***DESARROLLO***

### **Análisis de Requerimiento:**

Script Bases de Datos: Procedimiento SP\_REPORTE\_MAYOR\_CUANTIA.

Creación De Clases:

NINGUNA (el mismo formato de modificación clases o jsp).

Se modificaron los siguientes archivos:

Se desarrolló el procedimiento almacenado SP\_REPORTE\_MAYOR\_CUANTIA encargado del reporte actos mayores por cuantía.

<b>Procedimiento SP_REPORTE_MAYOR_CUANTIA</b>
create or replace PROCEDURE /***/ /* GUILLERMO CABRERA 08/02/2011 REPORTE: REPORTE ACTOS MAYOR CUANTIA */ /***/ SP_REPORTE_MAYOR_CUANTIA(SEC_PROCESO NUMBER, P_ID_CIRCULO VARCHAR2, P_FECHA_INICIO VARCHAR2, P_FECHA_FIN VARCHAR2, P_CUANTIA VARCHAR2) IS BEGIN INSERT INTO SIR.SIR_OP_Reporte_MAYOR_CUANTIA(SEC_PROCESO,TURNO,FECHA_RADICACION,CODIGO_ACTO, ACTO_NOMBRE,CUANTIA,DOCUMENTO,NUMERO,FECHA_DOC,OFICINA,MUNICIPIO) (SELECT SEC_PROCESO,T.TRNO_ID_WORKFLOW,S.SLCT_FECHA_SOLICITUD,AC.id_tipo_acto, ta.tpcet_nombre,ac.acto_cuantia, td.tpdc_nombre, d.dcmn_numero, d.dcmn_fecha, oo.fcrg_nombre, M.mncp_nombre FROM SIR.SIR_OP_SOLICITUD S INNER JOIN SIR.SIR_OP_ACTO AC ON S.ID_SOLICITUD = AC.ID_SOLICITUD INNER JOIN SIR.SIR_OP_TIPO_ACTO TA ON TA.ID TIPO_ACTO = AC.ID TIPO_ACTO INNER JOIN SIR.SIR_OP_TURNO T ON S.ID_SOLICITUD = T.ID_SOLICITUD AND S.ID PROCESO = T.ID PROCESO AND S.ID CIRCULO = T.ID CIRCULO INNER JOIN SIR.SIR_NE_DOCUMENTO D ON D.ID DOCUMENTO = S.ID DOCUMENTO INNER JOIN SIR.sir_ne_tipo_documento TD ON d.id_tipo_documento= td.id_tipo_documento INNER JOIN sir_ne_oficina_origen OO ON oo.id_oficina_origen = d.id_oficina_origen INNER JOIN SIR.SIR_NE_MUNICIPIO M ON M.ID_DEPARTAMENTO = OO.ID_DEPARTAMENTO AND M.ID_MUNICIPIO = OO.ID_MUNICIPIO WHERE S.ID_CIRCULO = P_ID_CIRCULO AND S.ID PROCESO = '6' AND TA.ID TIPO CALCULO = 'P' AND T.TRNO_ANULADO = 'N' AND S.SLCT_FECHA_SOLICITUD BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-dd')+1 AND ac.acto_cuantia>=P_CUANTIA); COMMIT; END;

CONFIGURACION PANTALLA "REPORTE 164"

	<pre>INSERT INTO SIR.SIR_OP_PANTALLA_ADMINISTRATIVA (ID_PANTALLA_ADMINISTRATIVA, id_tipo_pantalla, ptdm_nombre, ptdm_vista, ptdm_pagina) VALUES (1120, 13,'Reporte 164: Reporte Actos de Naturaleza Jurídica por cuantía','admin.reportes.report-164.param-form.view','PG_REPORTES')</pre>	
<b>ASIGNACION DE LA PANTALLA "REPORTE 164" AL ROL SIR_ROL_CONSULTA_ESTADISTICA_SNR</b>		
INSERT INTO SIR.SIR_OP_ROL_PANTALLA (ID_ROL, id_pantalla_administrativa) VALUES ('SIR_ROL_CONSULTA_ESTADISTICA_SNR', 1120)		
<p style="text-align: center;"><b>Configuración Reporte 164 – Reporte Actos de Naturaleza Jurídica por Cuantía</b></p> <p style="text-align: center;"><b>SIR-APP\src\org.auriga.smart.vistas.xml</b></p> <pre>&lt;!-- AHERRENTO 23/02/2011 report-164 --&gt;  &lt;vista&gt;     &lt;autenticacion&gt;true&lt;/autenticacion&gt;      &lt;excepcion&gt;gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector&lt;/excepcion&gt;         &lt;excepcion&gt;gov.sir.core.web.acciones.excepciones.ValidacionParametrosException&lt;/excepcion&gt;         &lt;nombre&gt;admin.reportes.report-164.param-form.view&lt;/nombre&gt;         &lt;parametro&gt;&lt;lave&gt;error&lt;/lave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/ERROR_MULTIPLE.jsp&lt;/valor&gt;&lt;/parametro&gt;         &lt;parametro&gt;&lt;lave&gt;head&lt;/lave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/HEADER_LOGOUT.jsp&lt;/valor&gt;&lt;/parametro&gt;         &lt;parametro&gt;&lt;lave&gt;body&lt;/lave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/administracion/ADMINREPORTES_REPORT164_PARAMFORM.jsp&lt;/valor&gt;&lt;/parametro&gt;         &lt;parametro&gt;&lt;lave&gt;bottom&lt;/lave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/FOOTER_DEFAULT.jsp&lt;/valor&gt;&lt;/parametro&gt;         &lt;/vista&gt;/FOOTER_DEFAULT.jsp&lt;/valor&gt;&lt;/parametro&gt;     &lt;/vista&gt;</pre> <p style="text-align: center;"><b>SIR-APP\src\org.auriga.smart.web.xml</b></p> <pre>&lt;!-- AHERRENTO 23/02/2011 report-164 --&gt; &lt;navegacion&gt;     &lt;resultado&gt;REPORTE_164_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK&lt;/resultado&gt;     &lt;url-redireccion&gt;admin.reportes.report-164.param-form.view&lt;/url-redireccion&gt; &lt;/navegacion&gt;</pre> <p style="text-align: center;"><b>SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES_REPORT164_PARAMFORM</b></p>		
Archivo Nuevo.		
<b>SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java</b>		
<pre>// AHERRENTO 23/02/2011 // Reporte 164 - public static final String REPORTE_164_SEND_ACTION = "REPORTE_164_SEND_ACTION"; public static final String REPORTE_164_BACK_ACTION = "REPORTE_164_BACK_ACTION";  // AHERRENTO 23/02/2011 // Reporte 164 public static final String REPORTE_164_LOADCIRCULOSBYUSUARIO = "REPORTE_164_LOADCIRCULOSBYUSUARIO";  // AHERRENTO 23/02/2011 // Declaración de las Variables para el reporte 164 public static final String REPORTE_164_PARAM_CMDKEY      = "reporte-164.jasper"; public static final String REPORTE_164_PARAM_PFECHAINI   = "V_FECHAINI"; public static final String REPORTE_164_PARAM_PFECHAFIN   = "V_FECHAFIN"; public static final String REPORTE_164_PARAM_PCIRCULONOMBRE = "V_CIRCULO"; public static final String REPORTE_164_PARAM_PCUANTIA     = "V_CUANTIA"; public static final String REPORTE_164_PARAM_USUARIOLOG  = "P_USUARIO_LOG";  //AHERRENTO 23/02/2011 //Reporte 164 if (REPORTE_164_LOADCIRCULOSBYUSUARIO.equals(accion)) {     return do_ReporteXX_LoadCirculosByUsuario_Action(request); }  // AHERRENTO 23/02/2011</pre>		

```

// Reporte 164
if (REPORTE_164_SEND_ACTION.equals(accion)) {
    do_Reporte164_Send_Action(request);
}
if (REPORTE_164_BACK_ACTION.equals(accion)) {
    do_Reporte164_Back_Action(request);
}

//AHERRENO 05/01/2011
//Reporte 162
private void do_Reporte164_Send_Action(HttpServletRequest request)
    throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_164_PARAM_PFECHAINI);
    String param_02 = request.getParameter(REPORTE_164_PARAM_PFECHAFIN);
    String param_03 = request.getParameter(REPORTE_164_PARAM_PCIRCULONOMBRE);
    String param_04 = request.getParameter(REPORTE_164_PARAM_USUARIOLOG);
    String param_05 = request.getParameter(REPORTE_164_PARAM_PCUANTIA);
    // apply validators
    if ((null == param_01) || ("").equalsIgnoreCase(param_01))) {
        exception.addError("parametro: Fecha Inicio Traslado: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio Solicitud: La fecha es inválida" + e.getMessage());
        }
    }
    // apply validators
    if ((null == param_02) || ("").equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Fecha Fin Solicitud: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_02);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_02 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Fin Solicitud: La fecha es inválida" + e.getMessage());
        }
    }
    // apply validators
    if ((null == param_03) || ("").equalsIgnoreCase(param_03))) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
            exception.addError("parametro: Circulo: Debe digitar un valor");
        }
    }
    // apply validators
    if ((null == param_04) || ("").equalsIgnoreCase(param_04))) {
        exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
            exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
        }
    }
}

```

```

        }

    // apply validators
    if ((null == param_05) || ("".equalsIgnoreCase(param_05))) {
        exception.addError("parametro: Cuantia: Debe digitar un valor");

    } else {
        boolean bandera = true;
        for (int i = 0; i < param_05.length(); i++) {
            if (!Character.isDigit(param_05.charAt(i))) {
                exception.addError("parametro: Cuantia: Debe ser un valor numerico");
                i = param_05.length();
                if(bandera){
                    bandera = false;
                }
            }
        }
        if(bandera){
            if(Double.valueOf(param_05.trim()).doubleValue() < Double.valueOf("7000000000").doubleValue()){
                exception.addError("parametro: Cuantia: Debe ser igual o mayor a 7.000.000.000");
            }
        }
    }

    String[] itemIds = new String[]{
        REPORTE_164_PARAM_PFECHAINI, REPORTE_164_PARAM_PFECHAFIN,
        REPORTE_164_PARAM_PCIRCULONOMBRE, REPORTE_164_PARAM_USUARIOLOG,
        REPORTE_164_PARAM_PCUANTIA
    };

    do_Reporte164_SaveState(request);

    // raise -----
    if (exception.getErroses().size() > 0) {
        throw exception;
    }
    // -----
    String[] params = new String[]{
        REPORTE_XX_PARAM_CMDKEY, REPORTE_164_PARAM_PFECHAINI,
        REPORTE_164_PARAM_PFECHAFIN, REPORTE_164_PARAM_PCIRCULONOMBRE,
        REPORTE_164_PARAM_USUARIOLOG, REPORTE_164_PARAM_PCUANTIA
    };

    String[] values = new String[]{
        REPORTE_164_PARAM_CMDKEY, param_01, param_02, param_03, param_04, param_05
    };

    String url = makeUrl(params, values);
    reportsServices_ConfigureParameterUri(request, url);

    do_ReporteXX_RemoveState(itemIds, request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

private void do_Reporte164_Back_Action(HttpServletRequest request) {
    do_Reporte164_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
} // :do_Reporte164_Back_Action

private void do_Reporte164_SaveState(HttpServletRequest request) {
    HttpSession session;
    String[] itemIds = new String[]{
        REPORTE_164_PARAM_PFECHAINI, REPORTE_164_PARAM_PFECHAFIN,
        REPORTE_164_PARAM_PCIRCULONOMBRE, REPORTE_164_PARAM_USUARIOLOG,
        REPORTE_164_PARAM_PCUANTIA
    };
}

```

```

        session = request.getSession();
        save_PageltemsState(itemIds, request, session);

    } // :do_Reporte164_SaveState

    private void do_Reporte164_RemoveState(HttpServletRequest request) {

        HttpSession session;

        String[] itemIds = new String[]{
            REPORTE_164_PARAM_PFECHAINI, REPORTE_164_PARAM_PFECHAFIN,
            REPORTE_164_PARAM_PCIRCULONOMBRE, REPORTE_164_PARAM_USUARIOLOG,
            REPORTE_164_PARAM_PCUANTIA
        };

        session = request.getSession();
        delete_PageltemsState(itemIds, request, session);

    } // :do_Reporte164_RemoveState

```

#### JASPER – reporte-164.jasper

Archivo Nuevo.

#### SIR-APP\src\gov\sisr\core\web\navegacion\administracion\ControlNavegacionReportes.java

```

//AHERRENO 23/02/2011
//Reporte 164
public static final String REPORTE_164_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK =
    "REPORTE_164_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK";

//AHERRENO 23/02/2011
//Reporte 164
listaReportesJasper.add(AWReportes.REPORTE_164_SEND_ACTION);

//AHERRENO 23/02/2011
// -- navegacion sobre pagina
if( AWReportes.REPORTE_164_LOADCIRCULOSBYUSUARIO.equals( accion ) ){
    return REPORTE_164_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK;
}

//AHERRENO 23/02/2011
//Reporte 164
if( AWReportes.REPORTE_164_BACK_ACTION.equals( accion ) ) {
    return REPORTE_XX_BACK_EVENTRESP_OK;
}

```

#### SIR-APP\src\gov\sisr\hermod\dao\impl\jdogenie\JDOResportesJasperDAO.java

```

//AHERRENO 23/02/2011
//Reporte 164
//Se obtienen los parámetros para enviar al procedimiento
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_164_PARAM_PFECHAINI);
listaParametros.add(AWReportes.REPORTE_164_PARAM_PFECHAFIN);
listaParametros.add(AWReportes.REPORTE_164_PARAM_PCIRCULONOMBRE);
listaParametros.add(AWReportes.REPORTE_164_PARAM_PCUANTIA);
mapaReportes.put(AWReportes.REPORTE_164_PARAM_CMDKEY, new Object[]{
    "SELECT seq_reporte_164.NEXTVAL FROM dual",
    "sp_reporte_164(?, ?, ?, ?, ?)", listaParametros,
    "DELETE FROM sir_op_reporte_164 WHERE sec_proceso = "));

```

#### mapas.jar

```

org.auriga.smart.web.xml
org.auriga.smart.vistas.xml

```

En conversación telefónica el día 23/02/2011 con la funcionaria Yamile Moreno se aclara que el parámetro Cuantía aceptara a partir de diez dígitos en adelante.

Ejemplo si se ingresa la cantidad 7.000.000.001 en la búsqueda debe traer todos los actos por valor de .7000.000.001 y todos los de ahí en adelante.

#### 220 REPORTE E01 -ESTADISTICAS POR INDICAD

#### **REQUERIMIENTO**

ESTADISTICAS POR INDICADORES DE REGISTRO E01

Se debe corregir el reporte E01, respecto a que se deben suprimir los siguientes ítems:  
Total de Certificados con mayor valor,  
Total de Certificados Antiguo Sistema,  
Total de Certificados de Pertenencia,  
Total de Certificados Exentos

Lo anterior, para que únicamente se refleje en este reporte el ítem Total Certificados de Tradición (Todos), en cuanto al proceso de certificados. Se anexa reporte generado de prueba.

### **DESARROLLO**

#### **Análisis de Requerimiento:**

Se modifica procedimiento SP\_REPORTE\_E01 en el cual se elimina las secciones de:

- Total de Certificados con mayor valor,
- Total de Certificados Antiguo Sistema,
- Total de Certificados de Pertenencia,
- Total de Certificados Exentos

### **221 Requerimiento Estadísticas Generales-2**

#### **REQUERIMIENTO**

#### **ESTADISTICAS GENERALES REPORTE EO2**

Se debe corregir el reporte E02 Estadísticas generales, en el primer bloque correspondiente a solicitudes de Certificados, toda vez que se realizo prueba en producción con el circulo de pruebas generando certificados de diferente tipo y tarifa, detectándose las siguientes inconsistencias:

No totaliza los certificados asociados a turnos de documento. Debe crearse el Ítem de CERTIFICADOS ASOCIADOS para que la información se refleje en la línea correspondiente En el ítem de CERTIFICADOS ANTIGUO SISTEMA, esta incluyendo tanto la cantidad como los valores, de los turnos de certificados de pertenencia, lo cual es erróneo, dado que esta información está incluida en el ítem de CERTIFICADOS PERTENENCIA.

Se anexan los reportes E02-Estadísticas Generales, Diario Radicador de certificados y E01-Estadísticas por indicadores de registro, radicando un total de 15 certificados

Así mismo, al finalizar la columna de valores en el bloque de Certificados por tarifas, se debe agregar la doble línea como en los otros bloques y seguido, en la línea gris incluir el valor de la sumatoria de valores de este bloque., es decir esta sumatoria debe ser igual a los valores del total del primer bloque de certificados.

Requerimiento enviado por **correo el día 13/09/2010** por la funcionaria Marleny Alaguna. en el requerimiento Estadísticas Generales hacen la siguiente observación:

Se anexan los reportes E02-Estadísticas Generales, Diario Radicador de certificados y E01-Estadísticas por indicadores de registro, radicando un total de 15 certificados

En el correo no se anexaron los reportes.

### **DESARROLLO**

Se modifica procedimiento SP\_REPORTE\_E02

1. Se crea Ítem de CERTIFICADOS ASOCIADOS.
2. En el ítem de CERTIFICADOS ANTIGUO SISTEMA, se agrega condición tipo\_certificado = 3.
3. se divide el bloque de tarifas y se agrega su total.

**227 Error en las relaciones de turnos  
Informativo**

**REQUERIMIENTO**

Requerimiento Glpi No 21864

La usuaria Sandra Ramirez nos informa que se generó el reparto el día de hoy 21 de Septiembre y se generó la relación 2010-070-1268, al verificar dicha relación los turnos no fueron asignados en el orden que aparecen pues al usuario Jarby le aparecen todos los turnos tanto los asignados a el que fueron 4 más los asignados a las usuarias Carmen duarte y Flor Vargas.

Se generó el reparto el día de hoy 21 de Septiembre y se generó la relación 2010-070- 1268, al verificar dicha relación los turnos no fueron asignados en el orden que aparecen pues al usuario Jarby le aparecen todos los turnos tanto los asignados a el que fueron 4 2010-070-6-12554, 2010-070-6-12558 2010-070-6-12559 y 2010-070-6-12564 más los asignados a las usuarias Carmen duarte y Flor Vargas, al ingresar al usuario de Jarby Puentes le aparecen todos los turnos y es necesario reasignar los demás a las usuarias correspondientes, se requiere verificar que pudo haber pasado en el momento de hacer el reparto registro.

Grupo Tunja

**DESARROLLO**

Descripción técnica:

Se verifica el proceso a nivel de datos y se encuentra que se ejecutó doble vez y en cada proceso de reparto se asignaron los turnos a cada usuario involucrado.

Se toma una muestra de los turnos y se verifica en la historia de éstos la asignación de la relación que se reporta en el incidente y está bien.

Para los usuarios calificadores se revisa la información de atención subtipo y están debidamente definidos los subtipos para cada uno de los usuarios (CARMEN.DUARTE (ID=1039), FLOR VARGAS (ID=1044), JARBY.PUENTES (ID=53296))

Se anexan, en una hoja excel, los id de los turnos asignados de cada proceso de reparto por usuario.

Descripción solución:

Se abre un caso general para crear un escenario donde se pueda reproducir el error y trabajar su solución. La solución puntual de este incidente es reasignar los turnos en la ORIP.

Se hicieron las respectivas pruebas al caso y no se replica el error, si se vuelve a presentar se abrirá de nuevo el acta para hacer seguimiento de la inconsistencia.

## 229 Error en correcciones Informativo

### **REQUERIMIENTO**

Requerimiento Glpi No 22133

Turno:2010-350-3-1659

Matricula(s):350-120641

Motivo Inconsistencia: La funcionaria Venovit Mejía Alarcón informa que la funcionaria ELICENIA.GUZMÁN que al verificar la salvedad del turno 2010-350-3-1659 en la columna de usuario aparece la usuaria LESVY.MUNOZ y si se verifica en el historial y la corrección la realizó la usuaria ELICENIA.GUZMÁN.

Se realizaron pruebas con el turno 2010-999-3-394 y el sistema toma los datos de quien avanza el turno de revisar y aprobar, como si dicho usuario hubiera realizado la corrección, por favor verificar.

Grupo Ibagué

### **DESARROLLO**

Se modificaron los siguientes archivos en SIR:

WEB-INF\jspSeguro\administracion\CONSULTA\_FOLIO\_SALVEDADES.jsp.  
gov.sir.impl.jdo.

#### **ARCHIVO JSP:**

**WEB-INF\jspSeguro\administracion\CONSULTA\_FOLIO\_SALVEDADES.jsp.**

#### **Código Modificado Línea 81:**

```
if( salvedad.getUsuarioCreacionTMP()!=null&&salvedad.getUsuarioCreacionTMP().getUsername()!=null){
```

#### **Código Modificado Línea 84 - 86:**

```
if(usr == null &&
salvedad.getUsuarioCreacion()!=null&&salvedad.getUsuarioCreacion().getUsername()!=null){
    usr = salvedad.getUsuarioCreacion().getUsername();
}
```

#### **Código Modificado Línea 221:**

```
if( salvedad.getUsuarioCreacionTMP()!=null&&salvedad.getUsuarioCreacionTMP().getUsername()!=null){
```

#### **Código Modificado Línea 224 - 226:**

```
if(usr == null &&
salvedad.getUsuarioCreacion()!=null&&salvedad.getUsuarioCreacion().getUsername()!=null){
    usr = salvedad.getUsuarioCreacion().getUsername();
}
```

#### **ARCHIVO JDO:**

**gov.sir.impl.jdo.**

#### **Código Modificado Línea 875 - 880:**

```
<!--
    Se cambia el valor de la columna ID_USUARIO por ID_USUARIO_TMP
    Caso mantis : 006332.
    Autor : Ellery Robles G.
-->
<extension vendor-name="versant" key="jdbc-column-name" value="ID_USUARIO_TMP"
/>
```

## 230 Reimpresion de Certificados - Cambio Matricula

## Informativo

### REQUERIMIENTO

Octubre 5 de 2010

Según conversación telefónica con la Doctora María Clemencia Rangel Directora de Registro -E- se definió que para la reimpresión de certificados, para el caso cambio de matrícula, se requiere que el sistema valide que la fecha límite para realizar el cambio de matrícula y reimpresión es de 5 días hábiles a partir de la fecha de radicación de la solicitud del certificado, si es de tipo inmediato.

Favor tener en cuenta lo expuesto en el desarrollo del requerimiento en mención y dar prioridad a esta solicitud, toda vez que existe hallazgo de la controlaría y de Control Interno de la SNR.

### DESARROLLO

Este caso se trabaja con los requerimientos 012 y 236.

## 234 Error Pantallas Administrativas - Reabrir Folios Informativo

### REQUERIMIENTO

Pantallas administrativas – Folios Anotaciones y Actos – Reabrir Folios

En ambiente de calidad, la matricula 300-667 se encuentra cerrada y por la opción de pantallas administrativas – Reabrir Folios se reabre. Esta matricula se toma para el proceso de correcciones con el turno 2010-300-3-942 se avanza al rol responsable Correcciones (Análisis) y al enviar a Antiguo Sistema genera mensaje “Existen cambios temporales creados con el turno”



### DESARROLLO

Se hace revisión en base de datos de las tablas temporales con los siguientes scripts:

```
select * from sir_ne_anota_ciudadano_tmp where id_matricula = '300-667'  
select * from sir_ne_anotacion_tmp where id_matricula = '300-667'  
select * from sir_ne_direccion_tmp where id_matricula = '300-667'  
select * from sir_ne_folio_datos_tmp where id_matricula = '300-667'  
select * from sir_ne_folio_derivado_tmp where id_matricula = '300-667'  
select * from sir_ne_rel_cancelacion_tmp where id_matricula = '300-667'  
select * from sir_ne_salvedad_an_tmp where id_matricula = '300-667'  
select * from sir_ne_salvedad_fo_tmp where id_matricula = '300-667'
```

NOTA: Se detectaron datos temporales en la tabla sir\_ne\_folio\_datos\_tmp

Se hace referencia de esta incidencia para su respectivo trámite. ref Glpi 23387.

<b>CLASE:</b>
<b>gov.sir.forseti.dao.impl.jdogenie.JDOGenieFolioDAO.java</b>
<b>Método:</b>
<b>protected void hacerDefinitivoFolioDatos(FolioEnhanced folio, FolioDatosTMP datos, PersistenceManager pm, TurnoEnhanced turno, boolean validarTurno, UsuarioEnhanced usuario) throws DAOException {</b>

**Código Insertado Línea 15706:**

```
/**  
 * @author : Julio Alcazar  
 * @change : Se permite eliminar temporales en el caso de que se modifique los datos sin utilizar  
folio.  
 * Caso Mantis : 0008036: Acta - Requerimiento No 234 - Error Pantallas Administrativas - Reabrir  
Folios  
 */  
if(set.size()==1 && set.contains(null)) {  
    pm.deletePersistent(datos);  
} else {
```

**235 Reporte 25\_Modificación llamado reporte  
Informativo**

**REQUERIMIENTO**

Reporte 25 – Informe recaudos del día(Tiracaja)

Actualmente el jasper se encarga de ejecutar la consulta a varias tablas lo cual genera lentitud en la generación del reporte. La metodología implementada para los reportes es ejecutar un procedimiento y almacenar en una tabla, esta tabla será la que se llame del jasper.

**DESARROLLO**

Se da solución a este caso con el requerimiento 158.

**238 Problemas Consulta índice de propietario  
Informativo**

**REQUERIMIENTO**

Requerimiento GLPI No 24100

Matrícula(s):017-43439

Motivo Inconsistencia : Al realizar la consulta por índice de propietario no arroja ningún resultado

Por favor verificar, ya que se realizaron pruebas y al consultar por el nombre o por cédula no arroja ningún resultado

Grupo: La ceja

**DESARROLLO**

Se revisó la información en la base de datos y el usuario aparece como propietario en el folio referenciado en la anotación 1.

Al consultar el detalle del folio 017-43439 se verifica que en la notación 1 si se encuentra el señor RUIZ BEDOYA RAMON ANTONIO, pero con el rol "DE" y no "A", por lo

tanto, al momento de realizar la consulta por índice de propietario esta anotación nunca será tenida en cuenta.

### 239 Informacion invalida en consulta por indice de propietario Informativa

#### **REQUERIMIENTO**

Requerimiento GLPI No 24361

Motivo Inconsistencia: Al realizar una consulta el sistema trae información de una anotación en estado INVÁLIDA.

Al realizar una consulta por índice de propietario del ciudadano PAREJA GARCIA ALEXANDER CC 10138009 de la ciudad de Pereira, el sistema lo muestra como propietario en la anotación 3 del folio 290-148078, al verificar dicho folio en el sistema se observa que la anotación 3 se encuentra en estado INVALIDA, se requiere verificar el por qué el sistema no valida la información con respecto a las anotación invalidas

Grupo Pereira

#### **DESARROLLO**

El aplicativo muestra las anotaciones invalidas donde el ciudadano consultado aparece como propietario.

Se debe corregir está consulta en el aplicativo para lo cual se solicita abrir un ACTA para desarrollo.

El caso puntual la mesa de ayuda lo trabajará con el usuario para que en la anotación inválida modifiquen el propietario con un turno de correcciones.

Se modificaron los siguientes archivos en SIR:

sir.forseti.dao.impl.jdogenie. JDOGenieFolioDAO.java.

#### **ARCHIVO JAVA:**

**sir.forseti.dao.impl.jdogenie. JDOGenieFolioDAO.java**

#### **Código Modificado Línea 7448 - 7451:**

```
/*
 * @author : Ellery David Robles Gómez
 * @change : Se filtran las anotaciones que estén en estado INVALIDA, para no obtenerlas.
 * Caso Mantis : 07228
```

#### **Código Insertado Línea 7452:**

```
filtro.append(" && (anotacion.estado.idEstadoAn!="" + CEstadoAnotacion.INVALIDO + "")");
```

### 242 Error Proceso Certificados al agregar Nota Informativa Informativo

#### **REQUERIMIENTO**

Certificados, se radica turno 2010-999-1- con matricula 999-963, se agrega Nota Informativa - NOTA INFORMATIVA SOLICITUD DE CERTIFICADOS – OK, al dar liquidar genera el siguiente error: “RuntimeException; CausedByException is: String index out of range: -1 : RuntimeException; CausedByException is: String index out of range: -1”



**DESARROLLO**

Se modificaron los siguientes archivos en SIR:

WEB-INF.jspSeguro.certificado. SOLICITUD\_CERTIFICADO.jsp.  
 sir.core.negocio.acciones.comun. ANPago.java.

<b>ARCHIVO JSP:</b>
WEB-INF.jspSeguro.certificado. SOLICITUD_CERTIFICADO.jsp

**Código Modificado Línea 955:**

```
<td><input name="ID_VALOR_SECUENCIAL" id="ID_VALOR_SECUENCIAL" type="text"
value="<%valorSecuencial%>" onFocus="campoactual('ID_VALOR_SECUENCIAL');"
class="camposformtext">
```

<b>ARCHIVO JAVA:</b>
sir.core.negocio.acciones.comun. ANPago.java.

**Código Eliminado Línea 352 - 371**

**Código Inserado Línea 658 - 671**

```
List listaNotasInformativas = evento.getListaNotasSinTurno();
if (listaNotasInformativas != null) {
    turno = pago.getLiquidacion().getSolicitud().getTurno();
    TurnoPk idTurno = new TurnoPk();
    idTurno.anio = turno.getAnio();
    idTurno.idCirculo = turno.getIdCirculo();
    idTurno.idTurno = turno.getIdTurno();
    idTurno.idProceso = turno.getIdProceso();
    for (int i = 0; i < listaNotasInformativas.size(); i++) {
        Nota notaTurno = (Nota) listaNotasInformativas.get(i);
        hermod.addNotaToTurno(notaTurno, idTurno);
    }
}
```

<b>243 Generación de notas devolutivas</b>
--

**REQUERIMIENTO**

Copias a expedir notas devolutivas.

Se solicita hacer los reajustes en la expedición de notas devolutiva, para que se generen 4 de acuerdo al comunicado SNR2011E000618, que se anexa a continuación;

Referencia: SNR2010IE021412

En atención al Oficio en referencia me permito informar que se requieren cuatro copias de la nota devolutiva de registro, una de estas copias se debe anexar al documento que se devuelve al usuario, otra acompaña a la copia del documento devuelto y que normalmente se conserva en archivo de la oficina, dos para notificar al ciudadano y que deben conservarse en carpeta para el archivo de la oficina, una reposara en dicha carpeta y otra hará parte del expediente en el caso que el usuario interponer recurso, además cuando se trata de un embargo, éste es remitido directamente al juzgado de origen, el usuario solicita a la oficina información sobre el motivo de devolución se debe suministrar una copia.

Cordialmente,

  
ZAIRAIDA GELIS CARRILLO

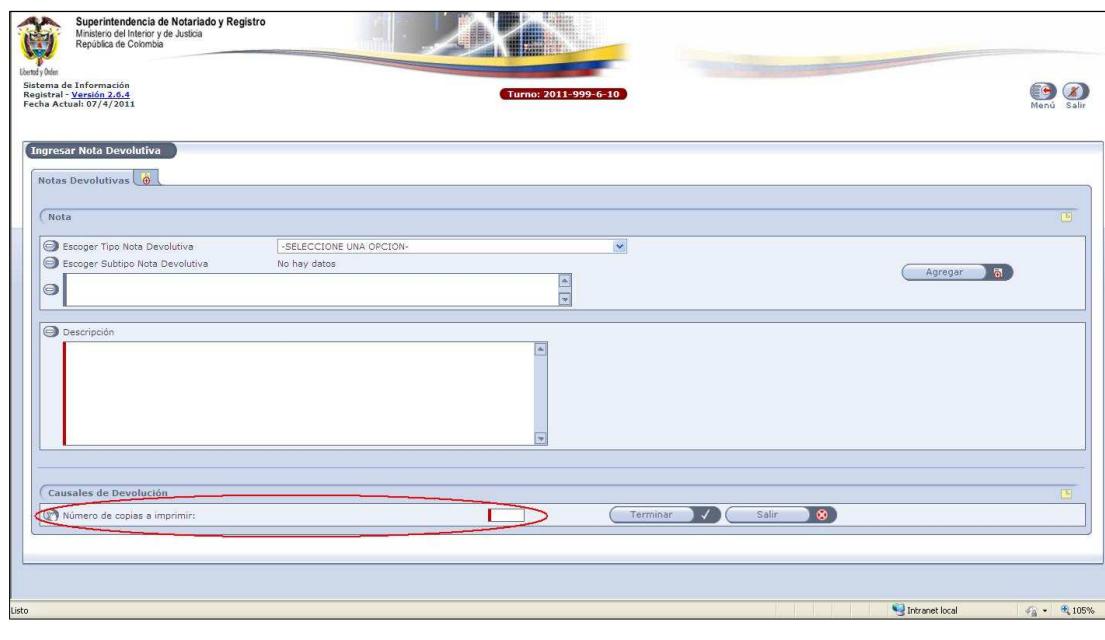
Directora de Registro

Proyecto: Isaiel Quirós P. 



## DESARROLLO

Actualmente SIR cumple con este requerimiento, en la interfaz de Notas Devolutivas existe un campo en donde se puede especificar el número de copias que se desean imprimir, se adjunta imagen en donde se indica de forma clara el campo en donde se ingresa el número de copias.



The screenshot shows the 'Ingresar Nota Devolutiva' (Enter Devolution Note) screen. At the top, there's a header with the logo of the Superintendencia de Notariado y Registro, the date (07/4/2011), and a turn number (2011-999-6-10). Below the header, there are sections for 'Nota' (Note) and 'Descripción' (Description). The 'Causales de Devolución' (Reasons for Devolution) section is circled in red. Within this section, there is a field labeled 'Número de copias a imprimir:' (Number of copies to print:) which has a red arrow pointing to it from the left side of the image.

Correo 08/04/2011 - Yamile Moreno

En el requerimiento de impresión número de copias -Notas Devolutivas - se requiere que en el campo del número de copias el aplicativo debe mostrar por defecto el numero 4- esta información no debe ser editable .

Script Bases de Datos:

Ninguno.

Creacion De Imagenes:

Ninguno.

Se crearon clases nuevas en SIR:

Ninguno.

Se modificaron los siguientes archivos en SIR:

WEB-INF.jspSeguro.comun. NOTAS\_DEVOLUTIVAS.jsp.

**ARCHIVO JSP:**

WEB-INF.jspSeguro.comun. NOTAS\_DEVOLUTIVAS.jsp.

**Código Modificado Línea 431:**

```
session.setAttribute(WebKeys.NUMERO_COPIAS_IMPRESION, "4");
```

**Código Modificado Línea 435:**

```
textHelper.setCssClase("campositem");
```

**Código Insertado Línea 435:**

```
textHelper.setReadonly(true);
```

**244 Error del turno en fase de certificados asociados  
Informativo**

**REQUERIMIENTO - GLPI No 25593**

Turno: 2011-170-6-143 - Matricula(s):170-2747,170-16679,170-34591.

Motivo Inconsistencia: Al intentar tomar el turno en la fase CERTIFICADOS ASOCIADOS el sistema genera error.

Se realizan pruebas en Mesa de Ayuda y se observa que el error se presenta con turnos Inscritos parcialmente con certificados asociados, favor verificar el inconveniente ya que en la oficina se requiere la impresión del certificado y el avance del turno, se adjunta imagen del error y excepción.

**DESARROLLO**

Se trata de acceder a una lista sin verificar la existencia de datos en ésta.

**Solución:** Se valida la lista en el código.

Clase: gov.sir.core.negocio.acciones.registro.ANMesaControl

línea 1121:

```
/**
```

```
* @author: Fernando Padilla
```

```
* @change: Mantis 7595: Error en la impresión, se generaba
```

error al tratar de

```
* de acceder a solT.getSolicitudFolios().get(0).
```

```

*/
if(solT.getSolicitudFolios()!= null &&
solT.getSolicitudFolios().size() > 0){
    SolicitudFolio solFolCert =
(SolicitudFolio)solT.getSolicitudFolios().get(0);

```

## 245 RolConsultasEspeciales

### **REQUERIMIENTO**

Con base en la solicitud OAP-022 de enero 27 de 2011, y la comunicación SNR2010EE003147 de febrero 3 de 2011, se transcribe el requerimiento de Nuevo Desarrollo para el aplicativo SIR:

Atendiendo instrucciones del Señor Superintendente, se solicita la creación de un usuario del aplicativo SIR para uso de la Oficina Asesora de Planeación con el cual se pueda consultar la siguiente información:

Numero de Documentos radicados mensualmente por cada Oficina de Registro

Numero de Documentos repartidos mensualmente por abogado calificador

Documentos pendientes por calificar por cada Abogado Calificador

Numero de reimpresiones de Certificados de Libertad por Oficina y por quien es autorizadas

### **DESARROLLO**

Dado que en el aplicativo SIR, la funcionalidad no está guiada por el usuario, se recomienda la creación de un nuevo rol que se denomine CONSULTAS\_ESTADISTICAS, el cual podrá ser asignado al usuario, o usuarios, que la SNR defina.

Cada una de las consultas se resuelve como un reporte, con los parámetros de fecha inicio y fecha fin, para establecer el rango de información a listar. Se recomienda colocar una restricción de no más de 6 meses en el rango de consulta debido al peso que el reporte tiene en su ejecución. Estas consultas cubren a la totalidad de las 91 ORIPs.

En la especificación del requerimiento NO es clara la periodicidad que deben tener los reportes, exceptuando el primero de ellos, por lo que se asume que todos corresponden a barridos mensuales de la información. Por esta razón se especifica el rango de tiempo con los parámetros Fecha\_Inicio y Fecha\_Fin.

El rol CONSULTAS\_ESTADISTICAS presentara un formulario de interface con el acceso a los cuatro reportes que se solicitan. Como se trata de consultas que afectan varias tablas del nivel transaccional, se debe desarrollar un paquete de código PL/SQL que trate la consulta, la almacene en espacio temporal, y finalmente la materialice en un reporte de iReport. Se reciben los parámetros de Fecha\_Inicio y Fecha\_Fin, y una lista de ORIPs en la que se incluye la opción Todas para generación del reporte completo (91 ORIPs).

Área de desarrollo involucrada: Base de Datos, Programación PL/SQL, y Capa de Presentación.

Se diseñan reportes:

- Reporte-165.jasper

- Reporte-166.jasper
- Reporte-167.jasper
- Reporte-168.jasper

**CREACION DEL ROL CONSULTAS ESPECIALES AL CUAL SE LE ASIGNARAN LOS REPORTE 165, 166, 167, 168**

```
INSERT INTO SIR_OP_ROL (ID_ROL, ROLL_NOMBRE, ROLL_DESCRIPCION) VALUES ('SIR_ROL_CONSULTAS_ESPECIALES', 'CONSULTAS ESPECIALES', 'SIR_ROL_CONSULTAS_ESPECIALES')
```

#### **CREACION DE LA TABLA PARA EL REPORTE\_165**

```
CREATE TABLE "SIR"."SIR_OP_REPORTE_165"
(
    "CIRCULO" VARCHAR2(100 BYTE),
    "TOTAL" NUMBER,
    "MES" DATE,
    "SEC_PROCESO" NUMBER
) PCTFREE 10 PCTUSED 40 INITTRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "TBS_OWB_RUNTIME" ;

CREATE INDEX "SIR"."IDX REP_165 SEC" ON "SIR"."SIR_OP_REPORTE_165" ("SEC_PROCESO")
PCTFREE 10 INITTRANS 2 MAXTRANS 255
STORAGE(INITIAL 16384 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "TBS_SIR_D_1"
PARALLEL 16 ;
```

#### **CREACIÓN DE LA SECUENCIA PARA EL SP\_REPORTE\_165**

```
CREATE SEQUENCE SIR.SEQ_Reporte_165
MINVALUE 1 MAXVALUE 99999999999999999999999999999999
INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE;
```

#### **CONFIGURACION PANTALLA "REPORTE 165: DOCUMENTOS RADICADOS MENSUALMENTE"**

```
INSERT INTO SIR.SIR_OP_PANTALLA_ADMINISTRATIVA
(ID_PANTALLA_ADMINISTRATIVA, id_tipo_pantalla, ptdm_nombre, ptdm_vista, ptdm_pagina)
VALUES
(1121, 13, 'Reporte 165: Documentos Radicados Mensualmente', 'admin.reportes.report-165.param-form.view', 'PG_REPORTES')
```

#### **ASIGNACION DE LA PANTALLA "REPORTE 165 - DOCUMENTOS RADICADOS MENSUALMENTE" AL ROL SIR\_ROL\_CONSULTAS\_ESPECIALES**

```
INSERT INTO SIR.SIR_OP_ROL_PANTALLA
(ID_ROL, id_pantalla_administrativa)
VALUES
('SIR_ROL_CONSULTAS_ESPECIALES', 1121)
```

#### **CREACION DE LA TABLA PARA EL REPORTE\_166**

```
CREATE TABLE "SIR"."SIR_OP_REPORTE_166"
(
    "CIRCULO" VARCHAR2(100 BYTE),
    "TOTAL" NUMBER,
    "USUARIO" VARCHAR2(100 BYTE),
    "MES" DATE,
    "SEC_PROCESO" NUMBER
) PCTFREE 10 PCTUSED 40 INITTRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "TBS_OWB_RUNTIME" ;
```

```
CREATE INDEX "SIR"."IDX REP_166 SEC" ON "SIR"."SIR_OP_REPORTE_166" ("SEC_PROCESO")
PCTFREE 10 INITTRANS 2 MAXTRANS 255
STORAGE(INITIAL 16384 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "TBS_SIR_D_1"
```

PARALLEL 16 ;

#### CREACIÓN DE LA SECUENCIA PARA EL SP\_Reporte\_166

```
CREATE SEQUENCE SIR.SEQ_Reporte_166  
    MINVALUE 1 MAXVALUE 99999999999999999999999999999999  
    INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE;
```

#### CONFIGURACION PANTALLA "Reporte 166: Documentos Repartidos Mensualmente"

```
INSERT INTO SIR.SIR_OP_PANTALLA_ADMINISTRATIVA  
(ID_PANTALLA_ADMINISTRATIVA, id_tipo_pantalla, ptdm_nombre, ptdm_vista, ptdm_pagina)  
VALUES  
(1122, 13,'Reporte 166: Documentos Repartidos Mensualmente','admin.reportes.report-166.param-form.view','PG_REPORTES')
```

#### ASIGNACION DE LA PANTALLA "Reporte 166 - Documentos Repartidos Mensualmente" AL ROL SIR\_ROL\_CONSULTAS\_ESPECIALES

```
INSERT INTO SIR.SIR_OP_ROL_PANTALLA  
(ID_ROL, id_pantalla_administrativa)  
VALUES  
(SIR_ROL_CONSULTAS_ESPECIALES', 1122)
```

#### CREACION DE LA TABLA PARA EL Reporte\_167

```
CREATE TABLE "SIR"."SIR_OP_Reporte_167"  
(  
    "CIRCULO" VARCHAR2(100 BYTE),  
    "TOTAL" NUMBER,  
    "USUARIO" VARCHAR2(100 BYTE),  
    "MES" DATE,  
    "SEC_PROCESO" NUMBER  
) PCTFREE 10 PCTUSED 40 INITTRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING  
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645  
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT  
TABLESPACE "TBS_OWB_RUNTIME";
```

```
CREATE INDEX "SIR"."IDX REP_167 SEC" ON "SIR"."SIR_OP_Reporte_167" ("SEC_PROCESO")  
PCTFREE 10 INITTRANS 2 MAXTRANS 255  
STORAGE(INITIAL 16384 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645  
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT  
TABLESPACE "TBS_SIR_D_1"  
PARALLEL 16 ;
```

#### CREACIÓN DE LA SECUENCIA PARA EL SP\_Reporte\_167

```
CREATE SEQUENCE SIR.SEQ_Reporte_167  
    MINVALUE 1 MAXVALUE 99999999999999999999999999999999  
    INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE;
```

#### CONFIGURACION PANTALLA "Reporte 167: Documentos Pendientes por Calificar por cada Abogado"

```
INSERT INTO SIR.SIR_OP_PANTALLA_ADMINISTRATIVA  
(ID_PANTALLA_ADMINISTRATIVA, id_tipo_pantalla, ptdm_nombre, ptdm_vista, ptdm_pagina)  
VALUES  
(1123, 13,'Reporte 167: Documentos Pendientes por Calificar por cada Abogado','admin.reportes.report-167.param-form.view','PG_REPORTES')
```

#### ASIGNACION DE LA PANTALLA "Reporte 167: Documentos Pendientes por Calificar por cada Abogado" AL ROL SIR\_ROL\_CONSULTAS\_ESPECIALES

```
INSERT INTO SIR.SIR_OP_ROL_PANTALLA  
(ID_ROL, id_pantalla_administrativa)  
VALUES  
(SIR_ROL_CONSULTAS_ESPECIALES', 1123)
```

#### CREACION DE LA TABLA PARA EL Reporte\_168

```

CREATE TABLE "SIR"."SIR_OP_Reporte_168"
(
    "CIRCULO" VARCHAR2(100 BYTE),
    "TURNO" VARCHAR2(100 BYTE),
    "MATRICULA" VARCHAR2(100 BYTE),
    "USUARIO" VARCHAR2(100 BYTE),
    "MES" DATE,
    "SEC_PROCESO" NUMBER
) PCTFREE 10 PCTUSED 40 INITTRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "TBS_OWB_RUNTIME" ;

CREATE INDEX "SIR"."IDX REP_168 SEC" ON "SIR"."SIR_OP_Reporte_168" ("SEC_PROCESO")
PCTFREE 10 INITTRANS 2 MAXTRANS 255
STORAGE(INITIAL 16384 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "TBS_SIR_D_1"
PARALLEL 16 ;

```

#### CREACIÓN DE LA SECUENCIA PARA EL SP\_Reporte\_168

```

CREATE SEQUENCE SIR.SEQ_Reporte_168
MINVALUE 1 MAXVALUE 99999999999999999999999999999999
INCREMENT BY 1 START WITH 1 CACHE 20 NOORDER NOCYCLE;

```

#### CONFIGURACION PANTALLA "Reporte 168: Reimpresiones de Certificados por Oficina"

```

INSERT INTO SIR.SIR_OP_PANTALLA_ADMINISTRATIVA
(ID_PANTALLA_ADMINISTRATIVA, id_tipo_pantalla, ptdm_nombre, ptdm_vista, ptdm_pagina)
VALUES
(1124, 13,'Reporte 168: Reimpresiones de Certificados por Oficina','admin.reportes.report-168.param-form.view','PG_REPORTES')

```

#### ASIGNACION DE LA PANTALLA "Reporte 168: Reimpresiones de Certificados por Oficina" AL ROL SIR\_ROL\_CONSULTAS\_ESPECIALES

```

INSERT INTO SIR.SIR_OP_ROL_PANTALLA
(ID_ROL, id_pantalla_administrativa)
VALUES
('SIR_ROL_CONSULTAS_ESPECIALES', 1124)

```

Se creó el procedimiento SP\_Reporte\_165

Se creó el procedimiento SP\_Reporte\_166

Se creó el procedimiento SP\_Reporte\_167

Se creó el procedimiento SP\_Reporte\_168

#### Configuración Reporte 165 – Documentos Radicados Mensualmente

SIR-APP\src\org.auriga.smart.vistas.xml

```

<!-- AHERRENO 14/03/2011
report-165 -->

    <vista>
        <autenticacion>true</autenticacion>

        <excepcion>gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector</e
xcepcion>
            <excepcion>gov.sir.core.web.acciones.excepciones.ValidacionParametrosException</excepcion>
            <nombre>admin.reportes.report-165.param-form.view</nombre>
            <parametro><llave>error</llave><valor>/WEB-
INF/jspSeguro/comun/ERROR_MULTIPLE.jsp</valor></parametro>
            <parametro><llave>head</llave><valor>/WEB-
INF/jspSeguro/comun/HEADER_LOGOUT.jsp</valor></parametro>
            <parametro><llave>body</llave><valor>/WEB-
INF/jspSeguro/administracion/ADMINREPORTES_REPORT165_PARAMFORM.jsp</valor></parametro>
            <parametro><llave>bottom</llave><valor>/WEB-
INF/jspSeguro/comun/FOOTER_DEFAULT.jsp</valor></parametro>
    </vista>

```

SIR-APP\src\org.auriga.smart.web.xml

```

<!-- AHERRENO 14/03/2011
report-165 -->

```

<navegacion>       <resultado>REPORTE_165_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK</resultado>       <url-redireccion>admin.reportes.report-165.param-form.view</url-redireccion>     </navegacion>	<b>SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES_REPORT165_PARAMFORM</b>
<b>Archivo Nuevo.</b>	
<b>SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java</b>	
<pre>// AHERRENO 14/03/2011 // Reporte 165 public static final String REPORTE_165_SEND_ACTION = "REPORTE_165_SEND_ACTION"; public static final String REPORTE_165_BACK_ACTION = "REPORTE_165_BACK_ACTION";  // AHERRENO 14/03/2011 // Reporte 165 public static final String REPORTE_165_LOADCIRCULOSBYUSUARIO = "REPORTE_165_LOADCIRCULOSBYUSUARIO";  // AHERRENO 14/03/2011 // Declaración de las Variables para el reporte 165 public static final String REPORTE_165_PARAM_CMDKEY      = "reporte-165.jasper"; public static final String REPORTE_165_PARAM_PFECHAINI   = "V_FECHAINI"; public static final String REPORTE_165_PARAM_PFECHAFIN   = "V_FECHAFIN"; public static final String REPORTE_165_PARAM_PCIRCULONOMBRE = "V_CIRCULO"; public static final String REPORTE_165_PARAM_USUARIOLOG  = "P_USUARIO_LOG";  //AHERRENO 14/03/2011 //Reporte 165 if (REPORTE_165_LOADCIRCULOSBYUSUARIO.equals(accion)) {   return do_ReporteXX_LoadCirculosByUsuario_Action(request); }  // AHERRENO 14/03/2011 // Reporte 165 if (REPORTE_165_SEND_ACTION.equals(accion)) {   do_Reporte165_Send_Action(request); } if (REPORTE_165_BACK_ACTION.equals(accion)) {   do_Reporte165_Back_Action(request); }  //AHERRENO 14/03/2011 //Reporte 165 private void do_Reporte165_Send_Action(HttpServletRequest request) throws ValidacionParametrosException {    ValidacionParametrosException exception =     new AdminReportes_InvalidParametersExceptionCollector();    String param_01 = request.getParameter(REPORTE_165_PARAM_PFECHAINI);   String param_02 = request.getParameter(REPORTE_165_PARAM_PFECHAFIN);   String param_03 = request.getParameter(REPORTE_165_PARAM_PCIRCULONOMBRE);   String param_04 = request.getParameter(REPORTE_165_PARAM_USUARIOLOG);    GregorianCalendar capturaFecha1 = null;   GregorianCalendar capturaFecha2 = null;   int ano1 = 0;   int ano2 = 0;   int mes1 = 0;   int mes2 = 0;   int dia1 = 0;   int dia2 = 0;    // apply validators   if ((null == param_01)    ("".equalsIgnoreCase(param_01))) {      exception.addError("parametro: Fecha Inicio Radicacion: Debe digitar un valor");    } else {     String send1 = null;     Date fecha = null;     try {       fecha = DateFormatUtil.parse(param_01);       send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);       param_01 = send1;     }   } }</pre>	

```

String[] fechas = param_01.split("-");//yyyy-MM-dd - 0-1-2
ano1 = Integer.parseInt(fechas[0]);
mes1 = Integer.parseInt(fechas[1]);
dia1 = Integer.parseInt(fechas[2]);
capturaFecha1 = new GregorianCalendar(ano1,mes1,dia1);

} catch (Exception e) {
    exception.addError("parametro: Fecha Inicio Radicacion: La fecha es inválida" + e.getMessage());
}
}

// apply validators
if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {

    exception.addError("parametro: Fecha Fin Radicacion: Debe digitar un valor");

} else {
    String send1 = null;

    Date fecha = null;
    try {
        fecha = DateFormatUtil.parse(param_02);
        send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
        param_02 = send1;

        String[] fechas = param_02.split("-");//yyyy-MM-dd - 0-1-2
        ano2 = Integer.parseInt(fechas[0]);
        mes2 = Integer.parseInt(fechas[1]);
        dia2 = Integer.parseInt(fechas[2]);
        capturaFecha2 = new GregorianCalendar(ano2,mes2-1,dia2);//Los meses van de 0 a 11

    } catch (Exception e) {
        exception.addError("parametro: Fecha Fin Radicacion: La fecha es inválida" + e.getMessage());
    }
}

if(capturaFecha1 != null && capturaFecha2 != null){
    if((mes1 == mes2 && ano1 == ano2)){

        Calendar valFec = Calendar.getInstance();
        valFec.set(ano1, mes1-1, dia1, 0, 0, 0); //Los meses van de 0 a 11
        valFec.add(Calendar.DATE, 180);

        if( capturaFecha2.after(valFec)){
            exception.addError("El Rango de fechas no puede exceder a seis (6) meses");
        }
    }
}

// apply validators
if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {

    exception.addError("parametro: Circulo: Debe digitar un valor");

} else {

    if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    }
}

// apply validators
if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
    exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
        exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
    }
}

String[] itemIds = new String[]{
    REPORTE_165_PARAM_PFECHAINI, REPORTE_165_PARAM_PFECHAFIN,
    REPORTE_165_PARAM_PCIRCLONOMBRE, REPORTE_165_PARAM_USUARIOLOG
};

```

```

do_Reporte165_SaveState(request);
// raise -----
if (exception.getErrores().size() > 0) {
    throw exception;
}
// -----
String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY, REPORTE_165_PARAM_PFECHAINI,
    REPORTE_165_PARAM_PFECHAFIN, REPORTE_165_PARAM_PCIRCULONOMBRE,
    REPORTE_165_PARAM_USUARIOLOG
};
String[] values = new String[]{
    REPORTE_165_PARAM_CMDKEY, param_01, param_02, param_03, param_04
};
String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_ReporteXX_RemoveState(itemIds, request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

private void do_Reporte165_Back_Action(HttpServletRequest request) {
    do_Reporte165_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
} // :do_Reporte165_Back_Action

private void do_Reporte165_SaveState(HttpServletRequest request) {
    HttpSession session;
    String[] itemIds = new String[]{
        REPORTE_165_PARAM_PFECHAINI, REPORTE_165_PARAM_PFECHAFIN,
        REPORTE_165_PARAM_PCIRCULONOMBRE, REPORTE_165_PARAM_USUARIOLOG
    };
    session = request.getSession();
    save_PageItemsState(itemIds, request, session);
} // :do_Reporte165_SaveState

private void do_Reporte165_RemoveState(HttpServletRequest request) {
    HttpSession session;
    String[] itemIds = new String[]{
        REPORTE_165_PARAM_PFECHAINI, REPORTE_165_PARAM_PFECHAFIN,
        REPORTE_165_PARAM_PCIRCULONOMBRE, REPORTE_165_PARAM_USUARIOLOG
    };
    session = request.getSession();
    delete_PageItemsState(itemIds, request, session);
} // :do_Reporte165_RemoveState

    if (accion.equals(REPORTE_165_LOADCIRCULOSBYUSUARIO) ||
accion.equals(REPORTE_166_LOADCIRCULOSBYUSUARIO) ||
accion.equals(REPORTE_167_LOADCIRCULOSBYUSUARIO) ||
accion.equals(REPORTE_168_LOADCIRCULOSBYUSUARIO)) {
        param_HelperCirculoListModel.add(new ElementoLista("TODOS", " TODOS"));
    }

```

#### JASPER – reporte-165.jasper

#### Archivo Nuevo.

**SIR-APP\src\gov\sir\core\web\navegacion\administracion\ControlNavegacionReportes.java**

```

//AHERRENO 14/03/2011
//Reporte 165
public static final String REPORTE_165_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK =
    "REPORTE_165_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK";

//AHERRENO 14/03/2011
//Reporte 165
listaReportesJasper.add(AWReportes.REPORTE_165_SEND_ACTION);

//AHERRENO 14/03/2011

```

```

// -- navegacion sobre pagina
if( AWReportes.REPORTE_165_LOADCIRCULOSBYUSUARIO.equals( accion ) ){
    return REPORTE_165_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK;
}

//AHERRENO 14/03/2011
//Reporte 165
if( AWReportes.REPORTE_165_BACK_ACTION.equals( accion ) ) {
    return REPORTE_XX_BACK_EVENTRESP_OK;
}

```

**SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDOResportesJasperDAO.java**

```

//AHERRENO 14/03/2011
//Reporte 165
//Se obtienen los parámetros para enviar al procedimiento
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_165_PARAM_PFECHAINI);
listaParametros.add(AWReportes.REPORTE_165_PARAM_PFECHAFIN);
listaParametros.add(AWReportes.REPORTE_165_PARAM_PCIRCULONOMBRE);
mapaReportes.put(AWReportes.REPORTE_165_PARAM_CMDKEY, new Object[]{
    "SELECT seq_reporte_165.NEXTVAL FROM dual",
    "sp_reporte_165(?, ?, ?, ?)", listaParametros,
    "DELETE FROM sir_op_reporte_165 WHERE sec_proceso = "});

```

## Configuración Reporte 166 – Documentos Repartidos Mensualmente

**SIR-APP\src\org.auriga.smart.vistas.xml**

```

<!-- AHERRENO 14/03/2011
report-166 -->

<vista>
    <autenticacion>true</autenticacion>

    <expcion>gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector</e
xpcion>
        <expcion>gov.sir.core.web.acciones.excepciones.ValidacionParametrosException</expcion>
        <nombre>admin.reportes.report-166.param-form.view</nombre>
        <parametro><llave>error</llave><valor>/WEB-
INF/jspSeguro/comun/ERROR_MULTIPLE.jsp</valor></parametro>
        <parametro><llave>head</llave><valor>/WEB-
INF/jspSeguro/comun/HEADER_LOGOUT.jsp</valor></parametro>
        <parametro><llave>body</llave><valor>/WEB-
INF/jspSeguro/administracion/ADMINREPORTES_REPORT166_PARAMFORM.jsp</valor></parametro>
        <parametro><llave>bottom</llave><valor>/WEB-
INF/jspSeguro/comun/FOOTER_DEFAULT.jsp</valor></parametro>
    </vista>

```

**SIR-APP\src\org.auriga.smart.web.xml**

```

<!-- AHERRENO 14/03/2011
report-166 -->
<navegacion>
    <resultado>REPORTE_166_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK</resultado>
    <url-redireccion>admin.reportes.report-166.param-form.view</url-redireccion>
</navegacion>

```

**SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES\_REPORT166\_PARAMFORM**

**Archivo Nuevo.**

**SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java**

```

// AHERRENO 14/03/2011
// Reporte 166
public static final String REPORTE_166_SEND_ACTION = "REPORTE_166_SEND_ACTION";
public static final String REPORTE_166_BACK_ACTION = "REPORTE_166_BACK_ACTION";

// AHERRENO 14/03/2011
// Reporte 166
public static final String REPORTE_166_LOADCIRCULOSBYUSUARIO =
"REPORTE_166_LOADCIRCULOSBYUSUARIO";

// AHERRENO 14/03/2011
// Declaración de las Variables para el reporte 166
public static final String REPORTE_166_PARAM_CMDKEY      = "reporte-166.jasper";
public static final String REPORTE_166_PARAM_PFECHAINI   = "V_FECHAINI";
public static final String REPORTE_166_PARAM_PFECHAFIN   = "V_FECHAFIN";
public static final String REPORTE_166_PARAM_PCIRCULONOMBRE = "V_CIRCULO";
public static final String REPORTE_166_PARAM_USUARIOLOG  = "P_USUARIO_LOG";

```

//AHERRENO 14/03/2011

```

//Reporte 166
if (REPORTE_166_LOADCIRCUOSBYUSUARIO.equals(accion)) {
    return do_ReporteXX_LoadCirculosByUsuario_Action(request);
}

// AHERRENO 14/03/2011
// Reporte 166
if (REPORTE_166_SEND_ACTION.equals(accion)) {
    do_Reporte166_Send_Action(request);
}
if (REPORTE_166_BACK_ACTION.equals(accion)) {
    do_Reporte166_Back_Action(request);
}

//AHERRENO 14/03/2011
//Reporte 166
private void do_Reporte166_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {
    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_166_PARAM_PFECHAINI);
    String param_02 = request.getParameter(REPORTE_166_PARAM_PFECHAFIN);
    String param_03 = request.getParameter(REPORTE_166_PARAM_PICRUCULONOMBRE);
    String param_04 = request.getParameter(REPORTE_166_PARAM_USUARIOLOG);

    GregorianCalendar capturaFecha1 = null;
    GregorianCalendar capturaFecha2 = null;
    int ano1 = 0;
    int ano2 = 0;
    int mes1 = 0;
    int mes2 = 0;
    int dia1 = 0;
    int dia2 = 0;

    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {
        exception.addError("parametro: Fecha Inicio Reparto: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;

            String[] fechas = param_01.split("-");//yyyy-MM-dd - 0-1-2
            ano1 = Integer.parseInt(fechas[0]);
            mes1 = Integer.parseInt(fechas[1]);
            dia1 = Integer.parseInt(fechas[2]);
            capturaFecha1 = new GregorianCalendar(ano1,mes1,dia1);

        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio Reparto: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Fecha Fin Reparto: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_02);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_02 = send1;
        }
    }
}

```

```

String[] fechas = param_02.split("-");//yyyy-MM-dd - 0-1-2
ano2 = Integer.parseInt(fechas[0]);
mes2 = Integer.parseInt(fechas[1]);
dia2 = Integer.parseInt(fechas[2]);
capturaFecha2 = new GregorianCalendar(ano2,mes2-1,dia2);//Los meses van de 0 a 11
} catch (Exception e) {
    exception.addError("parametro: Fecha Fin Reparto: La fecha es inválida" + e.getMessage());
}
}

if(capturaFecha1 != null && capturaFecha2 != null){
    if(!(mes1 == mes2 && ano1 == ano2)){

        Calendar valFec = Calendar.getInstance();
        valFec.set(ano1, mes1-1, dia1, 0, 0); //Los meses van de 0 a 11
        valFec.add(Calendar.DATE, 180);

        if( capturaFecha2.after(valFec)){
            exception.addError("El Rango de fechas no puede exceder a seis (6) meses");
        }
    }
}

// apply validators
if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {

    exception.addError("parametro: Circulo: Debe digitar un valor");

} else {

    if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    }
}

// apply validators
if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
    exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
        exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
    }
}
String[] itemIds = new String[]{
    REPORTE_166_PARAM_PFECHAINI, REPORTE_166_PARAM_PFECHAFIN,
    REPORTE_166_PARAM_PCIRULONOMBRE, REPORTE_166_PARAM_USUARIOLOG
};

do_Reporte166_SaveState(request);
// raise -----
if (exception.getErrores().size() > 0) {
    throw exception;
}
// ----

String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY, REPORTE_166_PARAM_PFECHAINI,
    REPORTE_166_PARAM_PFECHAFIN, REPORTE_166_PARAM_PCIRULONOMBRE,
    REPORTE_166_PARAM_USUARIOLOG
};

String[] values = new String[]{
    REPORTE_166_PARAM_CMDKEY, param_01, param_02, param_03, param_04
};

String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_ReporteXX_RemoveState(itemIds, request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

private void do_Reporte166_Back_Action(HttpServletRequest request) {
    do_Reporte166_RemoveState(request);
}

```

```

        do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
    } // :do_Reporte166_Back_Action

    private void do_Reporte166_SaveState(HttpServletRequest request) {
        HttpSession session;
        String[] itemIds = new String[]{};
        RÉPORTE_166_PARAM_PFECHAINI, RÉPORTE_166_PARAM_PFECHAFIN,
        RÉPORTE_166_PARAM_PCIRCULONOMBRE, RÉPORTE_166_PARAM_USUARIOLOG
    };

    session = request.getSession();
    save_PageItemsState(itemIds, request, session);

} // :do_Reporte166_SaveState

private void do_Reporte166_RemoveState(HttpServletRequest request) {

    HttpSession session;

    String[] itemIds = new String[]{};
    RÉPORTE_166_PARAM_PFECHAINI, RÉPORTE_166_PARAM_PFECHAFIN,
    RÉPORTE_166_PARAM_PCIRCULONOMBRE, RÉPORTE_166_PARAM_USUARIOLOG
};

    session = request.getSession();
    delete_PageItemsState(itemIds, request, session);

} // :do_Reporte166_RemoveState

```

#### JASPER – reporte-166.jasper

#### Archivo Nuevo.

##### SIR-APP\src\gov\sir\core\web\navegacion\administracion\ControlNavegacionReportes.java

```

//AHERRENO 14/03/2011
//Reporte 166
public static final String RÉPORTE_166_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK =
    "RÉPORTE_166_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK";

//AHERRENO 14/03/2011
//Reporte 166
listaReportesJasper.add(AWReportes.RÉPORTE_166_SEND_ACTION);

//AHERRENO 14/03/2011
// -- navegacion sobre pagina
if( AWReportes.RÉPORTE_166_LOADCIRCULOSBYUSUARIO.equals( accion ) ){
    return RÉPORTE_166_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK;
}

//AHERRENO 14/03/2011
//Reporte 166
if( AWReportes.RÉPORTE_166_BACK_ACTION.equals( accion ) ) {
    return RÉPORTE_XX_BACK_EVENTRESP_OK;
}

```

##### SIR-APP\src\gov\sir\hermod\dao\impl\jdomenie\JDOResportesJasperDAO.java

```

//AHERRENO 14/03/2011
//Reporte 166
//Se obtienen los parámetros para enviar al procedimiento
listaParametros = new ArrayList();
listaParametros.add(AWReportes.RÉPORTE_166_PARAM_PFECHAINI);
listaParametros.add(AWReportes.RÉPORTE_166_PARAM_PFECHAFIN);
listaParametros.add(AWReportes.RÉPORTE_166_PARAM_PCIRCULONOMBRE);
mapaReportes.put(AWReportes.RÉPORTE_166_PARAM_CMDKEY, new Object[]{
    "SELECT seq_reporte_166.NEXTVAL FROM dual",
    "sp_reporte_166(?, ?, ?, ?)", listaParametros,
    "DELETE FROM sir_op_reporte_166 WHERE sec_proceso = "});

```

#### Configuración Reporte 167 – Documentos Pendientes por Calificar por cada Abogado

##### SIR-APP\src\org.auriga.smart.vistas.xml

```

<!-- AHERRENO 14/03/2011
report-167 -->

<vista>
    <autenticacion>true</autenticacion>

```

```

<excepcion>gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector</e
xcepcion>
    <excepcion>gov.sir.core.web.acciones.excepciones.ValidacionParametrosException</excepcion>
        <nOMBRE>admin.reportes.report-167.param-form.view</nOMBRE>
        <parametro><llave>error</llave><valor>/WEB-
INF/jspSeguro/comun/ERROR_MULTIPLE.jsp</valor></parametro>
        <parametro><llave>head</llave><valor>/WEB-
INF/jspSeguro/comun/HEADER_LOGOUT.jsp</valor></parametro>
        <parametro><llave>body</llave><valor>/WEB-
INF/jspSeguro/administracion/ADMINREPORTES_REPORT167_PARAMFORM.jsp</valor></parametro>
        <parametro><llave>bottom</llave><valor>/WEB-
INF/jspSeguro/comun/FOOTER_DEFAULT.jsp</valor></parametro>
    </vista>

```

**SIR-APP\src\org.auriga.smart.web.xml**

```

<!-- AHERRENO 14/03/2011
report-167-->
<navegacion>
    <resultado>REPORTE_167_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK</resultado>
    <url-redireccion>admin.reportes.report-167.param-form.view</url-redireccion>
</navegacion>

```

**SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES\_REPORT167\_PARAMFORM**

**Archivo Nuevo.**

**SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java**

```

// AHERRENO 14/03/2011
// Reporte 167
public static final String REPORTE_167_SEND_ACTION = "REPORTE_167_SEND_ACTION";
public static final String REPORTE_167_BACK_ACTION = "REPORTE_167_BACK_ACTION";

// AHERRENO 14/03/2011
// Reporte 167
public static final String REPORTE_167_LOADCIRCULOSBYUSUARIO =
"REPORTE_167_LOADCIRCULOSBYUSUARIO";

// AHERRENO 14/03/2011
// Declaración de las Variables para el reporte 167
public static final String REPORTE_167_PARAM_CMDKEY      = "reporte-167.jasper";
public static final String REPORTE_167_PARAM_PFECHAINI   = "V_FECHAINI";
public static final String REPORTE_167_PARAM_PFECHAFIN   = "V_FECHAFIN";
public static final String REPORTE_167_PARAM_PCIRCULONOMBRE = "V_CIRCULO";
public static final String REPORTE_167_PARAM_USUARIOLOG  = "P_USUARIO_LOG";

//AHERRENO 14/03/2011
//Reporte 167
if (REPORTE_167_LOADCIRCULOSBYUSUARIO.equals(accion)) {
    return do_ReporteXX_LoadCirculosByUsuario_Action(request);
}

// AHERRENO 14/03/2011
// Reporte 167
if (REPORTE_167_SEND_ACTION.equals(accion)) {
    do_Reporte167_Send_Action(request);
}
if (REPORTE_167_BACK_ACTION.equals(accion)) {
    do_Reporte167_Back_Action(request);

//AHERRENO 14/03/2011
//Reporte 167
private void do_Reporte167_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_167_PARAM_PFECHAINI);
    String param_02 = request.getParameter(REPORTE_167_PARAM_PFECHAFIN);
    String param_03 = request.getParameter(REPORTE_167_PARAM_PCIRCULONOMBRE);
    String param_04 = request.getParameter(REPORTE_167_PARAM_USUARIOLOG);

    GregorianCalendar capturaFecha1 = null;
    GregorianCalendar capturaFecha2 = null;
    int ano1 = 0;
    int ano2 = 0;
    int mes1 = 0;
    int mes2 = 0;
}

```

```

int dia1 = 0;
int dia2 = 0;

// apply validators
if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {

    exception.addError("parametro: Fecha Inicio Radicacion: Debe digitar un valor");

} else {
    String send1 = null;

    Date fecha = null;
    try {
        fecha = DateFormatUtil.parse(param_01);
        send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
        param_01 = send1;

        String[] fechas = param_01.split("-");//yyyy-MM-dd - 0-1-2
        ano1 = Integer.parseInt(fechas[0]);
        mes1 = Integer.parseInt(fechas[1]);
        dia1 = Integer.parseInt(fechas[2]);
        capturaFecha1 = new GregorianCalendar(ano1,mes1,dia1);

    } catch (Exception e) {
        exception.addError("parametro: Fecha Inicio Radicacion: La fecha es inválida" + e.getMessage());
    }
}

// apply validators
if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {

    exception.addError("parametro: Fecha Fin Radicacion: Debe digitar un valor");

} else {
    String send1 = null;

    Date fecha = null;
    try {
        fecha = DateFormatUtil.parse(param_02);
        send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
        param_02 = send1;

        String[] fechas = param_02.split("-");//yyyy-MM-dd - 0-1-2
        ano2 = Integer.parseInt(fechas[0]);
        mes2 = Integer.parseInt(fechas[1]);
        dia2 = Integer.parseInt(fechas[2]);
        capturaFecha2 = new GregorianCalendar(ano2,mes2-1,dia2); //Los meses van de 0 a 11
    } catch (Exception e) {
        exception.addError("parametro: Fecha Fin Radicacion: La fecha es inválida" + e.getMessage());
    }
}

if(capturaFecha1 != null && capturaFecha2 != null){
    if(!(mes1 == mes2 && ano1 == ano2)){

        Calendar valFec = Calendar.getInstance();
        valFec.set(ano1, mes1-1, dia1, 0, 0, 0); //Los meses van de 0 a 11
        valFec.add(Calendar.DATE, 180);

        if( capturaFecha2.after(valFec)){
            exception.addError("El Rango de fechas no puede exceder a seis (6) meses");
        }
    }
}

// apply validators
if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {

    exception.addError("parametro: Circulo: Debe digitar un valor");

} else {

    if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    }
}

```

```

    }

    // apply validators
    if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
        exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
            exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
        }
    }
    String[] itemIds = new String[]{
        REPORTE_167_PARAM_PFECHAINI, REPORTE_167_PARAM_PFECHAFIN,
        REPORTE_167_PARAM_PCIRCULONOMBRE, REPORTE_167_PARAM_USUARIOLOG
    };

    do_Reporte167_SaveState(request);
    // raise -----
    if (exception.getErrores().size() > 0) {
        throw exception;
    }
    // -----
    String[] params = new String[]{
        REPORTE_XX_PARAM_CMDKEY, REPORTE_167_PARAM_PFECHAINI,
        REPORTE_167_PARAM_PFECHAFIN, REPORTE_167_PARAM_PCIRCULONOMBRE,
        REPORTE_167_PARAM_USUARIOLOG
    };

    String[] values = new String[]{
        REPORTE_167_PARAM_CMDKEY, param_01, param_02, param_03, param_04
    };

    String url = makeUrl(params, values);
    reportsServices_ConfigureParameterUri(request, url);

    do_ReporteXX_RemoveState(itemIds, request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

private void do_Reporte167_Back_Action(HttpServletRequest request) {
    do_Reporte167_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
} // :do_Reporte167_Back_Action

private void do_Reporte167_SaveState(HttpServletRequest request) {
    HttpSession session;
    String[] itemIds = new String[]{
        REPORTE_167_PARAM_PFECHAINI, REPORTE_167_PARAM_PFECHAFIN,
        REPORTE_167_PARAM_PCIRCULONOMBRE, REPORTE_167_PARAM_USUARIOLOG
    };

    session = request.getSession();
    save_PageItemsState(itemIds, request, session);
}

private void do_Reporte167_RemoveState(HttpServletRequest request) {
    HttpSession session;

    String[] itemIds = new String[]{
        REPORTE_167_PARAM_PFECHAINI, REPORTE_167_PARAM_PFECHAFIN,
        REPORTE_167_PARAM_PCIRCULONOMBRE, REPORTE_167_PARAM_USUARIOLOG
    };

    session = request.getSession();
    delete_PageItemsState(itemIds, request, session);
}

} // :do_Reporte167_RemoveState

```

#### JASPER – reporte-167.jasper

Archivo Nuevo.

SIR-APP\src\gov\sir\core\web\navegacion\administracion\ControlNavegacionReportes.java

//AHERRENO 14/03/2011

//Reporte 167

```

public static final String REPORTE_167_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK =
"REPORTE_167_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK";

//AHERRENO 14/03/2011
//Reporte 167
listaReportesJasper.add(AWReportes.REPORTE_167_SEND_ACTION);

```

```

//AHERRENO 14/03/2011
// -- navegacion sobre pagina
if( AWReportes.REPORTE_167_LOADCIRCULOSBYUSUARIO.equals( accion ) ){
    return REPORTE_167_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK;
}

//AHERRENO 14/03/2011
//Reporte 167
if( AWReportes.REPORTE_167_BACK_ACTION.equals( accion ) ){
    return REPORTE_XX_BACK_EVENTRESP_OK;
}

```

#### SIR-APP\src\gov\sir\hermod\dao\impl\jdomgenie\JDORreportesJasperDAO.java

```

//AHERRENO 14/03/2011
//Reporte 167
//Se obtienen los parámetros para enviar al procedimiento
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_167_PARAM_PFECHAINI);
listaParametros.add(AWReportes.REPORTE_167_PARAM_PFECHAFIN);
listaParametros.add(AWReportes.REPORTE_167_PARAM_PCIRULONOMBRE);
mapaReportes.put(AWReportes.REPORTE_167_PARAM_CMDKEY, new Object[]{
    "SELECT seq_reporte_167.NEXTVAL FROM dual",
    "sp_reporte_167(?, ?, ?, ?)", listaParametros,
    "DELETE FROM sir_op_reporte_167 WHERE sec_proceso = "});

```

### Configuración Reporte 168 – Reimpresiones de Certificados por Oficina

#### SIR-APP\src\org.auriga.smart.vistas.xml

```

<!-- AHERRENO 14/03/2011
report-168 -->

<vista>
    <autenticacion>true</autenticacion>

    <excepcion>gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector</e
xcepcion>
        <excepcion>gov.sir.core.web.acciones.excepciones.ValidacionParametrosException</excepcion>
        <nombre>admin.reportes.report-168.param-form.view</nombre>
        <parametro><llave>error</llave><valor>/WEB-
INF/jspSeguro/comun/ERROR_MULTIPLE.jsp</valor></parametro>
        <parametro><llave>head</llave><valor>/WEB-
INF/jspSeguro/comun/HEADER_LOGOUT.jsp</valor></parametro>
        <parametro><llave>body</llave><valor>/WEB-
INF/jspSeguro/administracion/ADMINREPORTES_REPORT168_PARAMFORM.jsp</valor></parametro>
        <parametro><llave>bottom</llave><valor>/WEB-
INF/jspSeguro/comun/FOOTER_DEFAULT.jsp</valor></parametro>
</vista>

```

#### SIR-APP\src\org.auriga.smart.web.xml

```

<!-- AHERRENO 14/03/2011
report-168 -->
<navegacion>
    <resultado>REPORTE_168_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK</resultado>
    <url-redirect>admin.reportes.report-168.param-form.view</url-redirect>
</navegacion>

```

#### SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES\_REPORT168\_PARAMFORM

### Archivo Nuevo.

#### SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java

```

// AHERRENO 14/03/2011
// Reporte 168
public static final String REPORTE_168_SEND_ACTION = "REPORTE_168_SEND_ACTION";
public static final String REPORTE_168_BACK_ACTION = "REPORTE_168_BACK_ACTION";

// AHERRENO 14/03/2011
// Reporte 168
public static final String REPORTE_168_LOADCIRCULOSBYUSUARIO =
"REPORTE_168_LOADCIRCULOSBYUSUARIO";

// Declaración de las Variables para el reporte 168

```

```

public static final String REPORTE_168_PARAM_CMDKEY      = "reporte-168.jasper";
public static final String REPORTE_168_PARAM_PFECHAINI   = "V_FECHAINI";
public static final String REPORTE_168_PARAM_PFECHAFIN   = "V_FECHAFIN";
public static final String REPORTE_168_PARAM_PCIRCULONOMBRE = "V_CIRCULO";
public static final String REPORTE_168_PARAM_USUARIOLOG  = "P_USUARIO_LOG";

//AHERRENO 14/03/2011
//Reporte 168
if (REPORTE_168_LOADCIRCULOSBYUSUARIO.equals(accion)) {
    return do_ReporteXX_LoadCirculosByUsuario_Action(request);
}

// AHERRENO 14/03/2011
// Reporte 168
if (REPORTE_168_SEND_ACTION.equals(accion)) {
    do_Reporte168_Send_Action(request);
}
if (REPORTE_168_BACK_ACTION.equals(accion)) {
    do_Reporte168_Back_Action(request);
}

//AHERRENO 14/03/2011
//Reporte 168
private void do_Reporte168_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_168_PARAM_PFECHAINI);
    String param_02 = request.getParameter(REPORTE_168_PARAM_PFECHAFIN);
    String param_03 = request.getParameter(REPORTE_168_PARAM_PCIRCULONOMBRE);
    String param_04 = request.getParameter(REPORTE_168_PARAM_USUARIOLOG);

    GregorianCalendar capturaFecha1 = null;
    GregorianCalendar capturaFecha2 = null;
    int ano1 = 0;
    int ano2 = 0;
    int mes1 = 0;
    int mes2 = 0;
    int dia1 = 0;
    int dia2 = 0;
    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {

        exception.addError("parametro: Fecha Inicio Reimpresion: Debe digitar un valor");

    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;

            String[] fechas = param_01.split("-");//yyyy-MM-dd - 0-1-2
            ano1 = Integer.parseInt(fechas[0]);
            mes1 = Integer.parseInt(fechas[1]);
            dia1 = Integer.parseInt(fechas[2]);
            capturaFecha1 = new GregorianCalendar(ano1,mes1,dia1);
        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio Reimpresion: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {

        exception.addError("parametro: Fecha Fin Reimpresion: Debe digitar un valor");

    } else {
        String send1 = null;

        Date fecha = null;

```

```

try {
    fecha = DateFormatUtil.parse(param_02);
    send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
    param_02 = send1;

    String[] fechas = param_02.split("-");//yyyy-MM-dd - 0-1-2
    ano2 = Integer.parseInt(fechas[0]);
    mes2 = Integer.parseInt(fechas[1]);
    dia2 = Integer.parseInt(fechas[2]);
    capturaFecha2 = new GregorianCalendar(ano2,mes2-1,dia2); //Los meses van de 0 a 11
} catch (Exception e) {
    exception.addError("parametro: Fecha Fin Reimpresion: La fecha es inválida" + e.getMessage());
}
}

if(capturaFecha1 != null && capturaFecha2 != null){
    if(!(mes1 == mes2 && ano1 == ano2)) {

        Calendar valFec = Calendar.getInstance();
        valFec.set(ano1, mes1-1, dia1, 0, 0, 0); //Los meses van de 0 a 11
        valFec.add(Calendar.DATE, 180);

        if( capturaFecha2.after(valFec)){
            exception.addError("El Rango de fechas no puede exceder a seis (6) meses");
        }
    }
}

// apply validators
if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {

    exception.addError("parametro: Circulo: Debe digitar un valor");

} else {

    if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    }
}

// apply validators
if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
    exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
        exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
    }
}
String[] itemIds = new String[]{
    REPORTE_168_PARAM_PFECHAINI, REPORTE_168_PARAM_PFECHAFIN,
    REPORTE_168_PARAM_PCIRULONOMBRE, REPORTE_168_PARAM_USUARIOLOG
};

do_Reporte168_SaveState(request);
// raise ----

if (exception.getErrores().size() > 0) {
    throw exception;
}
// ----

String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY, REPORTE_168_PARAM_PFECHAINI,
    REPORTE_168_PARAM_PFECHAFIN, REPORTE_168_PARAM_PCIRULONOMBRE,
    REPORTE_168_PARAM_USUARIOLOG
};

String[] values = new String[]{
    REPORTE_168_PARAM_CMDKEY, param_01, param_02, param_03, param_04
};

String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_ReporteXX_RemoveState(itemIds, request);

```

```

        do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
    }

private void do_Reporte168_Back_Action(HttpServletRequest request) {
    do_Reporte168_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
} // :do_Reporte168_Back_Action

private void do_Reporte168_SaveState(HttpServletRequest request) {
    HttpSession session;
    String[] itemIds = new String[]{};
    REPORTE_168_PARAM_PFECHAINI, REPORTE_168_PARAM_PFECHAFIN,
    REPORTE_168_PARAM_PCIRCULONOMBRE, REPORTE_168_PARAM_USUARIOLOG
};

session = request.getSession();
save_PageItemsState(itemIds, request, session);

} // :do_Reporte168_SaveState

private void do_Reporte168_RemoveState(HttpServletRequest request) {
    HttpSession session;

    String[] itemIds = new String[]{};
    REPORTE_168_PARAM_PFECHAINI, REPORTE_168_PARAM_PFECHAFIN,
    REPORTE_168_PARAM_PCIRCULONOMBRE, REPORTE_168_PARAM_USUARIOLOG
};

session = request.getSession();
delete_PageItemsState(itemIds, request, session);

} // :do_Reporte168_RemoveState
JASPER – reporte-168.jasper
Archivo Nuevo.
SIR-APP\src\gov\sir\core\web\navegacion\administracion\ControlNavegacionReportes.java
//AHERRENO 14/03/2011
//Reporte 168
public static final String REPORTE_168_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK =
"REPORTE_168_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK";

//AHERRENO 14/03/2011
//Reporte 168
listaReportesJasper.add(AWReportes.REPORTE_168_SEND_ACTION);

//AHERRENO 14/03/2011
// -- navegacion sobre pagina
if( AWReportes.REPORTE_168_LOADCIRCULOSBYUSUARIO.equals( accion ) ){
    return REPORTE_168_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK;
}

//AHERRENO 14/03/2011
//Reporte 168
if( AWReportes.REPORTE_168_BACK_ACTION.equals( accion ) ) {
    return REPORTE_XX_BACK_EVENTRESP_OK;
}
SIR-APP\src\gov\sir\hermod\dao\impl\jdom\JDOReportesJasperDAO.java
//AHERRENO 14/03/2011
//Reporte 168
//Se obtienen los parámetros para enviar al procedimiento
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_168_PARAM_PFECHAINI);
listaParametros.add(AWReportes.REPORTE_168_PARAM_PFECHAFIN);
listaParametros.add(AWReportes.REPORTE_168_PARAM_PCIRCULONOMBRE);
mapaReportes.put(AWReportes.REPORTE_168_PARAM_CMDKEY, new Object[]{
    "SELECT seq_reporte_168.NEXTVAL FROM dual",
    "sp_reporte_168(?, ?, ?, ?)", listaParametros,
    "DELETE FROM sir_op_reporte_168 WHERE sec_proceso = ''"});
mapas.jar
org.auriga.smart.web.xml
org.auriga.smart.vistas.xml

```

## 251 Fallo en folios derivados al hacer una segregación Informativo

***REQUERIMIENTO***

Requerimiento Glpi No 26181

Turno:2011-350-3-372

Matricula(s):350-116844

Motivo Inconsistencia : No avanza de revisar y aprobar

Se comunica la funcionaria Venovit Mejía informando que el turno no avanza de la fase de revisar y aprobar. Les genera el siguiente error "Fallo en el servicio para hacer definitivos los datos temporales, ver imagen"

Grupo Ibagué

***DESARROLLO***

Se realizan pruebas con la base de datos de desarrollo circulo Ibagué, se creó el turno 2011-350-3-10, se ingresan 3 matriculas

350-63326

350-63333

350-116844

1. En el folio 350-63326 se cambia el estado de las anotaciones 6 y 7 de validas a invalidas.
2. En el folio 350-63333 se crean dos anotaciones 3 y 4 teniendo como base las anotaciones 6 y 7 del folio 350-63326.
3. El folio 350-116844 tiene segregación con la matricula 350-63326 anotación 6 esta se elimina y se agrega segregación con la matricula 350-63333 anotación 6.

Se finaliza el turno si presentar la inconsistencia reportada.

CAJERO CORRECCIONES ANGELA.OSPINA

RESPONSABLE CORRECCIONES - ANALISIS LESVY.MUNOZ

AUXILIAR CORRECCIONES ELICINA.GUZMAN

RESPONSABLE CORRECCIONES - APROBAR LESVY.MUNOZ

MESA CONTROL CORRECCIONES LESVY.MUNOZ

ENTREGA CORRECCIONES ANGELA.OSPINA

Se realizan pruebas en ambiente de producción circulo pruebas, turno 2011-999-3-46 con las matriculas:

999-417

999-418

999-425

1. En el folio 999-417 se cambia el estado de las anotaciones 3 y 4 de validas a invalidas.

2. En el folio 999-425 se crean dos anotaciones 4 y 5 teniendo como base las anotaciones 3 y 4 del folio 999-417.

3. El folio 999-418 tiene segregación con la matricula 999-417 anotación 3 esta se elimina y se agrega segregación con la matricula 999-425 anotación 4

Avanza de la fase de Auxiliar de Correcciones a Responsable Correcciones-Revisar Aprobar, al tratar de avanzar de esta fase se replica el error reportado por los usuarios

Fallo en el servicio para hacer definitivos los datos temporales : No row for  
gov.sir.core.negocio.modelo.jdogenie.AnotacionTMP (SIR\_NE\_ANOTACION\_TMP)

OID:gov.sir.core.negocio.modelo.jdogenie.AnotacionTMP (Table:SIR\_NE\_ANOTACION\_TMP Column:ID\_ANOTACION\_TMP = 4 Column:ID\_MATRICULA = 999-418).

La inconsistencia presentada por este requerimiento, fue resuelta en el caso mantis 5306 Acta - Requerimiento No 203 - Desfase de Id anotación temporales.

Por lo tanto se procede a cerrar el requerimiento.

### 253 Error al Crear Vereda Informativo

#### **REQUERIMIENTO Glpi No 26488**

Motivo Inconsistencia: Se crea el Municipio Tuchín del Departamento de Córdoba por solicitud de la ORIP, pero al crear la Vereda Tuchín y agregarla como Cabecera Municipal se genera error.

Favor crear la Vereda Tuchín del Municipio: Tuchin, Departamento: Córdoba, y agregarla como Cabecera Municipal, se anexa imagen.

Grupo Chinu

#### **DESARROLLO**

Análisis: al revisar una de las consultas de validación se encontró el error, esto solo sucede cuando se crea la primera vereda de un municipio.

Se modifica la consulta corrigiendo este problema.

<b>CLASE o ARCHIVO JSP: gov.sir.forseti.dao.impl.jdogenie.JDOGenieZonaRegistralDAO.java</b>
<p><b>Línea 3523:</b></p> <pre>Linea 3523: /**  * @author : Fernando Padilla  * @change : Se realiza modificaciones sobre el codgio de el metodo, ya que,  *           al realizar la consulta generaba error.  * Caso Mantis : 0007875  */ private String getIdVereda(String departamento, String municipio, PersistenceManager pm) throws DAOException{     String idVereda = null;     int aux;     Connection con = null;     PreparedStatement pst = null;     ResultSet rs = null;      try{         VersantPersistenceManager pm2 = (VersantPersistenceManager) pm;         con = pm2.getJdbcConnection(null);         pst = con.prepareStatement("SELECT max(V.ID_VEREDA) FROM SIR.SIR_NE_VEREDA V WHERE V.ID_MUNICIPIO = ? and v.ID_DEPARTAMENTO = ?");         pst.setString(1, municipio);         pst.setString(2, departamento);         rs = pst.executeQuery();         if(rs.next()){             idVereda = rs.getString(1);         }         if(idVereda==null){             aux = 0;             idVereda = Integer.toString(aux);         }else{             aux = Integer.parseInt(idVereda)+1;             idVereda = Integer.toString(aux);         }         if(aux &lt; 100){             aux = 100;             idVereda = Integer.toString(aux);         }     }catch(Exception e){         e.printStackTrace();     } }</pre>

```

        if(aux < 10)
            idVereda = "0" + idVereda;
        idVereda = "0" + idVereda;
    }
} catch (SQLException ex) {
    Logger.getLogger(JDOGenieZonaRegistralDAO.class.getName()).log(Level.SEVERE, null, ex);
} catch (JDOException e) {
    Log.getInstance().error(JDOGenieZonaRegistralDAO.class, e.getMessage());
    throw new DAOException(e.getMessage(), e);
}
finally{
try {
    if(rs != null){
        rs.close();
    }
    if(pst != null){
        pst.close();
    }
}

if(con != null){
    con.close();
}
} catch (SQLException ex) {
    ex.printStackTrace();
}
}
return idVereda;
}

```

## 254 RESTITUCION TURNOS DE TESTAMENTOS

### **REQUERIMIENTO: RESTITUCION TURNOS DE REGISTRO ASOCIADOS CON ACTOS DE TESTAMENTOS**

Se requiere que en la opción de bloquear y restituir de pantallas administrativas, el aplicativo permita lo siguiente:

1-Al seleccionar la opción de Bloquear, el aplicativo debe permitir ingresar turnos de registro que contengan actos de Testamentos y realizar el mismo procedimiento, validaciones y reglas que se aplican actualmente para el bloqueo de los folios asociados a turnos de registro que requieren restitución.

2-Al seleccionar la opción de Restituir el aplicativo debe mostrar la pantalla con la información, procedimientos, validaciones y reglas que existen actualmente para la restitución de turnos de registro, agregando en el bloque lista de calificadores, los nombres de los usuarios activos que tienen el rol de testamentos.

El aplicativo debe validar que los turnos de registro correspondientes a la fase de testamentos al confirmar la restitución se deben re direccionar al usuario que le corresponda el rol de esta fase,

Así mismo al seleccionar un turno y el botón rechazar en la pantalla mencionada en este ítem , el aplicativo debe realizar las misma acciones definidas para los turnos de registro al seleccionar el botón mencionado.

### **DESARROLLO**

#### **ARCHIVO JSP:**

**WEB-INF.jspSeguro.administracion.revocatoriasdirectas. REANOTAR\_RECURSOS.jsp**

#### **Código Insertado Línea 43:**

```

/*
* @author : Julio Alcázar Rivas
* @change : nueva variable para manejar en el evento los usuarios con el rol testamentos
* Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
*/
List caliTestamentos =(List)session.getAttribute(WebKeys.LISTA_CAL_TESTAMENTOS);

```

### **Código Eliminado Lineas 173 y 198:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : se elimino la variable primerCalificador para no checked ninguno de los elementos del
radio
 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
*/
```

### **Cambio Código Línea 187:**

```
<td width="50"><input type='radio'
name='<%=gov.sir.core.negocio.modelo.constantes.CEstacion.ESTACION_ID%>'
value='<%=e.getId()%>' <%=primerCalificador?"checked":""%>></td>
```

### **POR:**

```
<!--
 * @author : Julio Alcázar Rivas
 * @change : se elimino la variable primerCalificador para no checked ninguno de los elementos
del radio
 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE
TESTAMENTOS
!-->
<td width="50"><input type='radio'
name='<%=gov.sir.core.negocio.modelo.constantes.CEstacion.ESTACION_ID%>'
value='<%=e.getId()%>'></td>
```

### **Código Insertado Línea 216:**

```
<!--
 * @author : Julio Alcázar Rivas
 * @change : se creo una nueva tabla para colocar los usuarios que son calificadores de
testamentos
 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
!-->
<br>
<table width="100%" border="0" cellpadding="0" cellspacing="0">
<tr>
    <td width="12"></td>
    <td class="bgnsub">Lista de calificadores de testamentos</td>
    <td width="16" class="bgnsub"></td>
    <td width="16" class="bgnsub"></td>
    <td width="15"></td>
</tr>
</table>
<%
if(caliTestamentos!=null && !caliTestamentos.isEmpty()){%>
<table width="100%" class="camposform">
<%
java.util.Iterator it = caliTestamentos.iterator();
while(it.hasNext()){
    org.auriga.core.modelo.transferObjects.Estacion e =
(org.auriga.core.modelo.transferObjects.Estacion)it.next();
    %>
<tr>
    <td width="20"></td>
<td width="50"><input type='radio' name='EST_TESTAMENTO'
value='<%=e.getEstacionId()%>'></td>
<td >
<%=e.getEstacionId()%>
</td>
</tr>
<%
}

}
<%>
</table>
<%>else{%
<table width="100%" class="camposform">
<tr>
<td width="20"></td>
<td >
No existen calificadores de testamentos para asignar los turnos a reanotar.
</td>
</tr>
</table>
<%}>

```

**CLASE:**

**gov.sir.core.negocio.acciones.administracion.ANRecursosRevocatorias.java**

**a**

**Método:**

**private EvnRespRecursosRevocatorias  
seleccionarTurno(EvnRecursosRevocatorias evento) throws  
EventoException {**

**Código Insertado Línea 225:**

```

/*
* @author : Julio Alcázar Rivas
* @change : Se agrega la validacion para turnos que pasaron por el rol_testamentos
* Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
*/
else if(th.getFase().equals(CFase.REG_TESTAMENTO)){
    valido = false;
    if(th.getRespuesta().equals(CRespuesta.NEGAR) ){
        valido = true;
    }
}

```

**Método:**

**private EvnRespRecursosRevocatorias  
consultarTurnosBloqueados(EvnRecursosRevocatorias evento) throws  
EventoException {**

**Código Insertado Línea 467:**

```

/*
* @author : Julio Alcázar Rivas
* @change : nueva variable para manejar los usuarios con el rol testamentos
* Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
*/
List caliTestamentos = null;
```

**Código Insertado Línea 483:**

```

/*
* @author : Julio Alcázar Rivas
* @change : se obtienen los usuarios con el rol testamentos
```

```

 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
 */
rol.setRolId(CRol.SIR_ROL_TESTAMENTOS);
caliTestamentos = fenrir.obtenerEstacionesUsuarioByEstadoRol(rol , circulo, true);

```

### **Código Insertado Linea 503:**

```

/*
 * @author    : Julio Alcázar Rivas
 * @change    : validacion para la variable que maneja los usuarios con el rol testamentos
 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
 */
if(caliTestamentos==null){
    caliTestamentos = new ArrayList();
}

```

### **Código Insertado Linea 515:**

```

/*
 * @author    : Julio Alcázar Rivas
 * @change    : set en la respuesta los usuarios con el rol testamentos
 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
 */
rta.setListaCalTestamentos(caliTestamentos);

```

#### **Método:**

```

private EvnRespRecursosRevocatorias
reanotarRecursosRevocatorias(EvnRecursosRevocatorias evento) throws
EventoException {

```

### **Código Insertado Linea 539:**

```

/*
 * @author    : Julio Alcázar Rivas
 * @change    : se agrega nueva variable para manejar las estaciones con rol testamento
 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
 */
Estacion estacionTest = evento.getEstacionTest();

```

### **Código Linea 581:**

```

hermod.reanotarTurnoModeloOperativo(turno.getIdWorkflow() , "1" , CFase.CAL_CALIFICACION,
CRespuesta.CONFIRMAR,estacion.getEstacionId() , usuario );

```

#### **POR:**

```

/*
 * @author    : Julio Alcázar Rivas
 * @change    : se selecciona el metodo reanotarTurnoModeloOperativo() con diferentes parametros
 dependiendo de la estacion.
 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
 */
if(estacionTest == null){
    hermod.reanotarTurnoModeloOperativo(turno.getIdWorkflow() , "1" , CFase.CAL_CALIFICACION,
CRespuesta.CONFIRMAR,estacion.getEstacionId() , usuario );
}else {
    hermod.reanotarTurnoModeloOperativo(turno.getIdWorkflow() , "1" , CFase.REG_TESTAMENTO,
CRespuesta.CONFIRMAR,estacionTest.getEstacionId() , usuario );
}

```

#### **CLASE:**

```

gov.sir.core.web.acciones.administracion. AWRecursosRevocatorias.java

```

#### **Método:**

```

private EvnRecursosRevocatorias
reanotarRecursosRevocatorias(HttpServletRequest request) throws
AccionWebException {

```

**Código Insertado Linea 259:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : nueva variable para tener en cuenta estaciones de rol testamentos
 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
 */
String idEstacionTest = request.getParameter("EST_TESTAMENTO");
```

**Código Linea 279:**

```
if(idEstacion==null || idEstacion.equals("")){  
    exception.addError("Debe seleccionar el calificador a quién le desea asignar el turno.");
```

**POR:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Cambio en la validacion para tener en cuenta estaciones de rol testamentos
 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
 */
if((idEstacion==null || idEstacion.equals("")) && (idEstacionTest==null || idEstacionTest.equals(""))){  
    exception.addError("Debe seleccionar el calificador (Turno de Calificacion o Testamento) a quién  
le desea asignar el turno.");
```

**Código Linea 297:**

```
Estacion estacion = new Estacion(idEstacion);
```

**POR:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Cambio en la validacion para tener en cuenta estaciones de rol testamentos
 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
 */
Estacion estacion = null;  
Estacion estacionTest = null;  
if(idEstacion==null || idEstacion.equals("")){  
    estacionTest = new Estacion(idEstacionTest);  
}else{  
    estacion = new Estacion(idEstacion);  
}
```

**Código Insertado Linea 311:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : se agrega al evento la estacion con rol testamento
 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
 */
evento.setEstacionTest(estacionTest);
```

**Método:**

```
public void doEnd(HttpServletRequest request, EventoRespuesta evento)  
{
```

**Código Insertado Linea 414:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Se agrega a la sesion los usuarios con rol testamentos
 * Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE TESTAMENTOS
 */
session.setAttribute(WebKeys.LISTA_CAL_TESTAMENTOS, respuesta.getListaCalTestamentos());
```

**CLASE:**

```
gov.sir.core.eventos.administracion.EvnRecursosRevocatorias.java
```

**Código Insertado Linea 56:**

```
/*
```

```

* @author : Julio Alcázar Rivas
* @change : se agrega la propiedad EstacionTest con sus metodos GET y SET.
* Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE
TESTAMENTOS
*/
private Estacion EstacionTest;
public Estacion getEstacionTest() {
    return EstacionTest;
}
public void setEstacionTest(Estacion EstacionTest) {
    this.EstacionTest = EstacionTest;
}

```

**CLASE:**

**gov.sir.core.eventos.administracion. EvnRespRecursosRevocatorias.java**

**Código Insertado Linea 32:**

```

/*
* @author : Julio Alcázar Rivas
* @change : nueva variable para manejar en el evento los usuarios con el rol testamentos
* Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE
TESTAMENTOS
*/
List listaCalTestamentos = null;
public List getListaCalTestamentos() {
    return listaCalTestamentos;
}

public void setListaCalTestamentos(List listaCalTestamentos) {
    this.listaCalTestamentos = listaCalTestamentos;
}

```

**CLASE:**

**gov.sir.core.web.WebKeys.java**

**Código Insertado Linea 1235:**

```

/*
* @author : Julio Alcázar Rivas
* @change : nueva constante para manejar los usuarios con el rol testamentos
* Caso Mantis : 0007920: Acta - Requerimiento No 254 - RESTITUCION TURNOS DE
TESTAMENTOS
*/
public static final String LISTA_CAL_TESTAMENTOS = "LISTA_CAL_TESTAMENTOS";

```

## 255 SOLICITUD CERTIFICADOS EXENTOS MASIVOS NUMERO DEL DOCUMENTO

**REQUERIMIENTO: CAMPO NÚMERO DE DOCUMENTO . SOLICITUD CERTIFICADOS EXENTOS MASIVOS. (ROL USUARIO OPERATIVO)**

Se requiere que en la pantalla solicitud de certificados exentos masivos en el bloque de encabezado del documento, - en el campo número de documento, el aplicativo debe permitir ingresar información alfanumérica.

Verificada la estructura de base de datos , en la tabla SIR\_OP\_SOLICITUD se observa que está definido un campo ID DOCUMENTO que tiene integridad referencial con la tabla SIR\_NE\_DOCUMENTO , tabla que es utilizada para el ingreso de datos en el proceso de radicación de documentos y en el cual la información relacionada con el numero de documento admite datos alfanuméricos, dado que el campo DCMN\_NUMERO es de tipo varchar2 ;razón por la cual no es aceptable que en el aplicativo se tenga una restricción de en el campo de numero de documento en el proceso de radicación de certificados exentos en solo admitir datos numéricos.

**DESARROLLO**

Script Bases de Datos:

Ninguno

Creacion De Imagenes:

Ninguno

Se modificaron los siguientes archivos:

Ninguno

**CLASE:**

**gov.sir.core.web.acciones.certificadosmasivos.AWLiquidacionCertificadosMasivos.java**

**Método:**

**private EvnCertificadoMasivo liquidar(HttpServletRequest request) throws AccionWebException{**

**Código Eliminado Lineas 714 y 717:**

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : Se eliminó la validacion de tipo numerico en el numero del documento
 * Caso Mantis : 0007915 Acta - Requerimiento No 255 - SOLICITUD CERTIFICADOS EXENTOS
MASIVOS NUMERO DEL DOCUMENTO
*/
```

**256 IMPRESION FOLIO SIMPLE****REQUERIMIENTO**

En la opción de IMPRESIÓN SIMPLE DE FOLIO, pantallas administrativas del aplicativo SIR, se requiere que el sistema no permita la impresión simple de folios de matricula inmobiliaria de otro circulo diferente desde el cual se realiza la impresión.

Es importante precisar, que el Aplicativo tenia esta regla de restricción que aplica no solo para impresión en mención, sino que también aplica para la consulta de turnos y consulta de folios de matricula inmobiliaria.

Se solicita se establezca el origen del error que genera esta inconsistencia.

**DESARROLLO****CLASE:**

**gov.sir.core.web.acciones.administracion.AWImpresionFolio.java**

**Método:**

**private EvnImprimirFolio imprimirFolio(HttpServletRequest request)**  
**throws AccionWebException {**

**Código Eliminado Lineas 212:**

```
/**
 * @author    : Julio Alcazar
 * @change    : se agrega la validación de que el folio pertenezca al circulo actual.
 * Caso Mantis : 0008005: Acta - Requerimiento No 256 - IMPRESION FOLIO SIMPLE
*/
Circulo circuloActual = (Circulo) session.getAttribute(WebKeys.CIRCULO);
if(!circuloActual.getIdCirculo().equals(id.substring(0, 3))){
    exception.addError("No se puede Imprimir Folios de otros circulos.");
}
```

**246 Reimpresion de Certificados V2****REQUERIMIENTO:** COMPLEMENTO REIMPRESION DE CERTIFICADOS

Revisado el despliegue 2-07-0 se encuentran las siguientes inconsistencias que deben ser corregidas:

- 1) Al incluir un turno de certificado que ya agoto las dos reimpresiones y si selecciona cambiar matrícula (así este dentro del término para el cambio de matrícula) el aplicativo validara y mostrará un mensaje restrictivo “El turno de certificado ha alcanzado el número máximo dos (2) reimpresiones por lo tanto no se permite el cambio de matrícula”.
- 2) El aplicativo no debe permitir cambiar la matrícula a un turno de certificado, si el turno ya fue reimpreso al menos una sola vez con la matrícula incluida inicialmente en la solicitud.
- 3) Si un turno de certificado de pertenencia, Antiguo Sistema y Ampliación de la tradición tiene resolución Devuelto al Público el aplicativo no debe permitir reimpresiones o cambio de matrícula, si el turno de certificado es asociado a un turno de registro con resolución devuelto al público no debe permitir la reimpresión ni el cambio de matrícula a excepción de los documentos radicados con acto Embargo.
- 4) Si un turno de certificado de ampliación de la tradición fue finalizado y se le incluyó información a la matrícula que se vincula en la solicitud, el aplicativo debe validar y permitirá la reimpresión mas no permitirá el cambio de matrícula.
- 5) Si a un turno de certificado de Pertenencia y/o Antiguo Sistema se le creó la matrícula en la fase hoja de ruta, quedando vinculada al turno de certificado, el aplicativo deberá validar y no permitirá el cambio de matrícula pero si permitirá la reimpresión con la matrícula creada.
- 6) El aplicativo no debe permitir que a un turno de certificado se le cambie la matrícula por el mismo número de matrícula que tiene asociado el turno de certificado, al incluir el mismo número de matrícula debe mostrar un mensaje restrictivo “El numero de matrícula ingresado es el mismo que está asociado al turno”
- 7) Para los turnos de certificados de Pertenencia, cada reimpresión del oficio de pertenencia el aplicativo las debe reflejar en el historial del turno como reimpresión del oficio y no como reimpresión del certificado de tradición, se debe diferenciar cuando es reimpresión del oficio y reimpresión del certificado en el Reporte de reimpresión de certificados se deben contar únicamente las reimpresiones del certificado de tradición .
- 8) Un turno de certificado que haya tenido cambio de matrícula, agotado las dos reimpresiones y haya transcurrido el término establecido para la reimpresión, al intentar reimprimirlo el aplicativo debe mostrar el mensaje correcto este es “Error al imprimir el folio: se ha alcanzado el número máximo (dos) de reimpresiones, se realizó cambio de matrícula y se venció el término establecido para la reimpresión”. Este mensaje debe aplicar para todos los turnos de certificados a excepción de los certificados Nacionales.

**DESARROLLO**

Script Bases de Datos:

- **PACKAGE pck\_utilidades\_validaciones.sql** firma del package que incluye la FUNCTION is\_create\_folio\_hr(p\_anio IN VARCHAR2, p\_id\_circulo IN VARCHAR2, p\_id\_proceso IN NUMBER, p\_id\_turno IN VARCHAR2) RETURN INTEGER y is\_asociado\_cumple (p\_turno IN VARCHAR2) RETURN INTEGER.
- **PACKAGE pck\_utilidades\_validaciones\_BODY.sql** cuerpo del package que incluye la

```
FUNCTION is_create_folio_hr(p_anio IN VARCHAR2, p_id_circulo IN VARCHAR2, p_id_proceso IN NUMBER, p_id_turno IN VARCHAR2) RETURN INTEGER y is_asociado_cumple (p_turno IN VARCHAR2) RETURN INTEGER;
```

➤ Creacion De Imagenes:

- Ninguno

➤ Se crearon los siguientes archivos:

- co.com.iridium.generalSIR.negocio.validaciones.CertificadosSIR.java
- co.com.iridium.generalSIR.dao.CertificadosSIRDAO.java

➤ Se modificaron los siguientes archivos:

Ninguno

**CLASE:**

**gov.sir.core.negocio.acciones.administracion.ANImprimirFolio.java**

**Método:**

**private EvnRespImprimirFolio imprimirCertificado(EvnImprimirFolio evento) throws EventoException{**

**Cambio Código Línea 993:**

```
this.DiasHabilesCertificado(turnoPersistente, circuloPk, CFase.CER_SOLICITUD, idTurno);
```

**Por**

```
/* @author : Julio Alcázar Rivas
 * @change : Se agrega nuevo parametro a la signatura del metodo int imprCert
 * Caso Mantis : 07647
 */
this.DiasHabilesCertificado(turnoPersistente, circuloPk, CFase.CER_SOLICITUD, idTurno,
maxImprUsuario, numImpresionesCert);
```

**Cambio Código Línea 995:**

```
this.DiasHabilesCertificado(turnoPersistente, circuloPk, turnoPersistente.getIdFase(), idTurno);
```

**Por**

```
/* @author : Julio Alcázar Rivas
 * @change : Se agrega nuevo parametro a la signatura del metodo int imprCert
 * Caso Mantis : 07647
 */
this.DiasHabilesCertificado(turnoPersistente, circuloPk, turnoPersistente.getIdFase(),idTurno,
maxImprUsuario, numImpresionesCert);
```

**Código Insertado Línea 1046:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Validaciones para realizar reimpresiones de turno de certificados
 * Caso Mantis : 07647
 */
List turnosHist = turnoPersistente.getHistorials();
for (Iterator iter = turnosHist.iterator(); iter.hasNext();) {
    TurnoHistoria turnoHistoria = (TurnoHistoria) iter.next();
    if (
((liquidacion.getTipoCertificado().getIdTipoCertificado().equals(CTipoCertificado.TIPO_ANTIGUO_SISTEMA_ID)) ||
liquidacion.getTipoCertificado().getIdTipoCertificado().equals(CTipoCertificado.TIPO_AMPLIACION_TRADICION_ID)) ||
liquidacion.getTipoCertificado().getIdTipoCertificado().equals(CTipoCertificado.TIPO_PERTENENCIA_ID)))
&&
        (turnoHistoria.getFase().equals(CFase.ANT_REVISION_INICIAL) &&
turnoHistoria.getRespuesta().equals(CRespuesta.NEGAR))) {
    throw new ImpresionException("El certificado de tipo:
```

```

        "+liquidacion.getTipoCertificado().getNombre() +" fue negado por lo tanto no se puede realizar
        reimpresión.");
    }

}

if
(liquidacion.getTipoCertificado().getIdTipoCertificado().equals(CTipoCertificado.TIPO_ASOCIADO_ID)) {
    CertificadosSIR certificadosSIR = new CertificadosSIR();
    if (!certificadosSIR.CertificadoAsociadoVal(turno.getIdWorkflow())){
        throw new ImpresionException("no se permite reimpresion de certificados asociados a registro
        con resolución devuelto");
    }
}

```

#### **Método:**

```

private void DiasHabilesCertificado(Turno turno, CirculoPk circuloPk, String
fase, TurnoPk idTurno, int imprUsuario, int imprCert) throws
EventoException, Throwable{

```

#### **Cambio de la firma Del método**

```

private void DiasHabilesCertificado(Turno turno, CirculoPk circuloPk, String fase, TurnoPk idTurno)
throws EventoException, Throwable{

```

#### **Por**

```

private void DiasHabilesCertificado(Turno turno, CirculoPk circuloPk, String fase, TurnoPk idTurno, int
imprUsuario, int imprCert) throws EventoException, Throwable{

```

#### **Código Insertado Línea 1305:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Validaciones para realizar reimpresiones de turno de certificados
 * Caso Mantis : 07647
 */
TurnoPk turnoId = new TurnoPk();
turnoId.anio = turno.getAnio();
turnoId.idCirculo = turno.getIdCirculo();
turnoId.idProceso = turno.getIdProceso();
turnoId.idTurno = turno.getIdTurno();
if (hermod.getNuevoMatriculaAuditoria(turnoId) && (imprUsuario<=imprCert-1)){
    throw new ImpresionException("Se ha alcanzado el número máximo ("+imprUsuario+) de
    reimpresiones, "+ "se realizó cambio de matrícula y se venció el término establecido para la
    reimpresión");
}

```

#### **Método:**

```

private boolean imprimirOficio(EvnImprimirFolio evento, Folio folio, Turno
turnoPersistente, SolicitudCertificado solicitudCert) throws
ImpresionException{

```

#### **Cambio Código Línea 1397:**

```

turnoHist.setFase(CFase.CER_IMPRIMIR);

```

#### **Por**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Cambio de Fase de reimpresión de Oficio de Pertenencia de CER_IMPRIMIR por
CER_OFICIO_PERTENENCIA
 * Caso Mantis : 07647
 */
turnoHist.setFase(CFase.CER_OFICIO_PERTENENCIA);

```

#### **CLASE:**

**gov.sir.core.negocio.acciones.administracion.AnTrasladoTurno.java****Método:****private EvnRespTrasladoTurno  
consultarTurnoCertificadoCambioMatricula(EvnTrasladoTurno evento)  
throws EventoException {****Código Insertado Línea 966:**

```
/*
 * @author    : Julio Alcázar Rivas
 * @change   : Validaciones para realizar cambio de matriculas de turno de certificados
 * Caso Mantis : 07647
 */
CertificadosSIR certificadosSIR = new CertificadosSIR();

if (
(liquidacion.getIdTipoCertificado().equals(CTipoCertificado.TIPO_ANTIGUO_SISTEMA_ID)) ||
liquidacion.getIdTipoCertificado().equals(CTipoCertificado.TIPO_PERTENENCIA_ID))
&&
certificadosSIR.CreacionFolioHR(turno.getAnio(), turno.getIdCirculo(), turno.getIdProceso(),
turno.getIdTurno()) {

    throw new ConsultaTurnoExceptionException("no se permite cambio de matricula de certificados de
Antiguo Sistema o Pertenencia "
+ "que crearon el folio en Hoja de Ruta.");
}

List turnosHist = turno.getHistorials();
for (Iterator iter = turnosHist.iterator(); iter.hasNext();) {
    TurnoHistoria turnoHistoria = (TurnoHistoria) iter.next();
    if (turnoHistoria.getFase().equals(CFase.CER_AMPLIACION_TRADICION) &&
turnoHistoria.getRespuesta().equals(CRespuesta.AMPLIACION_TRADICION)){
        throw new ConsultaTurnoExceptionException("no se permite el cambio de
matricula, por que el turno de certificado incluyo información al folio.");
    }
    if (
((liquidacion.getIdTipoCertificado().equals(CTipoCertificado.TIPO_ANTIGUO_SISTEMA_ID)) ||
liquidacion.getIdTipoCertificado().equals(CTipoCertificado.TIPO_AMPLIACION_TRADICION_ID)) ||
liquidacion.getIdTipoCertificado().equals(CTipoCertificado.TIPO_PERTENENCIA_ID))
&&
(turnoHistoria.getFase().equals(CFase.ANT_REVISION_INICIAL) &&
turnoHistoria.getRespuesta().equals(CRespuesta.NEGAR)) {

        throw new ConsultaTurnoExceptionException("El certificado de tipo:
"+liquidacion.getNombre() + " fue negado por lo tanto no se puede cambiar la
matricula.");
    }

    if (liquidacion.getIdTipoCertificado().equals(CTipoCertificado.TIPO_ASOCIADO_ID))
{
    if (!certificadosSIR.CertificadoAsociadoVal(turno.getIdWorkflow())){
        throw new ConsultaTurnoExceptionException("no se permite cambio de matricula de
certificados asociados a registro con resolución devuelto");
    }
}
```

**Código Insertado Linea 1007:**

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : Validaciones para realizar cambio de matriculas de turno de certificados
 * Caso Mantis : 07647
 */
int numImpresionesCert = ((SolicitudCertificado) solicitud).getNumImpresiones();

ProcesoPk oid = new ProcesoPk();
oid.idProceso = Long.parseLong(CProceso.PROCESO_CERTIFICADOS);
Proceso proceso = hermod.getProcesoById(oid);

Rol rol = new Rol();
rol.setRoleId(CRoles.SIR_ROL_REIMPRESION_CERTIFICADOS);
int maxImprUsuario = hermod.getNumeroMaximoImpresiones(rol,proceso);

if (maxImprUsuario<=numImpresionesCert-1 && (turno.getIdFase().equals(CFase.CER_ENTREGAR)||turno.getIdFase().equals(CFase.FINALIZADO))){
    throw new ConsultaTurnoExceptionException("El turno de certificado ha alcanzado el número máximo("+maxImprUsuario+) de reimpresiones por lo tanto no se permite el cambio de matrícula");
}
```

**Código Insertado Linea 1040:**

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : Validaciones para realizar cambio de matriculas de turno de certificados
 * Caso Mantis : 07647
 */
if (numImpresionesCert>1){
    throw new ConsultaTurnoExceptionException("El turno de certificado ha sido reimpresso ("+(numImpresionesCert-1)+") por lo tanto no se permite el cambio de matrícula");
}
```

**CLASE:****gov.sir.core.web.acciones.administracion.AWTrasladoTurno.java****Método:**

```
private EvnCambioMatricula
cambiarFolioTurnoCertificado(HttpServletRequest request) throws
AccionWebException {
```

**Código Insertado Linea 637:**

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : a)Se verifica que el exista un folio asociado a la solicitud
 *             b)Se reubica el lugar donde se lanzan las excepciones
 * Caso Mantis : 07647
 */
String matriculaActual = null;
if(folio != null){
    matriculaActual = folio.getIdMatricula();

    if (matriculaActual.equals(matriculaNueva)){
        exception.addError("El numero de matrícula ingresado es el mismo que está asociado al turno");
    }
}
if (exception.getErrores().size() > 0) {
    throw exception;
}
```

**CLASE:**

**gov.sir.core.negocio.modelo.constantes.CFase.java****Código Insertado Línea 226:**

```
/* @author : Julio Alcázar Rivas
 * @change : identificador de la fase reimpresion del oficio de pertenencia
 * Caso Mantis : 07647
 */
public static final String CER_OFICIO_PERTENENCIA="CER_OFICIO_PERTENENCIA";
```

**247 Traslado de Folios V2****REQUERIMIENTO**

Revisado el despliegue 2-07-0 se encuentran las siguientes inconsistencias respecto del traslado de folios que deben ser corregidas:

- 1) El aplicativo no debe permitir generar impresiones simples del folio de matrícula con estado trasladado.
- 2) En la pantalla de resultado de traslado de folio se debe cambiar el texto del botón Terminar por “Regresar”.
- 3) Después de confirmar el trasladado del folio, el aplicativo muestra la pantalla del resultado del traslado, al dar clic en el botón Regresar debe llevar al usuario a la pantalla inicial de traslado de folios y debe mostrar los campos limpios (Folios de esta solicitud, resolución y Circulo destino).
- 4) Se requiere que en los reportes de traslados de folios de la oficina origen se incluya: a) Una columna con nombre “Usuario que realizo el traslado” allí se publicitara el nombre y apellido del usuario frente al folio trasladado. b) En los parámetros para generar el reporte de traslado de folios se debe eliminar “Circulo Destino”. c) En el reporte se debe agrupar por circulo destino y totalizar cada grupo. d) Agregar columna con el nombre “Resolución”, allí se muestra la información incluida en el campo resolución de la pantalla traslado de folios. e) Al final del reporte tendrá un total de matrículas trasladadas en el circulo origen y en el rango de fechas seleccionado. f) Debe mostrar los folios de matrícula inmobiliaria que han sido trasladados del sistema SIR al sistema Folio Magnético el orden de presentación en el reporte será primero los folios de SIR a SIR y luego el bloque de SIR a Folio Magnético.
- 5) Se requiere crear un reporte de traslado de folios para la Oficina destino, con el nombre de Folios trasladados a este círculo cuyos parámetros para generarlos son: rango de fecha( fecha inicial fecha final), debe traer predeterminado el nombre y código del Circulo destino, con las siguientes columnas: Fecha de traslado, Resolución (información incluida en el campo de Resolución en la Orip Origen), Circulo Origen, Matricula Orip Origen, Dirección o Nombre del Predio , Matricula Orip destino , debe agrupar la información por círculos en estricto orden de fecha de traslado .y totalizar cada grupo y mostrar un gran total .
- 6) Si un folio de matrícula tiene(n) una(s) anotación(es) y ciudadanos identificados con secuencia que tienen antepuesto el ID del circulo (ID que fue incluido desde la migración), al trasladar el folio el aplicativo debe anteponer el ID del circulo destino a la secuencia asignada al ciudadano, en el folio destino. Este cambio debe guardarse en el historial del folio.
- 7) Teniendo en cuenta que al crear un ID\_CIUDADANO en la ORIP destino, se pierden los datos de identificación, es importante, que en la ORIP destino se tenga almacenado en algún campo, esta información. Se solicita guardar estos datos en el campo ntcid\_descripción, de la tabla SIR\_NE\_ANOTACION\_CIUDADANO
- 8) En la oficina Origen, al consultar un folio de matrícula inmobiliaria cuyo estado es trasladado, a través de la pantalla administrativa Ver detalles de folio, el aplicativo

<p>debe mostrar el nombre y código del círculo destino en el mismo campo donde se visualiza el estado del folio así: “Trasladado al círculo (Ejemp: Bucaramanga -300)” el nombre y código del círculo destino solo se visualiza en esta pantalla.</p> <p>9) En el Círculo destino al consultar un número de matrícula inmobiliaria asignada a un folio trasladado de otra Orip, el aplicativo debe mostrar una alerta informativa permanente indicando el círculo de Origen del cual fue trasladado, el número de matrícula de la oficina de origen y número y fecha de la resolución oficina origen y número y fecha de la resolución oficina Destino, esta información solo debe visualizarse en la consulta de folio mas no debe mostrarse en el certificado de tradición.</p> <p>10) Traslado de Folios de matrícula del Aplicativo SIR a FOLIO MAGNETICO: En la pantalla administrativa traslado de folios se debe:</p> <ul style="list-style-type: none"> <li>• agregar un campo obligatorio donde se seleccione si el traslado corresponde a una oficina que opera con SIR o con FOLIO.</li> <li>• En la lista desplegable asociada al campo “Círculo” debe mostrar todos los Círculos Registrales.</li> <li>• En el campo “Número de matrícula” relacionada con el nuevo número que se dará al folio trasladado para el caso de los trasladados de SIR a FOLIO, en este campo se debe asignar por defecto el texto “FOLIO&lt;CIRCULO&gt; donde Círculo corresponde al Círculo seleccionado. Este dato debe almacenarse en la Tabla SIR_NE_TRASLADO en el campo ID_MATRICULA_DESTINO.</li> </ul> <p>Como no se asigna un nuevo número de matrícula en los trasladados de SIR a FOLIO, el sistema solo debe generar el formato XML la información del folio que se pasa como parámetro. No se generaran secuencias de matrícula y ciudadanos ya que no hay conexión con la base de datos de folio magnético.</p> <p>11) La tabla sir_ne_salvedad_an, al trasladar al círculo destino debe complementar los campos id_nanotacion y id_nsaledad_an. Es decir toda la información de las tablas debe ser trasladada de un círculo a otro sin alterar ni omitir datos.</p> <p>12) Modificar el estado de los folios y anotaciones a trasladar de la ORIP origen, con el estado “T- Trasladado” y garantizar que no se puedan realizar trámites ni consultas sobre estos folios, es decir que tanto el estado del folio como de sus anotaciones deben quedar en estado trasladado. Incluir este nuevo estado en las tablas que identifican el estado de las anotaciones.</p> <p>13) Se requiere que cuando se de clic en el botón trasladar, el aplicativo debe mostrar un mensaje de confirmación al usuario “Esta seguro de trasladar la matrícula al círculo (Nombre del Círculo resaltado en tamaño de letra)”. Si la acción de trasladar es afirmativa, en la Orip destino se generara un bloqueo para el número de matrícula asignado al folio trasladado, el cual se mantendrá hasta que se confirme el traslado en la orip destino, esto para los trasladados entre oficinas que operan con SIR .</p> <p>14) En la oficina destino se requiere que se ajuste la pantalla de traslado de folios, incluyendo una sección donde le muestre la lista de matrículas inmobiliarias asignadas por la ORIP destino a los folios que llegaron trasladados, con un vínculo que permita consultar la información de la pantalla administrativa “Ver detalles de folio”, al seleccionar una matrícula mostrará la información de solo lectura de los siguientes campos: Círculo, Departamento, Municipio, Vereda, fecha de apertura, radicación, fundamento de la apertura (con), código Catastral y estado del folio.</p> <p>El aplicativo debe habilitar los siguientes campos de selección: Departamento, Municipio, Vereda y permitir editarlos y aceptar la modificación de los campos correspondientes sin salvedad alguna; los cambios se deben guardar en el historial del</p>	
---	--

folio seleccionado, el ID del usuario que realizo el cambio, fecha y la IP. Es optativa la modificación de estos campos.

Así mismo en esta pantalla el aplicativo debe mostrar un Titulo que diga “Resolución destino, Un campo de 20 caracteres para agregar el número de la resolución, siguiente campo para ingresar la fecha de la resolución con icono de calendario, este campo es obligatorio para todas las matriculas trasladadas en la Oficina destino.

El aplicativo permitirá mediante un botón la confirmación, al seleccionar este botón confirma el recibido del folio.

14) Mientras no se realice la confirmación del acto administrativo el aplicativo no desbloqueara la matricula para realizar trámites de los procesos registrales; en tal caso en la Orip destino al consultar o agregar una matrícula asignada a un folio trasladado de otra Orip el aplicativo mostrara un mensaje restrictivo “Folio pendiente de confirmar traslado de otra Orip”.

15) En lo referente al código catastral, la información de este campo se guardara en FLIO\_COD\_CATASTRAL\_ANT de la tabla sir\_ne\_folio, para mantener el historial y dejar el campo FLIO\_COD\_CATASTRAL nulo. De existir información en el campo FLIO\_COD\_CATASTRAL\_ANT, el sistema debe guardar en las tablas de auditoría esta información y proceder a guardar lo que tiene el campo FLIO\_COD\_CATASTRAL al campo FLIO\_COD\_CATASTRAL\_ANT.

16) Con respecto al estado, el folio debe trasladarse con el estado ORIGINAL antes del traslado, es decir, si estaba activo en la ORIP destino debe pasar como activo y si es cerrado de igual manera.

17) En la pantalla de traslado de folios sección Oficina Origen debe ajustarse el campo resolución así: Titulo que diga “Resolución Origen, Un campo de 20 caracteres para agregar el número de la resolución, siguiente campo para ingresar la fecha de la resolución con icono de calendario, este campo es obligatorio.

## **DESARROLLO**

### **MODIFICACION GeneralSIR**

#### **MODIFICACION SIR-APP**

- Script Bases de Datos:
  - Ninguno
- Creacion De Imagenes:
  - Ninguno
- Se crearon los siguientes archivos:
  - WEB-INF.jspSeguro.administracion. TRASLADO\_CONFIRMACION.jsp
- Se modificaron los siguientes archivos:
  - gov.sir.impl.jdo
  - org.auriga.smart.vistas.xml
  - org.auriga.smart.web.xml
- Se cambio sustancialmente el comportamiento de los siguientes archivos:
  - WEB-INF.jspSeguro.administracion. SOLICITUD\_TRASLADO.jsp

#### **ARCHIVO JSP:**

**WEB-INF.jspSeguro.administracion.CONSULTA\_FOLIO\_PRINCIPAL.jsp**

#### **Código Insertado Linea 86:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Variables para manejar la nueva informacion de Ver detalles
```

```

* Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
*/
Traslado trasladoOrigen = (Traslado) session.getAttribute(WebKeys.TRASLADO_ORIGEN);
Traslado trasladoDestino = (Traslado) session.getAttribute(WebKeys.TRASLADO_DESTINO);
Circulo circuloOrigen = (Circulo) session.getAttribute(WebKeys.CIRCULO_ORIGEN);
Circulo circuloDestino = (Circulo) session.getAttribute(WebKeys.CIRCULO_DESTINO);

```

#### Código Insertado Linea 208:

```

/*
 * @author    : Julio Alcázar Rivas
 * @change    : Variables para manejar la nueva informacion de Ver detalles
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
mFolio.setTrasladoDestino(trasladoDestino);
mFolio.setTrasladoOrigen(trasladoOrigen);
mFolio.setCirculoDestino(circuloDestino);
mFolio.setCirculoOrigen(circuloOrigen);

```

**WEB-INF.jspSeguro.administracion.TRASLADO\_FINAL.jsp**

#### Código Linea 157:

```

<a href="<%={request.getContextPath()}%>/admin.view"></a>

```

**por:**

```

<!--
@author    : Julio Alcázar Rivas
@change    : Cambio del comportamiento del boton
Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
!-->
<a href="<%={request.getContextPath()}%>/admin.traslado.view"></a>

```

**CLASE:**

**gov.sir.core.web.acciones.administracion.AWAdministracionForseti.java**

#### Código Insertado Linea 444:

```

/*
 * @author    : Julio Alcázar Rivas
 * @change    : nuevas constantes del proceso traslados
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
public static final String CIRCULOS_NOSIR= "CIRCULOS_NOSIR";
public static final String CIRCULO_OFICINA= "CIRCULO_OFICINA";
public static final String FOLIO_CONFIRMACION = "FOLIO_CONFIRMACION";
public static final String RESOLUCION_DESTINO = "RESOLUCION_DESTINO";
public static final String TRASLADO_CONFIRMACION = "TRASLADO_CONFIRMACION";

```

**Método:**

**public Evento perform(HttpServletRequest request) throws  
AccionWebException {**

#### Código Insertado Linea 675:

```

/*
 * @author    : Julio Alcázar Rivas
 * @change    : nuevos procesos de traslados
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
else if (accion.equals(AWAdministracionForseti.FOLIO_CONFIRMACION)) {
    return folioConfirmacion(request);
} else if (accion.equals(AWAdministracionForseti.TRASLADO_CONFIRMACION)) {
    return trasladoConfirmacion(request);
}

```

<p><b>Método:</b></p> <pre><b>private EvnAdministracionForseti trasladar(HttpServletRequest request) throws AccionWebException {</b></pre>
<pre>* @author : Julio Alcázar Rivas * @change : modificaciones sustanciales al comportamiento del metodo * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2 */</pre>
<p><b>Método:</b></p> <pre><b>public void doEnd(HttpServletRequest request, EventoRespuesta evento) {</b></pre>
<p><b>Código Insertado Linea 4590:</b></p> <pre>/* * @author : Julio Alcázar Rivas * @change : elimina atributos de la session * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2 */ session.removeAttribute(CIRCULOS_NOSIR); session.removeAttribute(CIRCULO_OFICINA);</pre>
<p><b>Código Insertado Linea 4697:</b></p> <pre>/* * @author : Julio Alcázar Rivas * @change : manejo de la respuesta en los procesos de traslado * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2 */ else if (tipoEvento.equals(EvnRespAdministracionForseti.FOLIO_CONFIRMACION)) {     Folio resultado = (Folio) respuesta.getPayload();      session.setAttribute(AWAdministracionForseti.FOLIO_CONFIRMACION, resultado);     return; } else if (tipoEvento.equals(EvnRespAdministracionForseti.TRASLADO_CONFIRMACION)) {     return; }</pre>
<p><b>Nuevo Método:</b></p> <pre><b>private EvnAdministracionForseti folioConfirmacion(HttpServletRequest request) throws ValidacionTrasladoException {</b></pre>
<pre>/* * @author : Julio Alcázar Rivas * @change : nuevo metodo folioConfirmacion() * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2 */</pre>
<p><b>Nuevo Método:</b></p> <pre><b>private EvnAdministracionForseti trasladoConfirmacion(HttpServletRequest request) throws ValidacionTrasladoException {{</b></pre>
<pre>/* * @author : Julio Alcázar Rivas * @change : nuevo metodo trasladoConfirmacion() * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2 */</pre>
<p><b>CLASE:</b></p> <pre><b>gov.sir.core.negocio.acciones.administracion.ANAdministracionForseti.jav a</b></pre>
<p><b>Método:</b></p> <pre><b>public EventoRespuesta perform(Evento e) throws EventoException {</b></pre>

**Código Insertado Linea 290:**

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : nuevos procesos traslado
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
else if (tipoEvento.equals(EvnAdministracionForseti.FOLIO_CONFIRMACION)) {
    return folioConfirmacion(evento);
} else if (tipoEvento.equals(EvnAdministracionForseti.TRASLADO_CONFIRMACION)) {
    return trasladoConfirmacion(evento);
}
```

**Método:**

```
private EvnRespAdministracionForseti trasladar(EvnAdministracionForseti
evento)
throws EventoException {
```

**Código Insertado Linea 2145:**

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : nuevas variables para ejecutar el traslado
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
String tipo_circulo = evento.getTipo_circulo();
Date fecharesolucionorigen = evento.getFechaResolucionOrigen();
```

**Cambio Linea 2157:**

```
matricula = traslado.TrasladoFolio(matriculaorigen, cirorigen, cirdestino,resolucion,idUsuario);
```

**Por:**

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : cambios parametros del metodo TrasladoFolio()
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
matricula = traslado.TrasladoFolio(matriculaorigen, cirorigen,
cirdestino,resolucion,fecharesolucionorigen,idUsuario,tipo_circulo);
```

**Nuevo Método:**

```
private EventoRespuesta folioConfirmacion(EvnAdministracionForseti
evento) throws EventoException {
```

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : nuevo metodo folioConfirmacion()
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
```

**Nuevo Método:**

```
private EventoRespuesta trasladoConfirmacion(EvnAdministracionForseti
evento) throws EventoException {
```

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : nuevo metodo trasladoConfirmacion()
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
```

**CLASE:**

```
gov.sir.core.negocio.acciones.administracion.ANConsultaFolio.java
```

**Método:**

```
private EventoRespuesta consultaFolioDefinitivo(EvnConsultaFolio
```

```
evento) throws EventoException {
```

#### Código Insertado Linea 795:

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : Consulta de informacion de traslado para ver detalles de folio
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
Traslado trasladoOrigen = hermod.getFolioOrigenTraslado(matricula);
Traslado trasladoDestino = hermod.getFolioDestinoTraslado(matricula);
Circulo circuloorigen = null;
Circulo circulodestino = null;
if(trasladoOrigen != null){
    CirculoPk circuloPk = new CirculoPk(trasladoOrigen.getFolioDestino().getCirculo());
    circulodestino = forseti.getCirculo(circuloPk);
}
if(trasladoDestino != null){
    CirculoPk circuloPk = new CirculoPk(trasladoDestino.getFolioOrigen().getCirculo());
    circuloorigen = forseti.getCirculo(circuloPk);
}
```

#### Código Insertado Linea 820:

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : Set de informacion de traslado para ver detalles de folio
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
resp.setTrasladoDestino(trasladoDestino);
resp.setTrasladoOrigen(trasladoOrigen);
resp.setCirculoDestino(circulodestino);
resp.setCirculoOrigen(circuloorigen);
```

**CLASE:**

**gov.sir.core.negocio.acciones.administracion.ANImprimirFolio.java**

**Método:**

**private EvnRespImprimirFolio imprimirFolio(EvnImprimirFolio evento)**  
**throws EventoException {**

#### Código Insertado Linea 615:

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : Bloque la posibilidad de imprimir folios en estado trasladado
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
if (forseti.trasladadoFolio(matricula)) {
    throw new EventoException("El folio identificado con matrícula:" + matricula + " ha sido
trasladado.");
}
```

**CLASE:**

**gov.sir.core.web.acciones.administracion.AWConsultaFolio.java**

**Método:**

**public void doEnd(HttpServletRequest request, EventoRespuesta evento)**  
**{**

#### Código Insertado Linea 449:

```
/*
 * @author    : Julio Alcázar Rivas
 * @change    : Set atributos en la session
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
session.setAttribute(WebKeys.TRASLADO_DESTINO, respuesta.getTrasladoDestino());
session.setAttribute(WebKeys.TRASLADO_ORIGEN, respuesta.getTrasladoOrigen());
```

<pre>session.setAttribute(WebKeys.CIRCULO_DESTINO, respuesta.getCirculoDestino()); session.setAttribute(WebKeys.CIRCULO_ORIGEN, respuesta.getCirculoOrigen());</pre> <p><b>CLASE:</b>  <b>gov.sir.core.web.navegacion.administracion.ControlNavegacionAdministracionForseti.java</b></p> <p><b>Método:</b></p> <pre><b>public String procesarNavegacion(HttpServletRequest request) throws ControlNavegacionException {</b></pre> <p><b>Código Insertado Línea 241:</b></p> <pre>/*  * @author : Julio Alcázar Rivas  * @change : Manejo de la navegacion  * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2  */ else if(accion.equals(AWAdministracionForseti.FOLIO_CONFIRMACION)){     return AWAdministracionForseti.FOLIO_CONFIRMACION; }else if(accion.equals(AWAdministracionForseti.TRASLADO_CONFIRMACION)){     return AWAdministracionForseti.TRASLADO_CONFIRMACION; }</pre> <p><b>CLASE:</b>  <b>gov.sir.core.eventos.administracion.EvnAdministracionForseti.java</b></p> <p><b>Código Insertado Línea 209:</b></p> <pre>/*  * @author : Julio Alcázar Rivas  * @change : constantes para el proceso traslado folio  * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2  */ public static final String FOLIO_CONFIRMACION = "FOLIO_CONFIRMACION"; public static final String TRASLADO_CONFIRMACION = "TRASLADO_CONFIRMACION";</pre> <p><b>Código Insertado Línea 293:</b></p> <pre>/*  * @author : Julio Alcázar Rivas  * @change : nuevos atributos para el manejo del proceso traslado folio  * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2  */ private Date fechaResolucionOrigen;  private Date fechaResolucionDestino;  private String resolucionDestino;  private String tipo_circulo;  public Date getFechaResolucionDestino() {     return fechaResolucionDestino; }  public void setFechaResolucionDestino(Date fechaResolucionDestino) {     this.fechaResolucionDestino = fechaResolucionDestino; }  public Date getFechaResolucionOrigen() {     return fechaResolucionOrigen; }  public void setFechaResolucionOrigen(Date fechaResolucionOrigen) {     this.fechaResolucionOrigen = fechaResolucionOrigen; }</pre>	
---	--

```

public String getResolucionDestino() {
    return resolucionDestino;
}

public void setResolucionDestino(String resolucionDestino) {
    this.resolucionDestino = resolucionDestino;
}

public String getTipo_circulo() {
    return tipo_circulo;
}

public void setTipo_circulo(String tipo_circulo) {
    this.tipo_circulo = tipo_circulo;
}

```

**CLASE:**

**gov.sir.core.eventos.administracion.EvnRespAdministracionForseti.java**

**Código Insertado Línea 177:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : constantes para el proceso traslado folio
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
public static final String FOLIO_CONFIRMACION = "FOLIO_CONFIRMACION";
public static final String TRASLADO_CONFIRMACION = "TRASLADO_CONFIRMACION";

```

**CLASE:**

**gov.sir.core.eventos.administracion.EvnRespConsultaFolio.java**

**Código Insertado Línea 54:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : nuevos atributos para el manejo del proceso traslado folio
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
private Traslado trasladoDestino;
private Traslado trasladoOrigen;
Circulo circuloDestino;
Circulo circuloOrigen;

public Traslado getTrasladoDestino() {
    return trasladoDestino;
}

public void setTrasladoDestino(Traslado trasladoDestino) {
    this.trasladoDestino = trasladoDestino;
}

public Traslado getTrasladoOrigen() {
    return trasladoOrigen;
}

public void setTrasladoOrigen(Traslado trasladoOrigen) {
    this.trasladoOrigen = trasladoOrigen;
}

public Circulo getCirculoDestino() {
    return circuloDestino;
}

public void setCirculoDestino(Circulo circuloDestino) {

```

```

        this.circuloDestino = circuloDestino;
    }

    public Circulo getCirculoOrigen() {
        return circuloOrigen;
    }

    public void setCirculoOrigen(Circulo circuloOrigen) {
        this.circuloOrigen = circuloOrigen;
    }

```

**CLASE:**

**gov.sir.hermode.HermodService.java**

**Código Insertado Línea 9696:**

```

    /**
     * @author : Julio Alcazar
     * @change : Se consulta la informacion origen de una matricula trasladada
     * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
     */
    public Traslado getFolioOrigenTraslado(String matricula) throws HermodException {
        try {
            return factory.getTrasladoTurnosDAO().getFolioOrigenTraslado(matricula);
        } catch (DAOException e) {
            HermodException he = new HermodException("Fallo en el servicio para obtener la informacion
de traslado: ", e);
            he.setHashErrores(e.getHashErrores());
            Log.getInstance().error(HermodService.class, he.getMessage(), e);
            throw he;
        }
    }

    /**
     * @author : Julio Alcazar
     * @change : Se consulta la informacion destino de una matricula trasladada
     * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
     */
    public Traslado getFolioDestinoTraslado(String matricula) throws HermodException {
        try {
            return factory.getTrasladoTurnosDAO().getFolioDestinoTraslado(matricula);
        } catch (DAOException e) {
            HermodException he = new HermodException("Fallo en el servicio para obtener la informacion
de traslado: ", e);
            he.setHashErrores(e.getHashErrores());
            Log.getInstance().error(HermodService.class, he.getMessage(), e);
            throw he;
        }
    }

```

**CLASE:**

**gov.sir.hermode.ejb.HermodServiceEJBBean.java**

**Código Insertado Línea 3781:**

```

    /**
     * @see gov.sir.forseti.interfaz.ForsetiServiceInterface#getFolioDestinoTraslado(java.lang.String
     * matricula)
     * @ejb:interface-method view-type="both"
     * @author : Julio Alcazar
     * @change : Se consulta la informacion destino de una matricula trasladada
     * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2

```

```

*/
public Traslado getFolioDestinoTraslado(String matricula) throws HermodException{
    return hermod.getFolioDestinoTraslado(matricula);
}

/**
 * @see gov.sir.forseti.interfaz.ForsetiServiceInterface#getFolioOrigenTraslado(java.lang.String
matricula)
 * @ejb:interface-method view-type="both"
 * @author : Julio Alcazar
 * @change : Se consulta la informacion destino de una matricula trasladada
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
public Traslado getFolioOrigenTraslado(String matricula) throws HermodException{
    return hermod.getFolioOrigenTraslado(matricula);
}

```

**CLASE:**

**gov.sir.hermod.interfaz.HermodServiceInterface.java**

**Código Insertado Línea 4627:**

```

/**
 * @author : Julio Alcazar
 * @change : Se consulta la informacion destino de una matricula trasladada
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
public Traslado getFolioDestinoTraslado(String matricula) throws Throwable;

/**
 * @author : Julio Alcazar
 * @change : Se consulta la informacion origen de una matricula trasladada
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
public Traslado getFolioOrigenTraslado(String matricula) throws Throwable;

```

**CLASE:**

**gov.sir.hermod.dao.impl.jdogenie.JDOTrasladoTurnosDAO.java**

**Código Insertado Línea 46:**

```

/**
 * @author : Julio Alcazar
 * @change : Se consulta la informacion origen de una matricula trasladada
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
public Traslado getFolioOrigenTraslado(String matricula) throws DAOException {
    PersistenceManager pm = AdministradorPM.getPM();

    TrasladoEnhanced rta = null;
    Traslado tras = null;
    try {
        Query query = pm.newQuery(TrasladoEnhanced.class);
        query.declareParameters("String matricula");
        query.setFilter("folioOrigen.idMatricula == matricula");
        Collection col = (Collection) query.execute(matricula);
        Iterator iter = col.iterator();
        while (iter.hasNext()) {
            rta = (TrasladoEnhanced) iter.next();
        }

        query.close(col);
        if(rta != null){
            tras = this.getTraslado(rta, pm);
        }
    }
}
```

```

} catch (JDOObjectNotFoundException e) {
    Log.getInstance().error(JDOTrasladoTurnosDAO.class,e.getMessage());
    throw new DAOException(e.getMessage(), e);
} catch (JDOException e) {
    Log.getInstance().error(JDOTrasladoTurnosDAO.class,e.getMessage());
    throw new DAOException(e.getMessage(), e);
}finally {
    pm.close();
}
return tras;
}

/**
* @author    : Julio Alcazar
* @change   : Se consulta la informacion destino de una matricula trasladada
* Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
*/
public Traslado getFolioDestinoTraslado(String matricula) throws DAOException {
    PersistenceManager pm = AdministradorPM.getPM();
    TrasladoEnhanced rta = null;
    Traslado tras = null;

    try {
        Query query = pm.newQuery(TrasladoEnhanced.class);
        query.declareParameters("String matricula");
        query.setFilter("folioDestino.idMatricula == matricula");
        Collection col = (Collection) query.execute(matricula);
        Iterator iter = col.iterator();
        while (iter.hasNext()) {
            rta = (TrasladoEnhanced) iter.next();
        }

        query.close(col);
        if(rta != null){
            tras = this.getTraslado(rta, pm);
        }
    }

    } catch (JDOObjectNotFoundException e) {
        Log.getInstance().error(JDOTrasladoTurnosDAO.class,e.getMessage());
        throw new DAOException(e.getMessage(), e);
    } catch (JDOException e) {
        Log.getInstance().error(JDOTrasladoTurnosDAO.class,e.getMessage());
        throw new DAOException(e.getMessage(), e);
    }finally {
        pm.close();
    }

    return tras;
}

private Traslado getTraslado(TrasladoEnhanced trasladoEnhanced, PersistenceManager pm){

    Traslado traslado = new Traslado();
    traslado.setResolucion(trasladoEnhanced.getResolucion());
    traslado.setFechaTraslado(trasladoEnhanced.getFechaTraslado());
    traslado.setIdTraslado(trasladoEnhanced.getIdTraslado());
    traslado.setResolucionDestino(trasladoEnhanced.getResolucionDestino());
    traslado.setFechaResolucionDestino(trasladoEnhanced.getFechaResolucionDestino());
    traslado.setFechaResolucionOrigen(trasladoEnhanced.getFechaResolucionOrigen());
    FolioEnhanced folioorigen = trasladoEnhanced.getFolioOrigen();
    FolioEnhanced foliodestino = trasladoEnhanced.getFolioDestino();
    pm.makeTransient(trasladoEnhanced);
}

```

```

Folio folio1 = new Folio();
Folio folio2 = new Folio();
folio1.setIdMatricula(folioorigen.getIdMatricula());
folio1.setCirculo(folioorigen.getCirculo());
folio2.setIdMatricula(foliodes destino.getIdMatricula());
folio2.setCirculo(foliodes destino.getCirculo());

traslado.setFolioOrigen(folio1);
traslado.setFolioDestino(folio2);

return traslado;
}

```

**CLASE:**

**gov.sir.core.web.helpers.comun.MostrarFolioHelper.java**

**Código Insertado Línea 315:**

```

/**
 * @author    : Julio Alcazar
 * @change   : variables para los nuevos datos en ver detalles de folio
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
private Traslado trasladoDestino;
private Traslado trasladoOrigen;
Circulo circuloDestino;
Circulo circuloOrigen;

public Traslado getTrasladoDestino() {
    return trasladoDestino;
}

public void setTrasladoDestino(Traslado trasladoDestino) {
    this.trasladoDestino = trasladoDestino;
}

public Traslado getTrasladoOrigen() {
    return trasladoOrigen;
}

public void setTrasladoOrigen(Traslado trasladoOrigen) {
    this.trasladoOrigen = trasladoOrigen;
}

public Circulo getCirculoDestino() {
    return circuloDestino;
}

public void setCirculoDestino(Circulo circuloDestino) {
    this.circuloDestino = circuloDestino;
}

public Circulo getCirculoOrigen() {
    return circuloOrigen;
}

public void setCirculoOrigen(Circulo circuloOrigen) {
    this.circuloOrigen = circuloOrigen;
}

```

**public void drawEncabezado(HttpServletRequest request1, JspWriter out)  
throws IOException, HelperException {**

**Código Insertado Línea 1027:**

```

/**
 * @author    : Julio Alcazar
 * @change    : nuevo metodo en ver detalles de folio
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
if(this.trasladoDestino != null){
    this.drawDatosRelevantesTraslado(request1, out);
}

public void drawAperturaFolio(HttpServletRequest request1, JspWriter out) throws IOException, HelperException {

```

**Cambio Código Línea 2036:**

```

out.println("      <td width=\"75%\" class=\""+this.eCampoTexto+"\"><strong>" +
(folio.getEstado().getNombre() != null ? folio.getEstado().getNombre() : " ") +
"</strong></td>");

```

**Por:**

```

/**
 * @author    : Julio Alcazar
 * @change    : Cambios en el estado ver detalles de folio
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
out.println("      <td width=\"75%\" class=\""+this.eCampoTexto+"\"><strong>" +
(folio.getEstado().getNombre() != null ? (trasladoOrigen != null ? folio.getEstado().getNombre() +
"+circuloDestino.getNombre() + "-" + circuloDestino.getIdCirculo() : folio.getEstado().getNombre()) :
" ") + "</strong></td>");

```

**NUEVO METODO public void  
drawDatosRelevantesTraslado(HttpServletRequest request1, JspWriter  
out)  
throws IOException, HelperException {**

**Cambio Código Línea 3998:**

```

/**
 * @author    : Julio Alcazar
 * @change    : nuevo metodo en ver detalles de folios trasladados
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
if(this.trasladoDestino != null){

    this.drawDatosRelevantesTraslado(request1, out);

}

```

**CLASE:**

**gov.sir.core.negocio.modelo.Traslado.java**

**Código Insertado Línea 17:**

```

/**
 * @author    : Julio Alcazar
 * @change    : nuevas propiedades de el objeto
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
private String resolucionDestino;
private Date fechaResolucionOrigen;
private Date fechaResolucionDestino;

public Date getFechaResolucionDestino() {
    return fechaResolucionDestino;
}

public void setFechaResolucionDestino(Date fechaResolucionDestino) {
    this.fechaResolucionDestino = fechaResolucionDestino;
}

```

```

public Date getFechaResolucionOrigen() {
    return fechaResolucionOrigen;
}

public void setFechaResolucionOrigen(Date fechaResolucionOrigen) {
    this.fechaResolucionOrigen = fechaResolucionOrigen;
}

public String getResolucionDestino() {
    return resolucionDestino;
}

public void setResolucionDestino(String resolucionDestino) {
    this.resolucionDestino = resolucionDestino;
}

```

**CLASE:**

**gov.sir.core.negocio.modelo.jdogenie.TrasladoEnhanced.java**

**Código Insertado Línea 17:**

```

/**
 * @author : Julio Alcazar
 * @change : nuevas propiedades de el objeto
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
private String resolucionDestino;
private Date fechaResolucionOrigen;
private Date fechaResolucionDestino;

public Date getFechaResolucionDestino() {
    return fechaResolucionDestino;
}

public void setFechaResolucionDestino(Date fechaResolucionDestino) {
    this.fechaResolucionDestino = fechaResolucionDestino;
}

public Date getFechaResolucionOrigen() {
    return fechaResolucionOrigen;
}

public void setFechaResolucionOrigen(Date fechaResolucionOrigen) {
    this.fechaResolucionOrigen = fechaResolucionOrigen;
}

public String getResolucionDestino() {
    return resolucionDestino;
}

public void setResolucionDestino(String resolucionDestino) {
    this.resolucionDestino = resolucionDestino;
}

```

**CLASE:**

**gov.sir.hermod.dao.TrasladoTurnosDAO.java**

**Código Insertado Línea 24:**

```

/**
 * Servicio que permite conocer la historia de traslados de un folio
 * @param matricula Identificación del folio del cual se desea conocer la historia de traslados
 * @return Objeto <code>TrasladoEnhanced</code> que contiene la historia de traslados del folio
 * @throws <code>DAOException</code>
 * @author : Julio Alcazar

```

```

* @change      : nuevos metodos de traslado folios
* Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
*/
public Traslado getFolioOrigenTraslado(String matricula) throws DAOException;
/**
 * Servicio que permite conocer la historia de traslados de un folio
 * @param matricula Identificación del folio del cual se desea conocer la historia de traslados
 * @return Objeto <code>TrasladoEnhanced</code> que contiene la historia de traslados del folio
 * @throws <code>DAOException</code>
 * @author      : Julio Alcazar
 * @change      : nuevos metodos de traslado folios
* Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
*/
public Traslado getFolioDestinoTraslado(String matricula) throws DAOException;

```

**CLASE:**

**gov.sir.core.negocio.modelo.transferutil. TrasladoTransferCopier.java**

**Método:**

```

public static org.auriga.core.modelo.TransferObject deepCopy(
gov.sir.core.negocio.modelo.jdogenie.TrasladoEnhanced enhancedObject,
java.util.HashMap cache)

```

**Código Insertado Línea 75:**

```

/**
 * @author      : Julio Alcazar
 * @change      : nuevas propiedades de el objeto
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
*/

// Resolucion Destino
try {
    transferObject.setResolucionDestino(enhancedObject.getResolucionDestino());
} catch (Throwable th) {}

// Fecha Resolucion Origen
try {
    if (enhancedObject.getFechaResolucionOrigen() != null)
    {
        transferObject.setFechaResolucionOrigen(new
Date(enhancedObject.getFechaResolucionOrigen().getTime()));
    }
} catch (Throwable th) {}

// Fecha Resolucion Destino
try {
    if (enhancedObject.getFechaResolucionDestino() != null)
    {
        transferObject.setFechaResolucionDestino(new
Date(enhancedObject.getFechaResolucionDestino().getTime()));
    }
} catch (Throwable th) {}

```

**Método:**

```

public static gov.sir.core.negocio.modelo.jdogenie Enhanced deepCopy(
gov.sir.core.negocio.modelo.Traslado transferObject, java.util.HashMap
cache)

```

**Código Insertado Línea 168:**

```

/**
 * @author      : Julio Alcazar
 * @change      : nuevas propiedades de el objeto
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
*/

```

```

// Resolucion Destino
try {
enhancedObject.setResolucionDestino(transferObject.getResolucionDestino());
} catch (Throwable th ) {}

// Fecha Resolucion Origen
try {
if (transferObject.getFechaResolucionOrigen() != null)
{
enhancedObject.setFechaResolucionOrigen(new
Date(transferObject.getFechaResolucionOrigen().getTime()));
}
} catch (Throwable th ) {}

// Fecha Resolucion Destino
try {
if (transferObject.getFechaResolucionDestino() != null)
{
enhancedObject.setFechaResolucionDestino(new
Date(transferObject.getFechaResolucionDestino().getTime()));
}
} catch (Throwable th ) {}

```

**CLASE:**

**gov.sir.core.web.WebKeys.java**

**Código Insertado Línea 1220:**

```

/**
 * @author : Julio Alcazar
 * @change : Se declaran las constantes para la navegacion en el proceso traslado folio
 * Caso Mantis : 0007676: Acta - Requerimiento No 247 - Traslado de Folios V2
 */
public static final String TRASLADO_ORIGEN = "TRASLADO_ORIGEN";
public static final String TRASLADO_DESTINO = "TRASLADO_DESTINO";
public static final String CIRCULO_ORIGEN = "CIRCULO_ORIGEN";
public static final String CIRCULO_DESTINO = "CIRCULO_DESTINO";
public static final String LISTADO_CIRCULOS_NOSIR = "LISTADO_CIRCULOS_NOSIR";

```

## 012 Reimpresión de certificados validaciones

### **REQUERIMIENTO**

#### **REIMPRESIÓN DE CERTIFICADOS DE LIBERTAD**

Por Disposición de la Resolución 4907 del 30 de junio de 2009 Articulo 12, de la SNR , “Queda prohibida la resolución de certificados de libertad sin la debida justificación. La nueva expedición que se origine por error de digitación del cajero o por fallas en el sistema automatizado, deberá ordenarse exclusivamente por el Coordinador del Grupo Operativo o por quien haga sus veces en donde no exista dicho cargo, una vez se acredite la necesidad de la misma, dejando constancia de este hecho en el conformador de recaudos o boletín diario de caja“

Para dar cumplimiento a lo ordenado en la resolución citada anteriormente, se requiere que en el rol de entrega de certificados se suprima la opción de reimpresión de certificados y la opción “Reimpresión de Certificados, Reimpresión de certificados de Pertenencia y cambio de matrícula Turno de Certificado” que se encuentran en las pantallas administrativas y el rol Reimpresión Especial, es decir que se suprima todas las opciones de reimpresión de Certificados del Aplicativo SIR. Así mismo para que el aplicativo SIR permita asignar la opción de reimpresión de certificados al usuario indicado, se debe crear el ROL REIMPRESIÓN DE CERTIFICADOS con las siguientes especificaciones :

En la pantalla creada para tal fin el sistema debe permitir ingresar el numero del turno, Seleccionar Impresora , Numero de copias a imprimir , Causal de Reimpresión obligatoria- lista de

selección “CERTIFICADO NO GENERADO, FALLO IMPRESORA, FALLO PAPEL ,OTRA CAUSAL DE REIMPRESIÓN “ y un campo de texto para descripción motivo de reimpresión.

Igualmente el aplicativo debe permitir cambiar el numero de matricula inicialmente digitado para el turno de certificado , por una sola vez, para tal efecto al seleccionar cambio de matricula debe mostrar un campo de solo lectura con el numero de la matricula que trae el turno y habilitarse un campo para ingresar el nuevo numero de matricula , obligando a ingresar la justificación del cambio de matricula o con la opción de agregar nota cuando se niega la reimpresión, los cambios y justificaciones de reimpresión se deben guardar en el historial del turno y que el usuario pueda cerrar ventana y volver a la pantalla inicial .

Para los dos casos siguientes el aplicativo debe permitir y restringir.

1.-El aplicativo debe permitir la reimpresión de un certificado máximo dos veces, y que el turno se encuentre en estado desanotado, es decir que el turno no haya sido entregado, si la reimpresión se efectuó el mismo día de su radicación, en caso contrario debe mostrar mensaje restrictivo coherente con la acción.

2.-En esta pantalla también debe tener la opción de reimpresión especial, únicamente para cuando el turno de certificado ha sido entregado o sea de fecha diferente a la radicación, al seleccionarla el aplicativo permitirá incluir el numero del turno del certificado a reimprimir, selección de impresora. Número de copias a imprimir, causal de reimpresión obligatoria y opción de cambio de numero de matricula conforme la descripción dada en párrafos anteriores.

Para todos los casos de expedición de certificados, el sistema debe validar que la matricula no se encuentre asociada a turnos de otros procesos, de lo contrario debe mostrar mensaje restrictivo que el folio se encuentra en trámite.

#### **LA REIMPRESIÓN APLICA PARA TODOS LOS TIPOS DE CERTIFICADOS A EXCEPCION DE LOS NACIONALES QUE NO SE DEBE PERMITIR LA REIMPRESIÓN.**

Se solicita que las reimpresiones solo se realicen en la fecha de la radicación del turno y para las reimpresiones en fechas posteriores se cree una opción de reimpresión especial, se determinó que esta parte de la solicitud no fuera establecida en la solución del requerimiento debido a que las dos posibilidades estarían plasmadas en el mismo ROL, por lo tanto queda al criterio propio de las funciones del funcionario que tenga el rol asignado determinar si ejecuta o no la reimpresión.

Al cargar el rol de reimpresión de certificados en el usuario registrador permite con este usuario la impresión n veces de certificados, lo correcto es que la aplicación permita la re- impresión máximo dos veces. La aplicación permite la re- impresión de los certificados que no se hayan finalizado, lo correcto para el usuario registrador es que permita la re- impresión, bien sea en estado desanotado o estado finalizado (entregado), para los demás usuarios la validación es correcta; desde el usuario registrador, la aplicación permite la re- impresión, sin validar la fecha de radicación lo cual es correcto.

El 30 de Abril 2010 se solicitó la re-especificación del requerimiento, derivada de la prueba del requerimiento, aun se está a la espera. Se reenvió nueva solicitud el día 08 de Julio de 2010.

#### **Re-especificación 13/04/2010**

- La SNR requiere que el rol Reimpresión de certificados aparezca al loguearse el usuario

en el sistema en la lista de roles y no las opciones separadas en las pantallas administrativas. Se deben suprimir todas las opciones de reimpresión que se encuentren en producción tanto en las pantallas administrativas y los botones que aparecen actualmente en la aplicación(reimpresión de certificados, reimpresión de certificados de pertenencia, cambio de matrícula, reimpresión especial) y que se usan para tal fin. El mismo rol debe contener la opción de cambio de matrícula, reimprimir certificado y reimprimir oficio de pertenencia. La nueva opción en la pantalla principal debe permitir:

Reimpresión 2 veces en las fases Desanotado y Finalizado; la reimpresión se puede realizar hasta dos(2) veces por cada uno de los estados. Si el nuevo rol “ROL REIMPRESIÓN DE CERTIFICADOS” es asignado al registrador va a tener las mismas opciones que cualquier otro funcionario autorizado para usar el rol de Reimpresión de Certificados (Se quita código donde se valida TFS 4588: EL USUARIO CON ROL REGISTRADOR PUEDE REIMPRIMIR hasta dos veces). El nuevo rol debe permitir:

1- Reimprimir hasta dos veces en la fase de Desanotado

2-Reimprimir hasta dos veces en la fase de Entregado

o Cambio de matrícula en alguna de las fases de Desanotado y Finalizado. Para la fase de Desanotado se permitirá un cambio de matrícula, en caso de encontrarse en este estado. Para la fase de Entregado se permitirá un cambio de matrícula, en caso de encontrarse en este estado.

o No debe validar fecha (reimpresión en fecha diferente a la radicación.)

o La reimpresión de Oficio de pertenencia se debe realizar cuando el turno se encuentre en la fase de desanotado o finalizado, n veces.

• La nueva pantalla debe tener las opciones de reimpresión, reimpresión de certificados de pertenencia (Oficio de Pertenencia) (Botón) y cambio de matrícula (Botón).

Si se ingresa un turno de certificado que no es de pertenencia y el usuario hace clic en el botón “Oficio Pertenencia” el sistema genera mensaje indicando que el turno no es certificado de Pertenencia.

Se debe tener en cuenta los ítems revisados y aprobados para esta funcionalidad en el despliegue No. 002\_5\_0 del 17 de marzo al 07 de abril de 2010.

En conversaciones con la funcionaria Yamile Moreno el día martes 10/08/2010 se llegó a la conclusión:

\* Eliminar el Rol Reimpresión Especial.

\* Teniendo en cuenta que el Rol Entrega Certificados tiene un botón que envia los turnos al Rol Reimpresión Especial este debe desaparecer además de los otros botones de reimpresión.

\* Los turnos que se encuentren en el rol Reimpresión Especial deben quedar en la Fase de Entrega mas no finalizados.

Para la opción cambio matricula se desplegará una ventana en la que muestra el número de matrícula actual y un campo en el cual se va a ingresar la nueva matrícula, para realizar el cambio de matrícula debe ingresar una nota informativa, al dar clic en aceptar se despliega ventana de validación donde dice “Esta seguro del cambio a realizar” SI - NO

\* SI - Regresa a la ventana inicial de reimpresión, con el cambio de matrícula.

\* NO - El sistema le permite al usuario modificar la matrícula

Se desplegará un botón volver el cual regresará a la ventana de reimpresión.

#### **Re-especificación (Yamile Moreno 13/10/2010) Correo**

Respecto al requerimiento de Reimpresión de certificados , hacemos las siguientes precisiones de la re-especificación del 13-04-2010:

El sistema debe permitir la reimpresión del certificado de tradición hasta dos veces, con las siguientes condiciones:

a)--Si el turno de certificado se encuentra en la fase desanotado se podrá reimprimir hasta dos veces , pero si este mismo turno avanza a la fase de finalizado el sistema no debe permitir reimprimirla, por haber agotado las opciones de reimpresión, validando que la fecha de radicación sea del mismo día.

b) Si el turno de certificado se encuentra en la fase finalizado el aplicativo debe permitir la reimpresión hasta dos veces, validando que no se hayan agotado las dos reimpresiones en el estado desanotado, validando que la fecha de radicación sea del mismo día.

c) El cambio de matrícula se podrá realizar únicamente una vez por turno de certificado, con una fecha límite de cinco días contados a partir de la fecha de radicación, el turno puede encontrarse en la fase desanotado o finalizado, el sistema debe permitir reimprimirla hasta dos veces en el término establecido.

#### **Re-especificación (Yamile Moreno 22/10/2010) Correo**

1) Para los turnos de certificados inmediatos que no requieren cambio de matrícula la fecha máxima para la reimpresión simple es hasta el día siguiente hábil de su radicación ,teniendo en cuenta las condiciones expuestas.

Para los certificados inmediatos que requieren cambio de matrícula la fecha máxima para la reimpresión es de cinco días hábiles a partir de la fecha de su radicación, teniendo en cuenta las condiciones expuestas.

2) Para los turnos de certificados asociados la fecha máxima para la reimpresión es hasta el día siguiente hábil a partir de la fecha que el turno de registro avanza al estado de desanotado fase de entrega y/o entregado .

Para los turnos de certificados asociados en el proceso de reimpresión el sistema debe permitir agregar el número de matrícula para el caso que no lo tenga asociado a la solicitud y permitir la reimpresión del certificado.

Para los turnos de certificados asociados que requieren cambio de matrícula la fecha

máxima para la reimpresión es de cinco días hábiles a partir de la fecha que el turno de registro avanza al estado de desanotado fase de entrega y/o entregado .

3) Para los turnos de certificados de Pertenencia, Ampliación de Tradición, antiguo sistema que no requieren cambio de matrícula la fecha máxima para la reimpresión simple es hasta el día siguiente hábil a partir de la fecha que el turno avanza a la fase de entrega y/o entregado.

Para los certificados de pertenencia que requieren cambio de matrícula la fecha máxima para la reimpresión es de cinco días hábiles a partir de la fecha que el turno avanza a la fase de entrega y/o entregado.

Para los certificados de antiguo sistema en el proceso de reimpresión el sistema debe permitir agregar el número de matrícula para el caso que no tenga número asociado a la solicitud y permitir la reimpresión del certificado.

Para los turnos de todos los tipos de certificados la reimpresión se debe permitir hasta dos veces, una sola vez el cambio de matrícula .

Si el turno de certificado se encuentra en la fase desanotado se podrá reimprimir hasta dos veces, pero si este mismo turno avanza a la fase de finalizado el sistema no debe permitir reimprimirla, por haber agotado las opciones de reimpresión.

Si el turno de certificado se encuentra en la fase finalizado el aplicativo debe permitir la reimpresión hasta dos veces, validando que no se hayan agotado las dos reimpresiones en el estado desanotado o entrega.

Si en estado desanotado fase entrega se imprimió una vez y el turno avanzó a la fase entregado el aplicativo debe permitir reimprimir una vez.

#### **Re-especificación (Correo 22/10/2010) Yamile Moreno**

El aplicativo solo debe permitir el cambio de matrícula por alguna de las matrículas vinculadas al turno de registro y que estas hayan tenido modificación. A excepción de las matrículas asociadas a un turno de registro cuyo acto a registrar fue embargo, y que el resultado de la calificación fue devuelto al público sin registrar o con registro parcial, caso en el cual el sistema debe permitir el cambio de matrícula por alguna de las matrículas asociadas al turno de registro, independientemente si tuvo inscripciones o no.

#### ***DESARROLLO***

- Script Bases de Datos:  
Ejecutar los scripts en el orden en el que se encuentran en el archivo adjunto:  
scripts.txt
- Creación De Clases:
  - NINGUNA

Se modificaron los siguientes archivos:

CLASE:

	<p style="text-align: center;"><b>gov.sir.core.web.helpers.comun.RolHelper.java</b></p> <p style="text-align: center;">Método:</p> <pre><b>public void drawGUI(HttpServletRequest request, JspWriter out)</b></pre>	
	<b>Código Línea 76:</b> <pre>if (!rol.getRoleId().equals(CRoles.ADMINISTRADOR_NACIONAL)) &amp;&amp; (!rol.getRoleId().equals(CRoles.ADMINISTRADOR_REGIONAL)) &amp;&amp; (!rol.getRoleId().equals(CRoles.CONSULTA_REGIONAL)) &amp;&amp; (!rol.getRoleId().equals(CRoles.CONSULTA_NACIONAL)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_GESTION)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL CONTRALORIA)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_CAJERO_CERTIFICADOS_SIR)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_CONSULTA_SIR)) &amp;&amp; (!rol.getRoleId().equals(CRoles.REPORTES_NACIONAL)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_ADMINISTRADOR_IMPRESION)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_CONSULTA_INDICE_PROPIETARIO)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_CREADOR_FOLIOS)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_REIMPRESION_CERTIFICADOS)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_CREACION_MASIVA_AS)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_CONSULTA_ESTADISTICA_SN))){</pre>	
	<b>Por:</b> <pre>if (!rol.getRoleId().equals(CRoles.ADMINISTRADOR_NACIONAL)) &amp;&amp; (!rol.getRoleId().equals(CRoles.ADMINISTRADOR_REGIONAL)) &amp;&amp; (!rol.getRoleId().equals(CRoles.CONSULTA_REGIONAL)) &amp;&amp; (!rol.getRoleId().equals(CRoles.CONSULTA_NACIONAL)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_GESTION)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL CONTRALORIA)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_CAJERO_CERTIFICADOS_SIR)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_CONSULTA_SIR)) &amp;&amp; (!rol.getRoleId().equals(CRoles.REPORTES_NACIONAL)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_ADMINISTRADOR_IMPRESION)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_CONSULTA_INDICE_PROPIETARIO)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_CREADOR_FOLIOS)) &amp;&amp; /*JALCAZAR 11/12/2009 Mantis 02359  * Evita que el rol se despliegue en el listado de inicio  */ &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_REIMPRESION_CERTIFICADOS)) *****!/rol.getRoleId().equals(CRoles.SIR_ROL_CREACION_MASIVA_AS)) &amp;&amp; (!rol.getRoleId().equals(CRoles.SIR_ROL_CONSULTA_ESTADISTICA_SN))}{</pre>	
	<p style="text-align: center;"><b>CLASE:</b></p> <p style="text-align: center;"><b>gov.sir.core.negocio.modelo.constantes.CRoles.java</b></p> <p style="text-align: center;">Método:</p> <pre><b>public void drawGUI(HttpServletRequest request, JspWriter out)</b></pre>	
	<p><b>Se agregaron las siguientes líneas al final de clase:</b></p> <pre>public static final String SIR_ROL_REIMPRESION_CERTIFICADOS = "SIR_ROL_REIMPRESION_CERTIFICADO"; public static final String SIR_ROL_CREACION_MASIVA_AS = "SIR_ROL_CREACION_MASIVA_AS"; public static final String SIR_ROL_CONSULTA_ESTADISTICA_SN = "SIR_ROL_CONSULTA_ESTADISTICA_SN";</pre>	
	<p style="text-align: center;"><b>CLASE:</b></p> <p style="text-align: center;"><b>gov.sir.core.negocio.acciones.administracion.AnTrasladoTurno.java</b></p> <p style="text-align: center;">Método:</p> <pre><b>private EvnRespTrasladoTurno ConsultarTurnoCertificadoCambioMatricula(EvnTrasladoTurno evento) throws EventoException</b></pre>	
	<p><b>Se agregaron las siguientes líneas:</b></p>	
	<p><b>Código Línea 851:</b></p> <pre>/*JAlcazar caso mantis 02359  * variable notas guarda las información de notas informativas al cambiar los turnos  */ List notas = null;</pre>	
	<p><b>Código Línea 893:</b></p> <pre>/*JAlcazar caso mantis 02359  * código para validar que la matrícula no pueda ser cambiada más de una vez  * notas = turno.getNotas(); Iterator i = notas.iterator(); while(i.hasNext()){     Nota nota = (Nota) i.next();</pre>	

```

        if(("12101").equals(nota.getTiponota().getIdTipoNota()) ||
("12103").equals(nota.getTiponota().getIdTipoNota())){
    throw new ConsultaTurnoExceptionException("La matricula solo      puede ser
cambiada una vez.");
}
}

```

**CLASE:**

**gov.sir.core.negocio.acciones.administracion.ANImprimirFolio.java**

**Método:**

**private EvnRespImprimirFolio imprimirCertificado(EvnImprimirFolio  
evento) throws EventoException**

**Código insertado línea 772:**

```

/* JAlcazar caso MANTIS 02359
 * Se definió el objeto Liquidacion global al método
 */
LiquidacionTurnoCertificado liquidacion = null;

```

**Código insertado línea 847:**

```

/* JAlcazar caso MANTIS 02359
 * se valida que el turno se encuentre en la fase de entrega
CFase.CER_ENTREGAR
 */

if(!turnoPersistente.getIdFase().equals(CFase.CER_ENTREGAR)){
    throw new ImpresionException("El Turno debe estar en Fase de ENTREGA
CERTIFICADOS");
}

```

**Código Línea 879:**

```

LiquidacionTurnoCertificado liquidacion = new
LiquidacionTurnoCertificado();

```

**Por: Línea 891**

```

/* JAlcazar caso MANTIS 02359
 * Se definió el objeto Liquidacion global al método
 */
liquidacion = new LiquidacionTurnoCertificado();

```

**Código Línea 889:**

```

if(!(liquidacion.getTipoCertificado().getIdTipoCertificado().equals("2")))

```

**Por: Línea 904**

```

/* JAlcazar caso MANTIS 02359
 * Valida CTipoCertificado.TIPO_PERTENENCIA_ID como constante en la
comparación
*/
if(!(liquidacion.getTipoCertificado().getIdTipoCertificado().equals(CTipoCert
ificado.TIPO_PERTENENCIA_ID)))

```

**Código Línea 960:**

```

if(evento.getTipoEvento().equals("IMPRIMIR_CERTIFICADO_PERTENENCIA
")){

```

**Por: Línea 978**

```
/* JAlcazar caso MANTIS 02359
 * Valida CTipoCertificado.TIPO_PERTENENCIA_NOMBRE no lance
 * excepcion si no posee folio
 */
if(liquidacion.getTipoCertificado().getNombre().equals(CTipoCertificado.TIP
O_PERTENENCIA_NOMBRE)){
```

## ➤ Script Bases de Datos:

- **SP\_CAMBIO\_REIMPRESION\_ESPECIAL.sql** crea Procedimiento para colocar en fase CER\_ENTREGAR los turnos de certificados que estén en el rol Reimpresión Especial.
- **Creacion\_Rol.sql** script para crear el nuevo rol SIR\_ROL\_REIMPRESION\_CERTIFICADOS encargado de la reimpresión.
- **Campo\_SIR\_OP\_SOLICITUD.sql** script creación nuevo campo para manejar las reimpresiones de certificados en la fase FINALIZADO.
- **Delete\_rol\_auriga.sql** script para eliminar las relaciones en auriga del rol SIR\_ROL\_REIMPRESION\_ESPECIAL.
- **Creacion\_Proceso\_workflow.sql** script para crear procesos relacionados al rol SIR\_ROL\_REIMPRESION\_CERTIFICADOS.

## ➤ Creacion De Impresión:

- web\jsp\images\ btn\_oficiopertenencia.gif botoncreado para manejar el proceso de los oficios de pertenencia.

## ➤ Se modificaron los siguientes archivos:

- Se modifco los archivo xml **gov.sir.impl**,  
**org.auriga.smart.web** y **org.auriga.smart.vistas**

**ARCHIVO JSP:**

**WEB-INF.jspSeguro.administracion.**  
**CONFIRMACION\_CAMBIO\_MATRICULA.jsp**

**Código Línea 3:**

```
String ToDo ="admin.turno.cambio.matricula.view";
```

**Por:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Se cambia la vista a la que sigue el proceso
REIMPRESION_CERTIFICADOS
 * Caso Mantis : 02359
 */
String uto ="admin.imprimircertificado.view";
```

**Código Línea 60:**

```
<form name="form1" method="post" action="bloquear.recursos.view">
```

**Por:**

```

<!--
 * @author    : Julio Alcázar Rivas
 * @change    : Se cambia la vista a la que sigue el proceso
REIMPRESION_CERTIFICADOS
 * Caso Mantis : 02359
!-->
<form name="form1" method="post"
action="admin.imprimircertificado.view">

```

### WEB-INF.jspSeguro.administracion.IMPRIMIR\_CERTIFICADO.jsp

#### **Código Insertado Linea 13:**

```

<!--
 * @author    : Julio Alcázar Rivas
 * @change    : Nueva funcion para cambiar action y value  ut form
 * Caso Mantis : 02359
!-->
<%@page import="gov.sir.core.web.acciones.administracion.AWTrasladoTurno" %>

```

#### **Código Insertado Linea 55:**

```

/*
 * @author    : Julio Alcázar Rivas
 * @change    : Nueva funcion para cambiar action y value  ut form
 * Caso Mantis : 02359
*/
function cambiarAcciónSubmit2(text1,text2) {
    document.BUSCAR.action = text2;
    document.BUSCAR.<%= WebKeys.ACCTION %>.value = text1;
}

```

#### **Código Insertado Linea 385:**

```

<!--
 * @author    : Julio Alcázar Rivas
 * @change    : Se agregan los botones de los procesos Oficio Pertenencia y Cambio de
Matrícula
Se elimina  u botón Volver cuya función pasa  u .jsp header de  u vista
 * Caso Mantis : 02359
!-->
<td width="13%">
    <input name="imageField1" type="image"
onClick="cambiarAcción('<%=AWImpresionFolio.IMPRIMIR_CERTIFICADO_PERTENENCIA%>');"
src=<%=request.getContextPath()%>/jsp/images/ ut_oficiopertenencia.gif" width="150"
height="21" border="0" >

    </td>
    <td width="10%">
        <input name="imageField2" type="image"
onClick="cambiarAcciónSubmit2('<%=AWTrasladoTurno.CONSULTAR_TURNO_CERTIFICADO_CAMBIO
_MATRÍCULA%>','trasladoTurno.do');"
src=<%=request.getContextPath()%>/jsp/images/ ut_cambiar_matrícula.gif" width="180"
height="21" border="0" >
    </td>
    <td width="67%">
        <!--
        <td>
            <input name="imageField" type="image"
onClick="cambiarAcción('<%=AWImpresionFolio.TERMINA%>');"
src=<%=request.getContextPath()%>/jsp/images/ ut_volver.gif" width="139" height="21"
border="0" >
        </td>
        !-->
    </td>

```

### WEB-INF.jspSeguro.certificado.TURNO\_CERTIFICADOS\_ENTREGA.jsp

#### **Código Línea 1036:**

```

int i =
((Integer)session.getAttribute(WebKeys.NUMERO_MAX_IMPRESIONES)).intValue();
    u(solicitud.getNumImpresiones() < i ){%>
<form name="ENTREGA" method="post" action="certificado.do">
<input type="hidden" name="<%>=WebKeys.ACCION%" value="<%>=AWCertificado.IMPRIMIR%>">
<td width="150" align="center"><p>
<input name="imageField52" type="image" src="<%>=request.getContextPath()%>/jsp/images/ ut_reimpresion.gif" width="139" height="21" border="0">
</p></td>
</form>
<%> else {%
<td width="150" align="center"><p>
<image name="imageField52" src="<%>=request.getContextPath()%>/jsp/images/ ut_imprimir_off.gif" width="139" height="21" border="0">
</p></td>
<%>
Por:
/*
* @author : Julio Alcázar Rivas
* @change : Se elimina la utor que los certificados sean impresos en el
proceso de Entrega de Certificados Comentariando este utor .
* Caso Mantis : 02359
*/
        //      int I =
((Integer)session.getAttribute(WebKeys.NUMERO_MAX_IMPRESIONES)).in
tValue();
        //      if(solicitud.getNumImpresiones() < I )
//{%
<!--      <form name="ENTREGA" method="post" action="certificado.do">
<input type="hidden" name="<%>=WebKeys.ACCION%" value="<%>=AWCertificado.IMPRIMIR%>">
<td width="150" align="center"><p>
<input name="imageField52" type="image" src="<%>=request.getContextPath()%>/jsp/images/btn_reimpresion.gif" width="139" height="21" border="0">
</p></td>
</form> !-->
<%//> else {%
<!--      <td width="150" align="center"><p>
<image name="imageField52" src="<%>=request.getContextPath()%>/jsp/images/btn_imprimir_off.gif" width="139" height="21" border="0">
</p></td> !-->
<%//>

```

#### Código Línea 60:

```

<form name="ENTREGA" method="post" action="certificado.do">

<input type="hidden" name="<%>=AWCertificado.PRINTESPECIAL%" value="<%>=AWCertificado.PRINTESPECIAL%>">
<input type="hidden" name="<%>=WebKeys.ACCION%" value="<%>=AWCertificado.IMPRIMIR_ESPECIAL%>">

```

```

<input type="hidden" name="<%={WebKeys.OCULTAR_TURNO%}" value="TRUE">
<input type="hidden" name="<%={WebKeys.OCULTAR_FOLIO%}" value="TRUE">
<td width="150" align="center"><p>
<input name="imageField52" type="image"
src="<%={request.getContextPath()%}/jsp/images/ ut_reimpresion_especial.gif"
width="180" height="21" border="0">
</p></td>
</form>

```

**Por:**

```

<!--  /*
 * @author : Julio Alcázar Rivas
 * @change : Se elimina la utor que los certificados sean
reimpresos en el proceso de Entrega de Certificados
Comentariando este utor .
* Caso Mantis : 02359
*/
<form name="ENTREGA" method="post" action="certificado.do">
<input type="hidden" name="<%={AWCertificado.PRINTESPECIAL%}" value="<%={AWCertificado.PRINTESPECIAL%}>">
<input type="hidden" name="<%={WebKeys.ACCTION%}" value="<%={AWCertificado.IMPRIMIR_ESPECIAL%}>">
<input type="hidden" name="<%={WebKeys.OCULTAR_TURNO%}" value="TRUE">
<input type="hidden" name="<%={WebKeys.OCULTAR_FOLIO%}" value="TRUE">
<td width="150" align="center"><p>
<input name="imageField52" type="image"
src="<%={request.getContextPath()%}/jsp/images/ico_reimpresion_especial.gif" width="180" height="21" border="0">
</p></td>
</form>

```

i-->

#### CLASE:

**gov.sir.core.negocio.modelo.SolicitudCertificado.java**

**Código Insertado Línea 25:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Se agrega nuevo campo atributo para manejar ut reimpresiones en u
fase FINALIZADO
* Caso Mantis : 02359
*/
private int numReimpresiones;

```

**Código Insertado Línea 172:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : SET Y GET para manejar ut reimpresiones en u fase FINALIZADO
* Caso Mantis : 02359
*/
public int getNumReimpresiones() {
    return numReimpresiones;
}

public void setNumReimpresiones(int numReimpresiones) {
    this.numReimpresiones = numReimpresiones;
}

```

**CLASE:**  
**gov.sir.core.negocio.modelo.jdogenie.SolicitudCertificadoEnhanced.java**

**Código Insertado Linea 26:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Se agrega nuevo campo atributo para manejar ut reimpresiones en u fase
FINALIZADO
 * Caso Mantis : 02359
 */
private int numReimpresiones;
```

**Código Insertado Linea 150:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : SET Y GET para manejar ut reimpresiones en u fase FINALIZADO
 * Caso Mantis : 02359
 */
public int getNumReimpresiones() {
    return numReimpresiones;
}

public void setNumReimpresiones(int numReimpresiones) {
    this.numReimpresiones = numReimpresiones;
}
```

**CLASE:**  
**gov.sir.core.negocio.modelo.transferutil.**  
**SolicitudCertificadoTransferCopier.java**

**Método:**

```
public static org.auriga.core.modelo.TransferObject deepCopy(
gov.sir.core.negocio.modelo.jdogenie.SolicitudCertificadoEnhanced
enhancedObject, uto.util.HashMap utor) {
```

**Código Insertado Linea 446:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : TransferObject u numReimpresiones
 * Caso Mantis : 02359
 */
try {
    transferObject.setNumReimpresiones(enhancedObject.getNumReimpresiones());
} catch (Throwable th) {}
```

**Método:**

```
public static gov.sir.core.negocio.modelo.jdogenie Enhanced deepCopy(
gov.sir.core.negocio.modelo.SolicitudCertificado transferObject,
uto.util.HashMap utor){
```

**Código Insertado Linea 889:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Enhanced u numReimpresiones
 * Caso Mantis : 02359
 */
try {
    enhancedObject.setNumReimpresiones(transferObject.getNumReimpresiones());
} catch (Throwable th) {}
```

**CLASE:**

**gov.sir.core.negocio.acciones.administracion.ANImpimirFolio.java**

**Método:**

```
private EvnRespImprimirFolio imprimirCertificado(EvnImprimirFolio
evento) throws EventoException {
```

**Código Insertado Linea 810:**

```
/*
```

```

    * @author : Julio Alcázar Rivas
    * @change : Se definió un objeto liquidacion y folio global y se define su
variable numReimpresionesCert
    * Caso Mantis : 02359
    */
LiquidacionTurnoCertificado liquidacion = null;
Folio folio = null;
int numReimpresionesCert = 0;

```

**Código Insertado Línea 890:**

```

/*
    * @author : Julio Alcázar Rivas
    * @change : se obtienen las reimpresiones en su estado finalizado
    * Caso Mantis : 02359
    */
numReimpresionesCert = solicitudCert.getNumReimpresiones();

```

**Código Línea 917:**

```

u(numImpresionesSolicitadas > solicitudCert.getNumeroCertificados()
&& !esUsuarioRegistrador){
    throw new ImpresionException("No se puede reimprimir más veces el número de
certificados solicitados.");
}

```

**Por:**

```

/*
    * @author : Julio Alcázar Rivas
    * @change : se eliminó el if la estrategia de esUsuarioRegistrador para restringir el
número de impresiones a los usuarios
    * que posean el rol Crol.SIR_ROL_REGISTRADOR
    * Caso Mantis : 02359
    */
if(numImpresionesSolicitadas > solicitudCert.getNumeroCertificados()){
    throw new ImpresionException("No se puede reimprimir más veces del número de
certificados solicitados.");
}

```

**Código Línea 923:**

```
LiquidacionTurnoCertificado liquidacion = new LiquidacionTurnoCertificado();
```

**Por:**

```

/*
    * @author : Julio Alcázar Rivas
    * @change : Se creó el objeto estrategia global al autor
    * Caso Mantis : 02359
    */
estrategia = new LiquidacionTurnoCertificado();

```

**Código Línea 932:**

```

u(evento.getTipoEvento().equals("IMPRIMIR_CERTIFICADO_PERTENENCIA")){
/* Jalcazar caso MANTIS 02359
 * Valida CtipoCertificado.TIPO_PERTENENCIA_ID como constante en la comparación
 */

u(!(liquidacion.getTipoCertificado().getIdTipoCertificado().equals(CtipoCertificado.TIPO_PERTENENCIA_ID)))
    throw new ImpresionException("No se puede reimprimir. El certificado no tiene una Pertenencia");
}

```

**Por:**

```

/*
    * @author : Julio Alcázar Rivas
    * @change : Modificar la estrategia de tipos de certificados

```

```

    * Inmediatos EvnImprimirFolio.IMPRIMIR_CERTIFICADO y Pertenencia
    EvnImprimirFolio.IMPRIMIR_CERTIFICADO_PERTENENCIA
    * Caso Mantis : 02359
    */

    if(evento.getTipoEvento().equals(EvnImprimirFolio.IMPRIMIR_CERTIFICADO)){
        if(i( utor strac.getTipoCertificado().getIdTipoCertificado().equals(CtipoCertificado.TIPO_IN
        MEDIATO_ID))){
            throw new ImpresionException("No se puede reimprimir. El certificado no es
        Inmediato");
        }
    }

    if(evento.getTipoEvento().equals(EvnImprimirFolio.IMPRIMIR_CERTIFICADO_PERTENENCIA))
    {

        if(i( utor strac.getTipoCertificado().getIdTipoCertificado().equals(CtipoCertificado.TIPO_PER
        TENENCIA_ID))){
            throw new ImpresionException("No se puede reimprimir el oficio de pertenencia. El
        certificado no es de Pertenencia");
        }
    }
}

```

### Código Insertado Línea 970:

```

/*
 * @author      : Julio Alcázar Rivas
 * @change      : Se valida que   u turno se encuentre em   u fase de entrega
Cfase.CER_ENTREGAR o Cfase.FINALIZADO
 * Caso Mantis : 02359
*/
    u(!turnoPersistente.getIdFase().equals(Cfase.CER_ENTREGAR) &&
!turnoPersistente.getIdFase().equals(Cfase.FINALIZADO)){
        throw new ImpresionException("El Turno debe estar en Fase de ENTREGA
CERTIFICADOS o FINALIZADO");
    }

    boolean certPertenenciaSinFolio = false;
    folio = new Folio();
    /***** Em caso de que   u certificado sea de pertenencia y no tenga asociado
* em matricula no se imprime   u certificado y no se lanza   u excepcion *****/
try{
    folio = this.getFolioFromTurno(turnoPersistente);
    u (folio == null){
        throw new ImpresionException("Error   u imprimir   u folio:   u folio no em
podido cargarse.");
    }
}catch(EventoException excepcion){
/*
 * @author      : Julio Alcázar Rivas
 * @change      : Valida CtipoCertificado.TIPO_PERTENENCIA_NOMBRE no lance excepcion si
no posee folio
 * Caso Mantis : 02359
*/
    u(liquidacion.getTipoCertificado().getNombre().equals(CtipoCertificado.TIPO_PERTENENCIA_NOMBRE)){
        certPertenenciaSinFolio = true;
    }else{
        throw excepcion;
    }
}

/*
 * @author      : Julio Alcázar Rivas
 * @change      : Se agrega   u para solo reimprimir los certificados inmediatos y los
utros de pertenencia
 * Caso Mantis : 02359
*/

```

```
u(!evento.getTipoEvento().equals(EvnImprimirFolio.IMPRIMIR_CERTIFICADO_PERTENENCIA)){
```

### Código Línea 997:

```
    u(!validarImpresionesAnteriores && !esUsuarioRegistrador){
        u (maxImprUsuario<=numImpresionesCert-1){
            throw new ImpresionException("Se ha alcanzado el número
máximo("+maxImprUsuario+) de impresiones para esa solicitud");
        }
    }

Por:
/*
 * @author : Julio Alcázar Rivas
 * @change : se elimina el rótulo stra de esUsuarioRegistrador para restringir el
numero de impresiones a los usuarios
 * que posean el rol Crol.SIR_ROL_REGISTRADOR y se agrega la rótulo stra para la fase
Cfase.FINALIZADO
 * Caso Mantis : 02359
 */
if(validarImpresionesAnteriores){

    if (maxImprUsuario<=numImpresionesCert-1 &&
turnoPersistente.getIdFase().equals(Cfase.CER_ENTREGAR)){
        throw new ImpresionException("Se ha alcanzado el número
máximo("+maxImprUsuario+) de impresiones para esa solicitud");
    }
    if (maxImprUsuario<=numReimpresionesCert &&
turnoPersistente.getIdFase().equals(Cfase.FINALIZADO)){
        throw new ImpresionException("Se ha alcanzado el número
máximo("+maxImprUsuario+) de impresiones para esa solicitud");
    }
}
```

### Código Línea 997:

```
solicitudCert.setNumImpresiones(numImpresionesCert+1);
```

### Por:

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Se agrega if para set el numero de rótulos strac en las fases
Cfase.FINALIZADO or Cfase.CER_ENTREGAR
 * Caso Mantis : 02359
 */
if (turnoPersistente.getIdFase().equals(Cfase.FINALIZADO)){

    solicitudCert.setNumReimpresiones(numReimpresionesCert+1);
} else {
    solicitudCert.setNumImpresiones(numImpresionesCert+1);
}
```

### Código Línea 1098:

```
u(evento.getTipoEvento().equals("IMPRIMIR_CERTIFICADO_PERTENENCIA")){
    imprimirOficio(evento);
}
```

### Por:

```
/*
 * @author : Julio Alcázar Rivas
 * @change : ELSE en el cual se realiza la rótulo strac de los oficios de pertenencia
 * Caso Mantis : 02359
*/
} else {
    imprimirOficio(evento, folio, turnoPersistente, solicitudCert);
}
```

Método:

```
private boolean imprimirOficio(EvnImprimirFolio evento, Folio folio,
    Turno turnoPersistente, SolicitudCertificado solicitudCert) throws
    ImpresionException{
```

**Código Línea 1159:**

```
private boolean imprimirOficio(EvnImprimirFolio evento) throws ImpresionException{
    Turno turno = evento.getTurno();
    Circulo circulo = evento.getCirculo();
    TurnoHistoria turnoHistoria;
    List<utor> s;
    try{
        TurnoPk turnoPk = new TurnoPk();
        turnoPk.anio = turno.getAnio();
        turnoPk.idCirculo = turno.getIdCirculo();
        turnoPk.idProceso = turno.getIdProceso();
        turnoPk.idTurno = turno.getIdTurno();
        utor s = hermod.getOficiosTurno(turnoPk);
        turnoHistoria = obtenerTurnoHistoriaActual(evento);
    }catch(Throwable e){
        throw new ImpresionException(e.getMessage());
    }
    boolean existeOficio = false;
    Oficio oficio = new Oficio();
    for(Iterator<utor> iter = s.iterator(); iter.hasNext();){
        oficio = (Oficio)iter.next();

        if(oficio.getTurnoHistoria().getFase().equals(turnoHistoria.getFaseAnterior())){
            existeOficio = true;
            break;
        }else{
            existeOficio = false;
        }
    }
    if(!existeOficio){
        throw new ImpresionException("No existe em oficio para imprimir");
    }
    ImprimibleOficioPertenencia imprimible = new
    ImprimibleOficioPertenencia(oficio.getTextoOficio(), CtipoImprimible.OFICIO);
    Hashtable tabla= new Hashtable();
    tabla.put(Processor.IMPRIMIBLE,imprimible);
    if(evento.getImpresoraSeleccionada()!=null){
        tabla.put(Processor.IMPRESORA_SELECCIONADA,
        evento.getImpresoraSeleccionada());
    }
    tabla.put(Processor.NUM_COPIAS_IMPRIMIBLE, new
    Integer(evento.getNumeroImpresiones()));
    tabla.put(Processor.ESPERA, Cimpresion.DEFAULT_ESPERA_IMPRESION);
    tabla.put(Processor.INTENTOS, Cimpresion.DEFAULT_INTENTOS_IMPRESION);
    tabla.put(Processor.IMPRIMIBLE,imprimible);
    return(imprimirAdministrador(circulo.getIdCirculo(),imprimible,tabla));
}
Por:
/*
 * @author : Julio Alcázar Rivas
 * @change : se cambia la signatura del utor para que reciba Folio, Turno y
SolicitudCertificado.
 * Ademas se realizaron diferentes modificaciones al comportamiento del utor como:
 * a) Permitir reimpresiones en la fase finalizado Cfase.FINALIZADO
 * b) Agregar la utor strac en el historial del turno
 * Caso Mantis : 02359
*/
utor s boolean imprimirOficio(EvnImprimirFolio evento, Folio folio, Turno
turnoPersistente, SolicitudCertificado solicitudCert) throws ImpresionException{
    Turno turno = evento.getTurno();
    Circulo circulo = evento.getCirculo();
    TurnoHistoria turnoHistoria;
    List<oficios>;
    Usuario usuario = null;
    boolean resultadoImpr = false;
```

```

try{
    usuario = fenrir.getUsuario(evento.getUsuario());
    TurnoPk turnoPk = new TurnoPk();
    turnoPk.anio = turno.getAnio();
    turnoPk.idCirculo = turno.getIdCirculo();
    turnoPk.idProceso = turno.getIdProceso();
    turnoPk.idTurno = turno.getIdTurno();
    oficios = hermod.getOficiosTurno(turnoPk);
    turnoHistoria = obtenerTurnoHistoriaActual(evento);

    boolean existeOficio = false;
    Oficio oficio = new Oficio();
    for(Iterator iter = oficios.iterator() ; iter.hasNext() ;){
        oficio = (Oficio)iter.next();

        if(oficio.getTurnoHistoria().getFase().equals(turnoHistoria.getFaseAnterior()) ||
turnoHistoria.getFase().equals(Cfase.FINALIZADO ) ){
            existeOficio = true;
            break;
        }
    }
    if(!existeOficio){
        throw new ImpresionException("No existe un oficio para imprimir");
    }
    ImprimibleOficioPertenencia imprimible = new
ImprimibleOficioPertenencia(oficio.getTextoOficio(),CtipoImprimible.OFICIO);
    Hashtable utor= new Hashtable();
    utor.put(Processor.IMPRIMIBLE,imprimible);
    if(evento.getImpresoraSeleccionada()!=null){
        tabla.put(Processor.IMPRESORA_SELECCIONADA,
        evento.getImpresoraSeleccionada());
    }
    tabla.put(Processor.NUM_COPIAS_IMPRIMIBLE, new
Integer(evento.getNumeroImpresiones()));
    tabla.put(Processor.ESPERA, Cimpresion.DEFAULT_ESPERA_IMPRESION);
    tabla.put(Processor.INTENTOS,
    Cimpresion.DEFAULT_INTENTOS_IMPRESION);
    utor.put(Processor.IMPRIMIBLE,imprimible);

    resultadoImpr=imprimirAdministrador(circulo.getIdCirculo(),imprimible,tabla);

    if (resultadoImpr){
        TurnoHistoria turnoHist=new TurnoHistoria();
        turnoHist.setUsuario(usuario);
        turnoHist.setUsuarioAtiende(usuario);
        turnoHist.setFase(Cfase.CER_IMPRIMIR);
        turnoHist.setIdTurno(turno.getIdWorkflow());

        turnoHist.setNumeroCopiasReimpresion(evento.getNumeroImpresiones());
        turnoHist.setIdMatriculaImpresa(folio.getIdMatricula());
        hermod.addTurnoHistoriaToTurno(turnoPersistente,turnoHist);

        Solicitud resultadoSolicitud = null;
        resultadoSolicitud =
hermod.updateSolicitudCertificado(solicitudCert);
        if (resultadoSolicitud == null){
            Log.getInstance().error(ANImprimirFolio.class,"Error en el
método que actualiza la solicitud");
        }
    }
}catch(Throwable e){
    throw new ImpresionException(e.getMessage());
}
return resultadoImpr;
}

```

**CLASE:**

**gov.sir.core.negocio.acciones.administracion.AnTrasladoTurno.java**

**Método:**

**private EvnRespTrasladoTurno  
consultarTurnoCertificadoCambioMatricula(EvnTrasladoTurno evento)**

**throws EventoException {**

**Código Insertado Linea 851:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Variable notas guarda ut informacion de notas informativas u cambiar
los turnos
 * Caso Mantis : 02359
 */
List notas = null;
```

**Código Linea 879:**

```
u (!turno.getIdFase().equals(Cfase.CER_ENTREGAR)) {
    throw new ConsultaTurnoExceptionException("El turno debe estar em u fase de
Entrega Certificados.");
}
```

**Por:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Los turnos deben estar en la fase Cfase.CER_ENTREGAR or
Cfase.FINALIZADO.
 * Caso Mantis : 02359
 */
if (iturno.getIdFase().equals(Cfase.CER_ENTREGAR) &&
iturno.getIdFase().equals(Cfase.FINALIZADO)) {
    new ConsultaTurnoExceptionException("El turno debe estar en la fase de Entrega
Certificados o Finalizado.");
}
```

**Código Insertado Linea 851:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Valida que u matricula solo pueda ser cambiada em vez em u fase
Cfase.CER_ENTREGAR y em vez em u fase Cfase.FINALIZADO.
 * Caso Mantis : 02359
 */
notas = turno.getNotas();
Iterator i = notas.iterator();
while(i.hasNext()){
    Nota nota = (Nota) i.next();
    u("12101").equals(nota.getTiponota().getIdTipoNota()) ||
("12103").equals(nota.getTiponota().getIdTipoNota())){
        u(turno.getIdFase().equals(nota.getIdFase())){
            throw new ConsultaTurnoExceptionException("La matricula solo puede ser
cambiada em vez em u fase: "+ nota.getIdFase());
        }
    }
}
```

**CLASE:**

**gov.sir.core.web.acciones.administracion.AWTrasladoTurno.java**

**Método:**

**public void doEnd(HttpServletRequest request, EventoRespuesta evento)**

{

**Código Linea 1592:**

```
proceso.setIdProceso(1);
```

**Por:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : se cambia el idProceso a 10 el cual corresponde al proceso
Reimpresion_Certificados
 * Caso Mantis : 02359
 */
proceso.setIdProceso(10);
```

**CLASE:**

**gov.sir.hermod.dao.impl.jdogenie.JDOSolicitudesDAO.java**

**Método:**

```

public SolicitudCertificado
modificarSolicitudCertificado(SolicitudCertificado solCer) throws
DAOException {

```

#### Código Insertado Linea 6806:

```

/*
 * @author    : Julio Alcázar Rivas
 * @change   : Se agregan los botones de los procesos Oficio Pertenencia y Cambio de
Matricula
 * Caso Mantis : 02359
 */
solCer.setNumReimpresiones(solCertificado.getNumReimpresiones());

```

#### Desarrollo de las ultimas Re-especificaciones

- Script Bases de Datos:

- **SP\_CAMBIO\_REIMPRESION\_ESPECIAL.sql** crea Procedimiento para colocar en fase CER\_ENTREGAR los turnos de certificados que estén en el rol Reimpresión Especial.
- **Creacion\_Rol.sql** script para crear el nuevo rol SIR\_ROL\_REIMPRESION\_CERTIFICADOS encargado de la reimpresión.
- **Delete\_rol\_auriga.sql** script para eliminar las relaciones en auriga del rol SIR\_ROL\_REIMPRESION\_ESPECIAL.
- **Creacion\_Proceso\_workflow.sql** script para crear procesos relacionados al rol SIR\_ROL\_REIMPRESION\_CERTIFICADOS.
- **PACKAGE\_pck\_utilidades\_validaciones.sql** firma del package que incluye la FUNCTION is\_cambio\_matricula\_cert (p\_turno IN VARCHAR2, p\_matricula IN VARCHAR2).
- **PACKAGE BODY\_pck\_utilidades\_validaciones.sql** cuerpo del package que incluye la FUNCTION is\_cambio\_matricula\_cert (p\_turno IN VARCHAR2, p\_matricula IN VARCHAR2).

- Creacion De Imagenes:

- web\jsp\images\ btn\_oficiopertenencia.gif botón creado para manejar el proceso de los oficios de pertenencia.

- Se modificaron los siguientes archivos:

- Se modificó los archivo xml **org.auriga.smart.web** y **org.auriga.smart.vistas**

#### ARCHIVO JSP:

```

WEB-INF.jspSeguro.administracion.
CONFIRMACION_CAMBIO_MATRICULA.jsp

```

#### Código Linea 3:

```
String ToDo = "admin.turno.cambio.matricula.view";
```

#### Por:

```

/*
 * @author    : Julio Alcázar Rivas
 * @change   : Se cambia la vista a la que sigue el proceso REIMPRESION_CERTIFICADOS
 * Caso Mantis : 02359
 */

```

```

String ToDo = "admin.imprimircertificado.view";
Código Linea 60:
<form name="form1" method="post" action="bloquear.recursos.view">

```

**Por:**

```

<!--
 * @author    : Julio Alcázar Rivas
 * @change    : Se cambia la vista a la que sigue el proceso
REIMPRESION_CERTIFICADOS
 * Caso Mantis : 02359
!-->
<form name="form1" method="post" action="admin.imprimircertificado.view">

```

## WEB-INF.jspSeguro.administracion.IMPRIMIR\_CERTIFICADO.jsp

**Código Insertado Linea 13:**

```

<!--
 * @author    : Julio Alcázar Rivas
 * @change    : Nueva funcion para cambiar action y value del form
 * Caso Mantis : 02359
!-->
<%@page import="gov.sir.core.web.acciones.administracion.AWTrasladoTurno" %>

```

**Código Insertado Linea 55:**

```

/*
 * @author    : Julio Alcázar Rivas
 * @change    : Nueva funcion para cambiar action y value del form
 * Caso Mantis : 02359
*/
function cambiarAcciónSubmit2(text1,text2) {
    document.BUSCAR.action = text2;
    document.BUSCAR.<%= WebKeys.ACCTION %>.value = text1;
}

```

**Código Insertado Linea 385:**

```

<!--
 * @author    : Julio Alcázar Rivas
 * @change    : Se agregan los botones de los procesos Oficio Pertenencia y Cambio de
Matrícula
    Se elimina el botón Volver cuya función pasa al .jsp header de la vista
 * Caso Mantis : 02359
!-->
<td width="13%">
    <input name="imageField1" type="image"
onClick="cambiarAcción('<%=AWImpresionFolio.IMPRIMIR_CERTIFICADO_PERTENENCIA%>');"
src="<%=request.getContextPath()%>/jsp/images/btn_oficiopertenencia.gif" width="150"
height="21" border="0" >

    </td>
    <td width="10%">
        <input name="imageField2" type="image"
onClick="cambiarAcciónSubmit2('<%=AWTrasladoTurno.CONSULTAR_TURNO_CERTIFICADO_CAMBIO
_MATRÍCULA%>','trasladoTurno.do');"
src="<%=request.getContextPath()%>/jsp/images/btn_cambiar_matrícula.gif" width="180"
height="21" border="0" >
    </td>
    <td width="67%">
        <!--
        <td>
            <input name="imageField" type="image"
onClick="cambiarAcción('<%=AWImpresionFolio.TERMINA%>');"
src="<%=request.getContextPath()%>/jsp/images/btn_volver.gif" width="139" height="21"
border="0" >
        </td>
        !-->
    </td>

```

## WEB-INF.jspSeguro.certificado.TURNO\_CERTIFICADOS\_ENTREGA.jsp

### Código Línea 1036:

```
int i = ((Integer)session.getAttribute(WebKeys.NUMERO_MAX_IMPRESIONES)).intValue();
if(solicitud.getNumImpresiones() < i ){%>
<form name="ENTREGA" method="post" action="certificado.do">
<input type="hidden" name="<%={WebKeys.ACCION%}" value="<%={AWCertificado.IMPRIMIR%}>">
<td width="150" align="center"><p>
<input name="imageField52" type="image" src="<%={request.getContextPath()%}/jsp/images/btn_reimpresion.gif" width="139" height="21" border="0">
</p></td>
</form>
<%} else {%
<td width="150" align="center"><p>
<image name="imageField52" src="<%={request.getContextPath()%}/jsp/images/btn_imprimir_off.gif" width="139" height="21" border="0">
</p></td>
<%}

```

### Por:

```
/*
* @author : Julio Alcázar Rivas
* @change : Se elimina la opcion que los certificados sean impresos en el proceso de Entrega de Certificados Comentariando este codigo.
* Caso Mantis : 02359
*/
// int i =
((Integer)session.getAttribute(WebKeys.NUMERO_MAX_IMPRESIONES)).intValue();
// if(solicitud.getNumImpresiones() < i )
//{%
!--<form name="ENTREGA" method="post" action="certificado.do">
<input type="hidden" name="<%={WebKeys.ACCION%}" value="<%={AWCertificado.IMPRIMIR%}>">
<td width="150" align="center"><p>
<input name="imageField52" type="image" src="<%={request.getContextPath()%}/jsp/images/btn_reimpresion.gif" width="139" height="21" border="0">
</p></td>
</form> !--
<%//} else {%
!--<td width="150" align="center"><p>
<image name="imageField52" src="<%={request.getContextPath()%}/jsp/images/btn_imprimir_off.gif" width="139" height="21" border="0">
</p></td> !--
<%//}

```

### Código Línea 60:

```
<form name="ENTREGA" method="post" action="certificado.do">

<input type="hidden" name="<%={AWCertificado.PRINTESPECIAL%}" value="<%={AWCertificado.PRINTESPECIAL%}>">
<input type="hidden" name="<%={WebKeys.ACCION%}" value="<%={AWCertificado.IMPRIMIR_ESPECIAL%}>">
<input type="hidden" name="<%={WebKeys.OCULTAR_TURNO%}" value="TRUE">
<input type="hidden" name="<%={WebKeys.OCULTAR_FOLIO%}" value="TRUE">
<td width="150" align="center"><p>
<input name="imageField52" type="image" src="<%={request.getContextPath()%}/jsp/images/ico_reimpresion_especial.gif" width="180" height="21" border="0">
</p></td>
</form>

```

### Por:

```

<!-- /*
 * @author : Julio Alcázar Rivas
 * @change : Se elimina la opcion que los certificados sean reimpresos en el proceso de
Entrega de Certificados Comentariando este codigo.
 * Caso Mantis : 02359
*/
<form name="ENTREGA" method="post" action="certificado.do">

<input type="hidden" name="<%>AWCertificado.PRINTESPECIAL%<%>" value="<%>AWCertificado.PRINTESPECIAL%<%>"/>
<input type="hidden" name="<%>WebKeys.ACCTION%<%>" value="<%>AWCertificado.IMPRIMIR_ESPECIAL%<%>"/>
<input type="hidden" name="<%>WebKeys.OCULTAR_TURNO%<%>" value="TRUE"/>
<input type="hidden" name="<%>WebKeys.OCULTAR_FOLIO%<%>" value="TRUE"/>
<td width="150" align="center"><p>
<input name="imageField52" type="image"
src="<%>request.getContextPath()%>/jsp/images/ico_reimpresion_especial.gif"
width="180" height="21" border="0">
</p></td>
</form>

!-->

```

### WEB-INF.jspSeguro.administracion. TURNO\_CERTIFICADOS\_CAMBIAR\_MATRICULA\_ADMIN.jsp

#### **Código Linea 17:**

String actualMatricula=folio.getIdMatricula();

#### **Por:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Se verifica que el folio asociado al turno sea diferente a null
 * Caso Mantis : 02359
*/
String actualMatricula = null;
if(folio != null){
    actualMatricula=folio.getIdMatricula();
}

```

#### **Código Insertado Linea 109:**

```

<%
/*
 * @author : Julio Alcázar Rivas
 * @change : if que verifica que actualMatricula sea diferente a null
 * Caso Mantis : 02359
*/
if (actualMatricula!= null){
%>

```

#### **Código Insertado Linea 122:**

```

<%
/*
 * @author : Julio Alcázar Rivas
 * @change : cierre if que verifica que actualMatricula sea diferente a null
 * Caso Mantis : 02359
*/
%>
%>

```

#### **Código Linea 111:**

<td>Nueva Matr&iacute;cula </td>

**Por:**

```
<!--
 * @author    : Julio Alcázar Rivas
 * @change    : Ajustes al diseño html
 * Caso Mantis : 02359
-->
<td width="100" align="left">Nueva Matrícula </td>
```

**Código Línea 114:**

```
<td><%=idCirculo%>-<input type="text" name=<%=AWTrasladoTurno.FOLIO_MATRICULA_ID%>" class="camposformtext" size="25" maxlength="25"/></td>
```

**Por:**

```
<!--
 * @author    : Julio Alcázar Rivas
 * @change    : Ajustes al diseño html
 * Caso Mantis : 02359
-->
<td align="left"><%=idCirculo%>-<input type="text"
name=<%=AWTrasladoTurno.FOLIO_MATRICULA_ID%>" class="camposformtext" size="25"
maxlength="25"/></td>
```

**Código Línea 138:**

```
%></td><td><input type="text" name=<%=AWTrasladoTurno.FOLIO_MATRICULA_ID%>" class="camposformtext" size="25" maxlength="25"/></td>
```

**Por:**

```
%>
<!--
 * @author    : Julio Alcázar Rivas
 * @change    : Ajustes al diseño html
 * Caso Mantis : 02359
-->
</td><td align="left"><input type="text"
name=<%=AWTrasladoTurno.FOLIO_MATRICULA_ID%>" class="camposformtext" size="25"
maxlength="25"/></td>
```

**CLASE:**

**gov.sir.core.negocio.acciones.administracion.ANImprimirFolio.java**

**Método:**

**private EvnRespImprimirFolio imprimirCertificado(EvnImprimirFolio evento) throws EventoException {**

**Se cambio casi todo el comportamiento del metodo:**

```
private EvnRespImprimirFolio imprimirCertificado(EvnImprimirFolio evento) throws EventoException
{
    //HAY QUE COLOCAR ESTA VARIABLE SI SE DESEA QUE EL SISTEMA VALIDE QUE SE PUEDA REIMPRIMIR
    //SOLO SI SE HAN AGOTA LOS INTENTOS DEL USUARIO SIR_ROL_CAJERO Y DEL USUARIO SIR_ROL_IMPRESION_ESPECIAL
    boolean validarImpresionesAnteriores = false;
    boolean esUsuarioRegistrador = false;

    Rol      rol = null;
    Turno   turno = null;
    Circulo circulo = null;
    Usuario usuario = null;
    ProcesoPk oid = null;
    Proceso proceso = null;
    /*
```

```

 * @author : Julio Alcázar Rivas
 * @change : Se definió el objeto liquidacion y folio global al método y define la variable
numReimpresionesCert
 * Caso Mantis : 02359
 */
LiquidacionTurnoCertificado liquidacion = null;
Folio folio = null;

Turno turnoPersistente = null;
UsuarioCirculo usuarioCirculo = null;

SolicitudCertificado solicitudCert = null;

String idCirculoUsuario = null;
String idCirculoTurno = null;

int numImpresionesSolicitadas = 0;
int numImpresionesCert = 0;
int maxImprUsuario = 0;
int maxNumImpresiones = 0;

Long idProceso = null;

List roles = null;
List usuarios = null;

Iterator itRoles = null;

try {
    turno = evento.getTurno();
    circulo = evento.getCirculo();
    usuario = fenrir.getUsuario(evento.getUsuario());
/*
 * @author : Julio Alcázar Rivas
 * @change : Se define la variable circuloPk
 * Caso Mantis : 02359
 */
CirculoPk circuloPk = new CirculoPk();
circuloPk.idCirculo = circulo.getIdCirculo();
idCirculoUsuario = null;

if(!usuario.getUsuarioCirculos().isEmpty()){
    usuarioCirculo =
(UsuarioCirculo)usuario.getUsuarioCirculos().get(0);
    idCirculoUsuario = usuarioCirculo.getIdCirculo();
}
else{
    throw new ImpresionException("El usuario no tiene círculos
asociados");
}

turnoPersistente = hermod.getTurnobyWF(turno.getIdWorkflow());

//comprobar que el turno no está anulado
if (turnoPersistente.getAnulado() != null &&
turnoPersistente.getAnulado().equals(CTurno.TURNO_ANULADO)){
    throw new EventoException("El turno está anulado");
}
evento.setTurno(turnoPersistente);
idCirculoTurno = turnoPersistente.getIdCirculo();

//Restricciones:
//Se debe validar que el turno ingresado, pertenece al círculo del usuario, y
es un turno de certificado
if(!idCirculoUsuario.equals(idCirculoTurno) &&
!idCirculoUsuario.equals(turnoPersistente.getSolicitud().getCirculo().getIdCirculo())){
    throw new ImpresionException("El círculo del usuario no
corresponde con el círculo del turno.");
}

```

```

//SIR-98
//validar que el turno no sea de orden nacional

if(!turnoPersistente.getIdCirculo().equals(turnoPersistente.getSolicitud().getCirculo().getIdCirculo())){
    throw new ImpresionException("No se permiten reimpresion de certificados de orden nacional");
}

//Restricciones:
//debe ser un turno de certificado
Solicitud solicitud = turnoPersistente.getSolicitud();
numImpresionesSolicitadas = evento.getNumeroImpresiones();

if(!(solicitud instanceof SolicitudCertificado)){
    throw new ImpresionException("El turno no es de certificados.");
}

solicitudCert = (SolicitudCertificado)solicitud;
numImpresionesCert = solicitudCert.getNumImpresiones();

idProceso = new Long(CProceso.PROCESO_CERTIFICADOS);
oid = new ProcesoPk();
oid.idProceso = idProceso.longValue();

proceso = hermod.getProcesoById(oid);

roles = evento.getRoles();
itRoles = roles.iterator();

maxImprUsuario = 0;
maxNumImpresiones = 0;

while(itRoles.hasNext()){
    rol = (Rol)itRoles.next();
    //TFS 4588: EL USUARIO CON ROL REGISTRADOR PUEDE REIMPRIMIR LAS VECES QUE DESEE
    if(rol.getRoleId().equals(CRol.SIR_ROL_REGISTRADOR)){
        esUsuarioRegistrador = true;
    }
    maxNumImpresiones =
hermod.getNumeroMaximoImpresiones(rol,proceso);

    if (maxImprUsuario<maxNumImpresiones){
        maxImprUsuario=maxNumImpresiones;
    }
}

/*
 * @author : Julio Alcázar Rivas
 * @change : se elimino del if la validacion de esUsuarioRegistrador para restringir el numero de impresiones a los usuarios
 * que posean el rol CRol.SIR_ROL_REGISTRADOR
 * Caso Mantis : 02359
 */
if(numImpresionesSolicitadas > solicitudCert.getNumeroCertificados()){
    throw new ImpresionException("No se puede reimprimir más veces del número de certificados solicitados.");
}

//Validacion que el certificado a imprimir no sea EXENTO
/*
 * @author : Julio Alcázar Rivas
 * @change : Se definió el objeto liquidacion global al metodo
 * Caso Mantis : 02359
 */
liquidacion = new LiquidacionTurnoCertificado();
List liquidaciones = solicitud.getLiquidaciones();
for(int i=0;i<liquidaciones.size();i++){

```

```

        double id = new
Double(((LiquidacionTurnoCertificado)liquidaciones.get(i)).getIdLiquidacion()).doubleValue();
        if(id==solicitud.getLastIdLiquidacion()){
            liquidacion =
(LiquidacionTurnoCertificado)liquidaciones.get(i);
        }
    }

/*
 * @author : Julio Alcázar Rivas
 * @change : Modificar la validacion de tipos de certificados
 * Inmediatos EvnImprimirFolio.IMPRIMIR_CERTIFICADO y Pertenencia
EvnImprimirFolio.IMPRIMIR_CERTIFICADO_PERTENENCIA
 * Caso Mantis : 02359
 */

if(evento.getTipoEvento().equals(EvnImprimirFolio.IMPRIMIR_CERTIFICADO_PERTENENCIA)){

if(!(liquidacion.getTipoCertificado().getIdTipoCertificado().equals(CTipoCertificado.TIPO_PERTENENCIA
_ID))){
    throw new ImpresionException("No se puede reimprimir el
oficio de pertenencia. El certificado no es de Pertenencia");
}

/* @author : Julio Alcázar Rivas
 * @change : Se valida que el turno se encuentre en la fase de entrega
CFase.CER_ENTREGAR o CFase.FINALIZADO
 * Caso Mantis : 02359
 */
if(!turnoPersistente.getIdFase().equals(CFase.CER_ENTREGAR) &&
!turnoPersistente.getIdFase().equals(CFase.FINALIZADO)){
    throw new ImpresionException("El Turno debe estar en Fase de ENTREGA
CERTIFICADOS o FINALIZADO");
}

/*
 * @author : Julio Alcázar Rivas
 * @change : Se elimina la variable boolean certPertenenciaSinFolio
 * Caso Mantis : 02359
 */
asociado
*****
try{
    folio = new Folio();
    //***** En caso de que el certificado sea de pertenencia y no tenga
    * un matricula no se imprime el certificado y no se lanza la excepcion
    folio = this.getFolioFromTurno(turnoPersistente);
    if (folio == null){
        throw new ImpresionException("Error al imprimir el folio: el
folio no ha podido cargarse.");
    }
}catch(EventoException excepcion){
    throw excepcion;
}
/*
 * @author : Julio Alcázar Rivas
 * @change : Se verifican los dias habiles para la reimpresion de un certificado
 * Caso Mantis : 02359
 */
TurnoPk idTurno = new TurnoPk();
idTurno.anio = turnoPersistente.getAnio();
idTurno.idCirculo = turnoPersistente.getIdCirculo();
idTurno.idProceso = turnoPersistente.getIdProceso();
idTurno.idTurno = turnoPersistente.getIdTurno();

if((liquidacion.getTipoCertificado().getIdTipoCertificado().equals(CTipoCertificado.TIPO_INMEDIATO_ID
))){
    this.DiasHabilesCertificado(turnoPersistente, circuloPk,

```

```

        CFase.CER_SOLICITUD, idTurno);
    }else{
        this.DiasHabilresCertificado(turnoPersistente, circuloPk,
turnoPersistente.getIdFase(),idTurno);
    }
    /*
     * @author : Julio Alcázar Rivas
     * @change : Se agrega if para solo reimprimir los certificados inmediatos y los oficios
de pertencia
     * Caso Mantis : 02359
    */
if(!evento.getTipoEvento().equals(EvnImprimirFolio.IMPRIMIR_CERTIFICADO_PERTENENCIA)){

    List solicitudesPadres = solicitudCert.getSolicitudesPadres();
    SolicitudAsociada solAsociada = null;
    Solicitud solPadre = null;

    if (solicitudesPadres != null && solicitudesPadres.size() > 0) {
        solAsociada = (SolicitudAsociada) solicitudesPadres.get(0);
        solPadre = solAsociada.getSolicitudPadre();
    }

    boolean validarExento = true;

    // El turno pertenece a un turno de Registro
    if( solPadre != null && solPadre instanceof SolicitudRegistro){
        validarExento = false;
    }

    if (liquidacion.getTipoTarifa()!=null && validarExento) {
        if(liquidacion.getTipoTarifa().equals(CTipoTarifa.EXENTO)){
            throw new ImpresionException("No se puede reimprimir un
certificado con tipo de tarifa EXENTO.");
        }
    }

    /*if(solicitudCert.getNumImpresiones() >
solicitudCert.getNumeroCertificados()){
        throw new ImpresionException("El certificado ya fue impreso el
número de copias solicitadas. No puede reimprimirse.");
    }*/
}

if(validarImpresionesAnteriores){
    Rol cajero = new Rol();
    cajero.setRolId(CRoles.CAJERO_CERTIFICADOS);
    cajero.setNombre(CRoles.CAJERO_CERTIFICADOS);

    Rol impEspecial = new Rol();
    impEspecial.setRolId(CRoles.REIMPRESION_ESPECIAL);
    impEspecial.setNombre(CRoles.REIMPRESION_ESPECIAL);

    Proceso procesoCert = new Proceso();
    procesoCert.setIdProceso(new
Long(CProceso.PROCESO_CERTIFICADOS).longValue());

    procesoCert.setNombre(CProceso.NOMBRE_PROCESO_CERTIFICADOS);

    int numeroImpresionesCajero =
getNumMaxImpresiones(cajero,procesoCert);
    int numeroImpresionesImpEsp =
getNumMaxImpresiones(impEspecial,procesoCert);
    int totalImpAnteriores = numeroImpresionesCajero +
numeroImpresionesImpEsp;

    if( solicitudCert.getNumImpresiones() < totalImpAnteriores ){
        throw new ImpresionException("No se puede reimprimir el
certificado, porque no se han terminado los intentos de reimpresión en el cajero y el usuario
}

```

```

        reimpresión especial.");
    }

}

/*
 * @author      : Julio Alcázar Rivas
 * @change      : se elimino del if la validacion de esUsuarioRegistrador para
restringir el numero de impresiones a los usuarios
 * que posean el rol CRol.SIR_ROL_REGISTRADOR y se agrega la validacion para la
fase CFase.FINALIZADO
 * Caso Mantis : 02359
*/
if(!validarImpresionesAnteriores){

    if (maxImprUsuario<=numImpresionesCert-1 &&
        (turnoPersistente.getIdFase().equals(CFase.CER_ENTREGAR)||
turnoPersistente.getIdFase().equals(CFase.FINALIZADO))){
        throw new ImpresionException("Se ha alcanzado el número
máximo("+maxImprUsuario+) de impresiones para esa solicitud");
    }

    String uid = evento.getUid();
}

//}

/*
 * @author      : Julio Alcázar Rivas
 * @change      : Se elimina el uso de la variable boolean certPertenenciaSinFolio
 * Caso Mantis : 02359
*/
//gov.sir.core.negocio.modelo.Usuario usuarioNeg =
evento.getUsuarioNeg();

if (forseti.enTramiteFolio(folio.getIdMatricula())){
    throw new ImpresionException("El folio esta en trámite.");
}

CirculoPk cid = new CirculoPk();
cid.idCirculo = turno.getIdCirculo();

FolioPk fid = new FolioPk();
fid.idMatricula = folio.getIdMatricula();

List padres=forseti.getFoliosPadre(fid);

List hijos=forseti.getFolioHijosEnAnotacionesConDireccion(fid);

String fechaImpresion= this.getFechaImpresion();
//obtener textos base de los separadores
String tbase1 = ",";
String tbase2 = "";
List variablesImprimibles =
hermod.getOPLookupCodes(COPLookupTypes.VARIABLES_IMPRIMIBLES);
for(Iterator i= variablesImprimibles.iterator(); i.hasNext();){
    OPLookupCodes op = (OPLookupCodes) i.next();
    if(op.getCodigo().equals(COPLookupCodes.TEXTO_BASE1)){
        tbase1= op.getValor();
    }
    if(op.getCodigo().equals(COPLookupCodes.TEXTO_BASE2)){
        tbase2 = op.getValor();
    }
}
ImprimibleCertificado impCerti = new
ImprimibleCertificado(folio,turnoPersistente,padres,hijos, fechaImpresion,
CTipoImprimible.CERTIFICADO_INMEDIATO, tbase1, tbase2);
impCerti.setPrintWatermarkEnabled(true);
impCerti.setUsuario(usuario);

String nombre = "";

```

```

String archivo = "";

String idCirculoFolio = folio.getCirculo();
impCerti =
this.getImprimibleCertificadoConFirmaDigital(impCerti,idCirculoFolio);
//llenar parametros hashtable
Hashtable tabla= new Hashtable();
tabla.put(Processor.IMPRIMIBLE,impCerti);
if(evento.getImpresoraSeleccionada() != null){
    tabla.put(Processor.IMPRESORA_SELECCIONADA,
evento.getImpresoraSeleccionada());
}
tabla.put(Processor.NUM_COPIAS_IMPRIMIBLE, new
Integer(evento.getNumeroImpresiones()));
tabla.put(Processor.ESPERA, CImpresion.DEFAULT_ESPERA_IMPRESION);
tabla.put(Processor.INTENTOS,
CImpresion.DEFAULT_INTENTOS_IMPRESION);
tabla.put(Processor.IMPRIMIBLE,impCerti);

//obtener imprimible base
ImprimibleBase imprimibleb =
(ImprimibleBase)tabla.get(Processor.IMPRIMIBLE);
boolean
resultadoImpr=imprimirAdministrador(circulo.getIdCirculo(),imprimibleb,tabla);

if (resultadoImpr){
    TurnoHistoria turnoHist=new TurnoHistoria();
    turnoHist.setUsuario(usuario);
    turnoHist.setUsuarioAtiende(usuario);
    turnoHist.setFase(CFase.CER_IMPRIMIR);
    turnoHist.setIdTurno(turno.getIdWorkflow());

    turnoHist.setNumeroCopiasReimpresion(evento.getNumeroImpresiones());
    turnoHist.setIdMatriculaImpresa(folio.getIdMatricula());
    hermod.addTurnoHistoriaToTurno(turnoPersistente,turnoHist);

    solicitudCert.setNumImpresiones(numImpresionesCert+1);
    Solicitud resultadoSolicitud = null;
    resultadoSolicitud = hermod.updateSolicitudCertificado(solicitudCert);
    if (resultadoSolicitud == null){
        Log.getInstance().error(ANImprimirFolio.class,"Error en el método
que actualiza la solicitud");
    }
}

/*
 * @author : Julio Alcázar Rivas
 * @change : ELSE en el cual se realiza la reimpresion de los oficios de pertenencia
 * Caso Mantis : 02359
 */
} else {
    imprimirOficio(evento, folio, turnoPersistente, solicitudCert);
}

//this.imprimir(impCerti,uid);

//se manda a imprimir el recibo por el identificador unico de usuario

//Se agrega la nota informativa al turno.
String notaId = evento.getTipoNota();
String notaDescripcion = evento.getDescripcionNota();

TipoNota tipoNota = new TipoNota();
tipoNota.setIdTipoNota(notaId);

Nota notaAgregada = new Nota();
notaAgregada.setTiponota(tipoNota);
notaAgregada.setDescripcion(notaDescripcion);
notaAgregada.setAnio(turnoPersistente.getAnio());
notaAgregada.setIdFase("CER_REIMPRESION_CERTIFICADOS");

```

```

        notaAgregada.setIdCirculo(idCirculoUsuario);
        notaAgregada.setIdProceso(1);
        notaAgregada.setTime(new Date());
        notaAgregada.setUsuario(usuario);

        hermod.addNotaToTurno(notaAgregada, idTurno);

    } catch (Throwable t) {
        t.printStackTrace();
        throw new ImpresionException("Error al imprimir el folio: " +
t.getMessage());
    }

    EvnRespImprimirFolio evnRespuesta =
        new EvnRespImprimirFolio(usuarios,
EvnRespImprimirFolio.IMPRESION_FOLIO_OK);
    return evnRespuesta;
}

```

**Método:**

```
private boolean imprimirOficio(EvnImprimirFolio evento, Folio folio, Turno turnoPersistente, SolicitudCertificado solicitudCert) throws ImpresionException{
```

**Código Línea 1159:**

```

private boolean imprimirOficio(EvnImprimirFolio evento) throws ImpresionException{
    Turno turno = evento.getTurno();
    Circulo circulo = evento.getCirculo();
    TurnoHistoria turnoHistoria;
    List oficios;
    try{

        TurnoPk turnoPk = new TurnoPk();
        turnoPk.anio = turno.getAnio();
        turnoPk.idCirculo = turno.getIdCirculo();
        turnoPk.idProceso = turno.getIdProceso();
        turnoPk.idTurno = turno.getIdTurno();
        oficios = hermod.getOficiosTurno(turnoPk);
        turnoHistoria = obtenerTurnoHistoriaActual(evento);
    }catch(Throwable e){
        throw new ImpresionException(e.getMessage());
    }
    boolean existeOficio = false;
    Oficio oficio = new Oficio();
    for(Iterator iter = oficios.iterator() ; iter.hasNext() ;){
        oficio = (Oficio)iter.next();

        if(oficio.getTurnoHistoria().getFase().equals(turnoHistoria.getFaseAnterior())){
            existeOficio = true;
            break;
        }else{
            existeOficio = false;
        }
    }
    if(!existeOficio){
        throw new ImpresionException("No existe un oficio para imprimir");
    }
    ImprimibleOficioPertenencia imprimible = new
ImprimibleOficioPertenencia(oficio.getTextoOficio(),CTipoImprimible.OFICIO);
    Hashtable tabla= new Hashtable();
    tabla.put(Processor.IMPRIMIBLE,imprimible);
    if(evento.getImpresoraSeleccionada()!=null){
        tabla.put(Processor.IMPRESORA_SELECCIONADA,
evento.getImpresoraSeleccionada());
    }
    tabla.put(Processor.NUM_COPIAS_IMPRIMIBLE, new
Integer(evento.getNumeroImpresiones()));
    tabla.put(Processor.ESPERA, CImpresion.DEFAULT_ESPERA_IMPRESION);
}

```

```

        tabla.put(Processor.INTENTOS, CImpresion.DEFAULT_INTENTOS_IMPRESION);
        tabla.put(Processor.IMPRIMIBLE,imprimible);
        return(imprimirAdministrador(circulo.getIdCirculo(),imprimible,tabla));
    }

Por:
/*
 * @author : Julio Alcázar Rivas
 * @change : se cambia la signature del metodo para que reciba Folio, Turno y
SolicitudCertificado.
 * Ademas se realizaron diferentes modificaciones al comportamiento del metodo como:
 * a) Permitir reimpresiones en la fase finalizado CFase.FINALIZADO
 * b) Agregar la reimpresion en el historial del turno
 * Caso Mantis : 02359
 */
private boolean imprimirOficio(EvnImprimirFolio evento, Folio folio, Turno
turnoPersistente, SolicitudCertificado solicitudCert)throws ImpresionException{
    Turno turno = evento.getTurno();
    Circulo circulo = evento.getCirculo();
    TurnoHistoria turnoHistoria;
    List oficios;
    Usuario usuario = null;
    boolean resultadoImpr = false;
    try{
        usuario = fenrir.getUsuario(evento.getUsuario());
        TurnoPk turnoPk = new TurnoPk();
        turnoPk.anio = turno.getAnio();
        turnoPk.idCirculo = turno.getIdCirculo();
        turnoPk.idProceso = turno.getIdProceso();
        turnoPk.idTurno = turno.getIdTurno();
        oficios = hermod.getOficiosTurno(turnoPk);
        turnoHistoria = obtenerTurnoHistoriaActual(evento);

        boolean existeOficio = false;
        Oficio oficio = new Oficio();
        for(Iterator iter = oficios.iterator() ; iter.hasNext() ;){
            oficio = (Oficio)iter.next();

            if(oficio.getTurnoHistoria().getFase().equals(turnoHistoria.getFaseAnterior()) ||
turnoHistoria.getFase().equals(CFase.FINALIZADO) ){
                existeOficio = true;
                break;
            }
        }
        if(!existeOficio){
            throw new ImpresionException("No existe un oficio para imprimir");
        }
        ImprimibleOficioPertenencia imprimible = new
        ImprimibleOficioPertenencia(oficio.getTextoOficio(),CTipoImprimible.OFICIO);
        Hashtable tabla= new Hashtable();
        tabla.put(Processor.IMPRIMIBLE,imprimible);
        if(evento.getImpresoraSeleccionada()!=null){
            tabla.put(Processor.IMPRESORA_SELECCIONADA,
            evento.getImpresoraSeleccionada());
        }
        tabla.put(Processor.NUM_COPIAS_IMPRIMIBLE, new
        Integer(evento.getNumeroImpresiones()));
        tabla.put(Processor.ESPERA, CImpresion.DEFAULT_ESPERA_IMPRESION);
        tabla.put(Processor.INTENTOS,
        CImpresion.DEFAULT_INTENTOS_IMPRESION);
        tabla.put(Processor.IMPRIMIBLE,imprimible);

        resultadoImpr=imprimirAdministrador(circulo.getIdCirculo(),imprimible,tabla);

        if (resultadoImpr){
            TurnoHistoria turnoHist=new TurnoHistoria();
            turnoHist.setUsuario(usuario);
            turnoHist.setUsuarioAtiende(usuario);
            turnoHist.setFase(CFase.CER_IMPRIMIR);
            turnoHist.setIdTurno(turno.getIdWorkflow());
            turnoHist.setNumeroCopiasReimpresion(evento.getNumeroImpresiones());
        }
    }
}

```

```

        turnoHist.setIdMatriculaImpresa(folio.getIdMatricula());
        hermod.addTurnoHistoriaToTurno(turnoPersistente,turnoHist);

        Solicitud resultadoSolicitud = null;
        resultadoSolicitud =
    hermod.updateSolicitudCertificado(solicitudCert);
        if (resultadoSolicitud == null){
            Log.getInstance().error(ANImprimirFolio.class,"Error en el
método que actualiza la solicitud");
        }
    }
}catch(Throwable e){
    throw new ImpresionException(e.getMessage());
}
return resultadoImpr;
}

```

#### NUEVO Método:

```
private void DiasHabilesCertificado(Turno turno, CirculoPk circuloPk,
String fase, TurnoPk idTurno) throws EventoException, Throwable{
```

#### Código Línea 1199:

```

/*
 * @author : Julio Alcázar Rivas
 * @change : DiasHabilesCertificado metodo que verifica los dias habiles por tipo de certificado
 * Caso Mantis : 02359
 */
private void DiasHabilesCertificado(Turno turno, CirculoPk circuloPk, String fase, TurnoPk idTurno)
throws EventoException, Throwable{
    int numDiasHabiles = 0;
    List turnosHistoria = turno.getHistorials();
    for (Iterator iter = turnosHistoria.iterator(); iter.hasNext(); ) {
        TurnoHistoria turnoHistoria = (TurnoHistoria) iter.next();
        if (turnoHistoria.getFase().equals(fase)) {
            Calendar calendarsol = Calendar.getInstance();
            calendarsol.setTime(turnoHistoria.getFecha());
            calendarsol.add(Calendar.DATE, 1);
            Calendar calendaractual = Calendar.getInstance();
            calendaractual.setTime(new Date());

            while((calendarsol.compareTo(calendaractual) <= 0)){
                try {
                    if(!forseti.isFestivo(calendarsol.getTime(),circuloPk )){
                        numDiasHabiles++;
                    }
                } catch (ForsetiException e) {
                    Log.getInstance().error(AnTrasladoTurno.class,e.getMessage(), e);
                    throw new EventoException(e.getMessage(), e);
                } catch (Throwable e) {
                    Log.getInstance().error(AnTrasladoTurno.class,e.getMessage(), e);
                    throw new EventoException(e.getMessage(), e);
                }
                calendarsol.add(Calendar.DATE, 1);
            }
            break;
        }
    }
    if (hermod.getNuevoCambioMatriculaAuditoria(idTurno)){
        if(numDiasHabiles>5){
            throw new ImpresionException("El certificado solo puede ser impreso los 5 primeros
dias habiles siguientes a la fecha de la fase: "+fase );
        }
    }else
    {

```

```

        if(numDiasHabiles>1){
            throw new ImpresionException("El certificado solo puede ser impreso hasta el
siguiente dia habil apartir de la fase: "+fase );
        }
    }
}

```

**CLASE:**

**gov.sir.core.negocio.acciones.administracion.AnTrasladoTurno.java**

**Método:**

```

private EvnRespTrasladoTurno
consultarTurnoCertificadoCambioMatricula(EvnTrasladoTurno evento)
throws EventoException {

```

**Se cambio casi todo el comportamiento del metodo:**

```

private EvnRespTrasladoTurno consultarTurnoCertificadoCambioMatricula(EvnTrasladoTurno evento)
throws EventoException {

    Turno turno = null;
    Folio folio = null;
    List tiposNota = null;
    Circulo circulo = evento.getCirculo();
    org.auriga.core.modelo.transferObjects.Usuario user = evento.getUsuario();

    CirculoPk circuloPk = new CirculoPk();
    circuloPk.idCirculo = circulo.getIdCirculo();

    try {
        boolean puedeConsultarTurno = false;
        Circulo circuloTemp = null;

        turno = hermod.getTurnobyWF(evento.getTurno().getIdWorkflow());

        if (turno == null) {
            throw new TurnoNoEncontradoException("El turno solicitado no
existe.", null);
        }

        //SI SE QUIERE CONSULTAR UN TURNO DE OTRO CIRCULO

        if(!turno.getSolicitud().getCirculo().getIdCirculo().equals(circulo.getIdCirculo())
        &&
turno.getSolicitud().getCirculo().getIdCirculo().equals(turno.getIdCirculo())){
            throw new ConsultaTurnoExceptionException("No puede consultar
turnos de otros circulos.");
        }

        if
(!String.valueOf(turno.getIdProceso()).equals(CProceso.PROCESO_CERTIFICADOS)) {
            throw new ConsultaTurnoExceptionException("El turno debe ser de
Certificados.");
        }

        /*
         * @author : Julio Alcázar Rivas
         * @change : Los turnos deben estar en la fase CFase.CER_ENTREGAR or
CFase.FINALIZADO.
         * Caso Mantis : 02359
         */
        if (!turno.getIdFase().equals(CFase.CER_ENTREGAR) &&
!turno.getIdFase().equals(CFase.FINALIZADO)) {
            throw new ConsultaTurnoExceptionException("El turno debe estar
en la fase de Entrega Certificados o Finalizado.");
        }

        if
((!turno.getIdCirculo().equals(turno.getSolicitud().getCirculo().getIdCirculo()))){
            throw new ConsultaTurnoExceptionException("no se permiten
cambios de matricula en certificados de orden nacional");
        }
    }
}

```

```

        Solicitud solicitud = turno.getSolicitud();
        LiquidacionTurnoCertificado liquidacion = new LiquidacionTurnoCertificado();
        List liquidaciones = solicitud.getLiquidaciones();
        for(int i=0;i<liquidaciones.size();i++){
            double id = new
Double(((LiquidacionTurnoCertificado)liquidaciones.get(i)).getIdLiquidacion()).doubleValue();
            if(id==solicitud.getLastIdLiquidacion()){
                liquidacion =
(LiquidacionTurnoCertificado)liquidaciones.get(i);
            }
        }

if(((liquidacion.getTipoCertificado().getIdTipoCertificado().equals(CTipoCertificado.TIPO_ANTIGUO_SIS
TEMA_ID))||

(liquidacion.getTipoCertificado().getIdTipoCertificado().equals(CTipoCertificado.TIPO_ASOCIADO_ID)))
){
    if (!turno.getSolicitud().getSolicitudFolios().isEmpty()) {
        folio =
((SolicitudFolio)turno.getSolicitud().getSolicitudFolios().get(0)).getFolio();
        if (folio == null){
            throw new ConsultaTurnoExceptionException("El turno no tiene un folio
asociado.");
        }
    }else{
        folio = ((SolicitudFolio)turno.getSolicitud().getSolicitudFolios().get(0)).getFolio();
        if (folio == null){
            throw new ConsultaTurnoExceptionException("El turno no tiene un folio
asociado.");
        }
    }
    ProcesoPk id = new ProcesoPk();
    id.idProceso = 1;

    tiposNota =
hermod.getTiposNotaProceso(id,CFase.CER_REIMPRESION_ESPECIAL,false);

    /*
     * @author : Julio Alcázar Rivas
     * @change : Valida que la matricula solo pueda ser cambiada una sola vez, que
puede ser
     * en la fase CFase.CER_ENTREGAR o en la fase CFase.FINALIZADO.
     * Caso Mantis : 02359
     */
    TurnoPk turnoId = new TurnoPk();
    turnoId.anio = turno.getAnio();
    turnoId.idCirculo = turno.getIdCirculo();
    turnoId.idProceso = turno.getIdProceso();
    turnoId.idTurno = turno.getIdTurno();
    if (hermod.getNuevoMatriculaAuditoria(turnoId)){
        throw new ConsultaTurnoExceptionException("La matricula solo puede ser cambiada
una vez");
    }

    /*
     * @author : Julio Alcázar Rivas
     * @change : Verifica la fecha limite para cambiar la matricula a un turno de
certificado
     * Caso Mantis : 02359
     */
    List turnosHistoria = turno.getHistorials();
    for (Iterator iter = turnosHistoria.iterator(); iter.hasNext()) {
        TurnoHistoria turnoHistoria = (TurnoHistoria) iter.next();
        if (turnoHistoria.getFase().equals(CFase.CER_SOLICITUD)) {
            Calendar calendarsol = Calendar.getInstance();
            calendarsol.setTime(turnoHistoria.getFecha());
            calendarsol.add(Calendar.DATE, 1);
            Calendar calendaractual = Calendar.getInstance();

```

```

        calendaractual.setTime(new Date());

        int numDiasHabiles = 0;
        while((calendarsol.compareTo(calendaractual) <= 0)){
            try {
                if(!forseti.isFestivo(calendarsol.getTime(),circuloPk )){ 
                    numDiasHabiles++;
                }
            } catch (ForsetiException e) {
                Log.getInstance().error(AnTrasladoTurno.class,e.getMessage(), e);
                throw new EventoException(e.getMessage(), e);
            } catch (Throwable e) {
                Log.getInstance().error(AnTrasladoTurno.class,e.getMessage(), e);
                throw new EventoException(e.getMessage(), e);
            }
            calendarsol.add(Calendar.DATE, 1);
        }
        if(numDiasHabiles>5){
            throw new ConsultaTurnoExceptionException("La matrícula solo puede
ser cambiada en los 5 primeros días hábiles apartir de la radicación" );
        }
        break;
    }

}

} catch (HermodException e) {
    Log.getInstance().error(AnTrasladoTurno.class,"Ocurrió un error al consultar
el turno .", e);
    throw new TurnoNoEncontradoException(e.getMessage(), e);
} catch (ConsultaTurnoExceptionException e) {
    Log.getInstance().error(AnTrasladoTurno.class,"Ocurrió un error al consultar
el turno.", e);
    throw e;
} catch (TurnoNoEncontradoException e) {
    Log.getInstance().error(AnTrasladoTurno.class,"Ocurrió un error al consultar
el turno.", e);
    throw e;
} catch (Throwable e) {
    Log.getInstance().error(AnTrasladoTurno.class,"Ocurrió un error al consultar
el turno .", e);
    throw new TurnoNoEncontradoException(e.getMessage(), e);
}

EvnRespTrasladoTurno evRespuesta;
evRespuesta = new EvnRespTrasladoTurno(turno,
EvnRespTrasladoTurno.TURNO_CERTIFICADO_CAMBIO_MATRICULA );
evRespuesta.setTurno(turno);
evRespuesta.setFolio(folio);
evRespuesta.setListaTipoNota(tiposNota);

return evRespuesta;
}

```

**CLASE:**

**gov.sir.core.web.acciones.administracion.AWTrasladoTurno.java**

**Método:**

**private EvnCambioMatricula  
cambiarFolioTurnoCertificado(HttpServletRequest request) throws  
AccionWebException {**

**Código Línea 639:**

String matriculaActual = folio.getIdMatricula();

**Por:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Se verifica que el exista un folio asociado a la solicitud
 * Caso Mantis : 02359
 */
```

```
String matriculaActual = null;
if(folio != null){
    matriculaActual = folio.getIdMatricula();
}
```

**Método:**

```
public void doEnd(HttpServletRequest request, EventoRespuesta evento)
{
```

**Código Linea 1592:**

```
proceso.setIdProceso(1);
```

**Por:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : se cambia el idProceso a 10 el cual corresponde al proceso
 Reimpresion_Certificados
 * Caso Mantis : 02359
 */
proceso.setIdProceso(10);
```

**CLASE:**

```
gov.sir.core.web.acciones.certificado.ANCertificado.java
```

**Método:**

```
private EvnRespCambioMatricula cambiarMatricula(EvnCambioMatricula
evento) throws EventoException {
```

**Se cambio casi todo el comportamiento del metodo:**

```
private EvnRespCambioMatricula cambiarMatricula(EvnCambioMatricula evento) throws
EventoException {
```

```
String matriculaActual = evento.getMatriculaActual();
String matriculaNueva = evento.getMatriculaNueva();

Turno turno;
//validar la matricula
try {
    EvnValidacionMatricula evnVal = new
EvnValidacionMatricula(evento.getUsuario(), matriculaNueva, evento.getTipoCertificado());
    evnVal.setTurnoNacional(evento.isNacional());
    EvnRespValidacionMatricula evnRespVal = this.validarMatricula(evnVal);
} catch (ValidacionMatriculaCertificadoHTException val) {
    throw val;
} catch (EventoException ee) {
    throw new EventoException("Error validando la matrícula " + matriculaNueva
+ ee.getMessage(), ee);
}

TurnoPk tid = evento.getTurno_id();
/*
 * @author : Julio Alcázar Rivas
 * @change : se asigna a idWorkFlow los datos correspondientes desde tid
 * Caso Mantis : 02359
 */
String idWorkFlow = tid.anio + "-" + tid.idCirculo + "-" + tid.idProceso + "-" + tid.idTurno;

try {
    /*
     * @author : Julio Alcázar Rivas
     * @change : se realizan las validaciones para cambio de matricula en
isCambioMatriculaCertRegistro(idWorkFlow, matriculaNueva)
     * Caso Mantis : 02359
     */
    turno=hermod.getTurno(tid);
    if(turno.getSolicitud().getSolicitudesPadres() != null){
        ValidacionesSIRDAO validacion = new ValidacionesSIRDAO();
        if(!validacion.isCambioMatriculaCertRegistro(idWorkFlow, matriculaNueva)){
            throw new EventoException();
        }
    }
}
```

```

//cambiar la matricula
/*
 * @author    : Julio Alcázar Rivas
 * @change    : se verifica matriculaActual sea diferente a null
 * Caso Mantis : 02359
 */
if (matriculaActual != null){
    hermod.removeFolioFromTurno(matriculaActual, tid);
}else{
    matriculaActual = matriculaNueva;
}
hermod.addFolioToTurno(matriculaNueva, tid, evento.getUsuarioNeg());
//Se ingresa la auditoria del Cambio
hermod.addCambioMatriculaAuditoria(matriculaActual, matriculaNueva, tid,
evento.getUsuarioNeg());

//obtener de nuevo el turno para subirlo actualizado
turno=hermod.getTurno(tid);

/*
 * @author    : Julio Alcázar Rivas
 * @change    : Se ajustan las excepciones
 * Caso Mantis : 02359
 */
}catch (SQLException e) {
    Log.getInstance().error(ANCertificado.class,e);
    throw new ValidacionMatriculaCertificadoHTEception("Error cambiando la
matricula " + matriculaActual + " por la matrícula " + matriculaNueva + "
"+Utilidades.mensajeErrorOracle(e.getMessage(), "ORA-20001"));
}
catch(EventoException e){
    throw new ValidacionMatriculaCertificadoHTEception("Error cambiando la matricula "
+ matriculaActual + " por la matrícula " + matriculaNueva +": No cumple las condiciones de cambio de
matricula para turno de certificado asociado a un turno de registro",e);
}
catch (Throwable e) {

    Log.getInstance().error(ANCertificado.class,e);
    throw new ValidacionMatriculaCertificadoHTEception("Error cambiando la
matricula " + matriculaActual + " por la matrícula " + matriculaNueva, e);
}
Folio folio = null;
try {
    folio = forseti.getFolioByMatricula(matriculaNueva);
} catch (Throwable e1) {

    e1.printStackTrace();
    throw new ValidacionMatriculaCertificadoHTEception("Error obteniendo el
nuevo folio con matricula " + matriculaNueva, e1);
}

return new EvnRespCambioMatricula(matriculaNueva, folio, turno);
}
}

```

**CLASE:**

**gov.sir.hermod.dao.impl.jdogenie.JDOSolicitudesDAO.java**

**Método:**

**public SolicitudCertificado  
modificarSolicitudCertificado(SolicitudCertificado solCer) throws  
DAOException {**

**Código Insertado Linea 8055:**

```

/**
 * @author    : Julio Alcázar Rivas
 * Caso Mantis : 02359
 * @return true si sobre el turno se realizo un cambio de matricula
 * @param tID Identificador del turno.
 * @throws HermodException
 */

```

```

public boolean getCambioMatriculaAuditoria(TurnoPk tID ) throws DAOException {
    boolean rta = false;
    PersistenceManager pm = AdministradorPM.getPM();

    try {
        String idWorkFlow = tID.anio + "-" + tID.idCirculo + "-" + tID.idProceso + "-"
" + tID.idTurno;

        pm.currentTransaction().begin();

        Query query = pm.newQuery(CambioMatriculaCerEnhanced.class);
        query.declareParameters("String idWorkFlow");
        query.setFilter( "this.idWorkflow == idWorkFlow ");

        Collection col = (Collection) query.execute(idWorkFlow);

        Iterator iter = col.iterator();

        if (iter.hasNext()) {
            rta = true;
        }

        pm.currentTransaction().commit();

    } catch (JDOException e) {
        if (pm.currentTransaction().isActive()) {
            pm.currentTransaction().rollback();
        }
        Log.getInstance().error(JDOSolicitudesDAO.class,e.getMessage());
        throw new DAOException(e.getMessage(), e);
    } catch (Exception e) {
        if (pm.currentTransaction().isActive()) {
            pm.currentTransaction().rollback();
        }
        Log.getInstance().error(JDOSolicitudesDAO.class,e.getMessage());
        throw new DAOException(e.getMessage(), e);
    }
    finally {
        pm.close();
    }
    return rta;
}

```

**CLASE:**

**gov.sir.hermod.HermodException.java**

**Código Insertado Línea 1886:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Cadena de exception para metodo getCambioMatriculaAuditoria(TurnoPk tID)
 * Caso Mantis : 02359
 */
public static final String TURNO_NO_ENCONTRADO = "No fue posible obtener información del turno"

```

**CLASE:**

**gov.sir.hermod.HermodService.java**

**Código Insertado Linea 2140:**

```
* verifica que a el turno se le realizo un cambio de matricula
 * @param tID Identificador del turno.
 * @return true o false dependiendo de si se hizo cambio al turno.
 * @throws HermodException
 * @author    : Julio Alcázar Rivas
 * Caso Mantis : 02359
 */
public boolean getCambioMatriculaAuditoria(TurnoPk tID ) throws HermodException {
    try {
        return factory.getSolicitudesDAO().getCambioMatriculaAuditoria(tID);
    } catch (DAOException e) {
        HermodException he = new
HermodException(HermodException.TURNO_NO_ENCONTRADO, e);
        ExceptionPrinter ep = new ExceptionPrinter(he.fillInStackTrace());
        Log.getInstance().error(this.getClass(),he.getMessage(), e);
        throw he;
    }
}
```

**CLASE:****gov.sir.hermod.ejb.HermodServiceEJBBean.java****Código Insertado Linea 1005:**

```
/**
 * @author    : Julio Alcázar Rivas
 * Caso Mantis : 02359
 * @see
gov.sir.hermod.interfaz.HermodServiceInterface#getCambioMatriculaAuditoria(gov.sir.core.negocio.mo
dulo.Turno.TurnoPk)
 * @ejb:interface-method view-type="both"
 */
public boolean getCambioMatriculaAuditoria(TurnoPk tID ) throws HermodException {
    return hermod.getCambioMatriculaAuditoria(tID );
}
```

**CLASE:****gov.sir.hermod.interfaz.HermodServiceInterface.java****Código Insertado Linea 1232:**

```
/**
 * @author    : Julio Alcázar Rivas
 * Caso Mantis : 02359
 * verifica la asociación de un folio a un turno
 * @param tID Identificador del turno de registro.
 * @return true o false dependiendo de si se hizo cambio al turno.
 * @throws HermodException
 */
public boolean getCambioMatriculaAuditoria(TurnoPk tID ) throws Throwable;
```

**CLASE:****gov.sir.hermod.dao.SolicitudesDAO.java****Código Insertado Linea 92:**

```
/**
 * @author    : Julio Alcázar Rivas
 * Caso Mantis : 02359
 * verifica que a el turno se le realizo un cambio de matricula.
 * @param tID
 * @return <code>true</code> (cambio) o <code>false</code> (no cambio)
 * @throws DAOException
 */
public boolean getCambioMatriculaAuditoria(TurnoPk tID ) throws DAOException;
```

**➤ Modificacion En el proyecto GeneralSIR:****CREACION CLASE:****co.com.iridium.generalSIR.dao.util.Utilidades.java**

package co.com.iridium.generalSIR.dao.util;

/\*\*

```

 * Clase de Utilidades
 * @version 1.0, 08/11/2010
 * @author Julio Alcázar
 */
public class Utilidades {

    /**
     * Metodo que devuelve la cadena de error definida por el programador
     * en una función o procedimiento Oracle
     * Caso Mantis : 02359
     * @return <code>String</code>
     * @author Julio Alcázar
     */
    public static String mensajeErrorOracle(String msj, String error){

        String []cadena = msj.split(error);
        if (cadena != null){
            return cadena[1].split("ORA")[0].substring(2);
        }
        return msj;
    }
}

```

**CLASE:**

**co.com.iridium.generalSIR.negocio.validaciones.ValidacionesSIR.java**

**Código Insertado Línea 32:**

```

    /**
     * Metodo para determinar si la matricula de un turno de certificado puede ser cambiada.
     * @return <code>boolean</code>
     * Caso Mantis : 02359
     * @author Julio Alcazar
     */
    public boolean isCambioMatriculaCertRegistro(String turno, String matricula) throws
GeneralSIRException{
    ValidacionesSIRDAO validacionesSIRDAO = new ValidacionesSIRDAO();
    try {
        return validacionesSIRDAO.isCambioMatriculaCertRegistro(turno, matricula);
    } catch (SQLException ex) {
        throw new GeneralSIRException("Ocurrio un error al tratar de realizar "
            + "la validación del turno: " + turno + " :
"+Utilidades.mensajeErrorOracle(ex.getMessage(), "ORA-20001"),ex);
    }
}

```

**CLASE:**

**co.com.iridium.generalSIR.dao.ValidacionesSIRDAO.java**

**Código Insertado Linea 48:**

```
/*
 * Metodo que consulta el objeto de bd
PCK_UTILIDADES_VALIDACIONES.isCambioMatriculaCertRegistro
 * para determinar si la matricula de un turno de certificado puede ser cambiada.
 * @return <code>true</code> si puede ser cambiada; <code>false</code> si no
 * cumple las condiciones para ser cambiada.
 * Caso Mantis : 02359
 * @author Julio Alcazar
 */
public boolean isCambioMatriculaCertRegistro(String turno, String matricula) throws
SQLException{
    CallableStatement cs = null;
    StringBuilder sb = new StringBuilder();
    try {
        sb.append("BEGIN ? := PCK_UTILIDADES_VALIDACIONES.IS_CAMBIO_MATRICULA_CERT
(?,?); END;");
        cs = conexionBaseDatos.prepareCall(sb.toString());
        cs.registerOutParameter (1, Types.INTEGER);
        cs.setString(2, turno);
        cs.setString(3, matricula);
        cs.execute();
        return (cs.getInt(1)==1);

    }
    catch (SQLException e) {
        throw new SQLException(e.getMessage(), e);
    }finally{
        cerrarConexion(cs,null,null);
    }
}
```

**CLASE:****gov.sir.hermod.dao.impl.jdogenie.JDOVariablesOperativasDAO.java****Código Linea 1592:**

```
if (nombre.equals(CRoles.CAJERO_CERTIFICADOS)) {
```

**Por:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : se agrega al if CRoles.SIR_ROL_REIMPRESION_CERTIFICADOS para
obtener el maximo numero de impresiones del rol
 * Caso Mantis : 02359
 */
if (nombre.equals(CRoles.CAJERO_CERTIFICADOS) ||
nombre.equals(CRoles.SIR_ROL_REIMPRESION_CERTIFICADOS)) {
```

**237 Traslado de folios entre oficinas de registro****REQUERIMIENTO****TRASLADO DE FOLIOS ENTRE OFICINAS DE REGISTRO**

Teniendo en cuenta el procedimiento de traslado de folios, que hasta el momento se ha realizado en las Oficinas de Registro que operan con el aplicativo FOLIO MAGNETICO y que fue implementado en el aplicativo SIR, se requiere verificar y actualizar, de ser necesario, esta funcionalidad en el aplicativo SIR, de acuerdo a las siguientes especificaciones y a lo contenido en la Resolución 9177 del 11 de Octubre de 2010, mediante la cual “se adopta el procedimiento para el traslado de folios de matricula inmobiliaria y convalidación de turnos asignados a documentos públicos cuyas matriculas correspondan a Oficinas de Registro diferentes por competencia territorial”:

TRASLADO DE FOLIOS ENTRE ORIPS QUE OPERAN CON SIR.

1. Determinar en las ORIPs origen, los números de matricula a trasladar (ORIP).
2. Generar en las ORIPs origen, el acto administrativo relacionado con el traslado de los folios. Incluyendo toda la documentación como son los antecedentes, folio de cartulina (si existe) e impresión simple del folio (ORIP).
3. Establecidos los folios a trasladar, los cuales deben estar en estado definitivo (no temporales), se debe proceder a:
  - a) Pueden ser objeto de traslado uno o un rango de folios.
  - b) Crear o adicionar en las ORIPs destino, los códigos de zona registral (circulo, departamento, municipio y vereda). Debe validarse, que si existe una zona registral ya asignada en el circulo registral destino, no es necesario la creación de la misma. Ejemplo: El municipio de Piamonte y sus veredas, se encuentra asociadas al círculo registral de Bolívar, y se requiere trasladar folios a la ORIP de Popayán, por lo tanto, se debe asociar al círculo registral de Popayán, este municipio y por ende las veredas.
  - c) De acuerdo a la cantidad de folios a trasladar, reservar en la ORIP destino, el secuencial de matriculas requerido. Modificar en la ORIP destino, el dato de ultimo folio, para que no se utilicen los números de matricula reservados en esa ORIP.
  - d) Para los ciudadanos asociados a las anotaciones de los folios a trasladar, se debe asignar un nuevo secuencial de ID\_CIUDADANO, para los folios trasladados en la ORIP destino
  - e) Para los folios de matrícula de la ORIP origen, debe trasladarse toda la información del folio (encabezado, dirección, complementación, linderos, anotaciones, personas, salvedades, cancelaciones) y así mismo, debe generarse una copia en medio magnético de la información que contiene cada folio (archivos tipo export (dmp) de cada matricula).

NOTA: La información de la tabla folios derivados, no se trasladara a la ORIP destino, dado que no en todos los casos al trasladar un folio, se trasladan sus segregados. Esta información, en caso de ser requerida en la ORIP destino, debe ser adiciona en los folios trasladados a través del modulo de correcciones.

- f) Una vez realizada la copia, se debe proceder a generar nuevos registros en todas las tablas relacionadas con la información del folio a trasladar, asignándole el nuevo número de matrícula del secuencial reservado en la ORIP destino.
- g) Incluir en la tabla de trasladados (histórico), el número de matricula del folio trasladado de la ORIP origen y el nuevo número de matrícula asignado en la ORIP destino.
- h) Modificar, el estado de los folios y anotaciones a trasladar de la ORIP origen, con el estado “T-Trasladado” y garantizar que no se puedan realizar trámites ni consultas sobre estos folios.
- i) Generar un reporte de las matriculas trasladadas y el nuevo número de matrícula

<p>asignada.</p> <p>j) El sistema debe permitir realizar consulta del historial de matriculas trasladadas, en los casos que se requiere validar esta información.</p> <p>4. Generar en las ORIPs destino, el acto administrativo que indique cual es el nuevo numero asignado de los predios trasladados (ORIP).</p> <p><b>TRASLADO DE FOLIOS DE ORIPS QUE OPERAN CON SIR A FOLIO.</b></p> <ol style="list-style-type: none"> <li>1. Determinar en las ORIPs origen, los números de matricula a trasladar (ORIP).</li> <li>2. Generar en las ORIPs origen, el acto administrativo relacionado con el traslado de los folios. Incluyendo toda la documentación como son los antecedentes, folio de cartulina (si existe) e impresión simple del folio (ORIP).</li> <li>3. Establecidos los folios a trasladar, los cuales deben estar en estado definitivo (no temporales), se debe proceder a:             <ol style="list-style-type: none"> <li>a) Pueden ser objeto de traslado uno o un rango de folios.</li> <li>b) Para los folios de matrícula de la ORIP origen, debe trasladarse toda la información del folio (encabezado, dirección, complementación, linderos, anotaciones, personas, salvedades, cancelaciones) y así mismo, debe generarse una copia en medio magnético de la información que contiene cada folio (archivos tipo export (dmp) de cada matricula).</li> </ol> <p>NOTA: La información de la tabla folios derivados, no se trasladara a la ORIP destino, dado que no en todos los casos al trasladar un folio, se trasladan sus segregados. Esta información, en caso de ser requerida en la ORIP destino, debe ser adiciona en los folios trasladados a través del modulo de correcciones.</p> <ol style="list-style-type: none"> <li>c) Incluir en la tabla de trasladados (histórico), el número de matrícula del folio trasladado de la ORIP origen y para el nuevo número de matrícula asignado en la ORIP destino, se incluirá el texto “TRASLADADO A FOLIO”.</li> <li>d) Modificar, el estado de los folios y anotaciones a trasladar de la ORIP origen, con el estado “T-Trasladado” y garantizar que no se puedan realizar trámites ni consultas sobre estos folios.</li> <li>e) Generar un reporte de las matriculas trasladadas y el nuevo número de matrícula asignada.</li> <li>f) El sistema debe permitir realizar consulta del historial de matriculas trasladadas, en los casos que se requiere validar esta información.</li> </ol> <p>4. Generar en las ORIPs destino, el acto administrativo que indique cual es el nuevo numero asignado de los predios trasladados (ORIP).</p> <p>El ítem III de la Resolución 9177, correspondiente a CONVALIDACION DE TURNOS,</p> </li> <td style="width: 10%;"></td></ol>	
--	--

no será objeto de desarrollo en este requerimiento.

### **CORREO 13/12/2010 - ETB**

Imelda, buenos días,

El requerimiento de traslado de folios, se encuentra en la fase de desarrollo. La solución tiene dos componentes: Paquete PL!SQL y Forma en SIR. Para los dos casos se tiene el siguiente análisis del requerimiento:

#### **TRASLADO DE FOLIOS ENTRE OFICINAS DE REGISTRO**

Teniendo en cuenta el procedimiento de traslado de folios, que hasta el momento se ha realizado en las Oficinas de Registro que operan con el aplicativo FOLIO MAGNETICO y que fue implementado en el aplicativo SIR, se requiere verificar y actualizar, de ser necesario, esta funcionalidad en el aplicativo SIR, de acuerdo a las siguientes especificaciones y a lo contenido en la Resolución 9177 del 11 de Octubre de 2010, mediante la cual “se adopta el procedimiento para el traslado de folios de matrícula inmobiliaria y convalidación de turnos asignados a documentos públicos cuyas matrículas correspondan a Oficinas de Registro diferentes por competencia territorial”:

#### **TRASLADO DE FOLIOS ENTRE ORIPS QUE OPERAN CON SIR.**

Determinar en las ORIPs origen, los números de matrícula a trasladar (ORIP).

Solución: Se construye una forma (nueva funcionalidad SIR que permite “tomar” matrículas para traslado, dejándolas en un almacenamiento temporal que permita la ejecución iterada de un paquete PL/SQL de traslado efectivo de los folios de matrícula.

Generar en las ORIPs origen, el acto administrativo relacionado con el traslado de los folios. Incluyendo toda la documentación como son los antecedentes, folio de cartulina (si existe) e impresión simple del folio (ORIP).

Anotación: Se trata de un procedimiento administrativo de la Orip, en el que SIR no interviene.

Establecidos los folios a trasladar, los cuales deben estar en estado definitivo (no temporales), se debe proceder a:

Regla de Negocio 1. No pueden ser “tomadas” matrículas en estado NO definitivo por la forma de traslado.

Mensaje de Error: La matrícula no está en estado Definitivo, y no puede ser trasladada.

Pueden ser objeto de traslado uno o un rango de folios.

Solución: Para que la forma pueda hacer una evaluación previa de las reglas de negocio, antes de tomar el folio para el traslado, no se utilizará un rango de folios; solamente se usará el proceso folio a folio. Un rango de folios implica un log de proceso que debe ser leído con posterioridad al proceso, mientras el proceso folio-a-folio permite enviar el mensaje de error adecuado al usuario en la interface.

Crear o adicionar en las ORIPs destino, los códigos de zona registral (circulo, departamento, municipio y vereda). Debe validarse, que si existe una zona registral ya

asignada en el círculo registral destino, no es necesario la creación de la misma. Ejemplo: El municipio de Piamonte y sus veredas, se encuentra asociadas al círculo registral de Bolívar, y se requiere trasladar folios a la ORIP de Popayán, por lo tanto, se debe asociar al círculo registral de Popayán, este municipio y por ende las veredas.

Regla de Negocio 2. Se debe verificar la existencia de la zona registral en destino. Si no está creada, el procedimiento PL/SQL la creara en tiempo de ejecución.

De acuerdo a la cantidad de folios a trasladar, reservar en la ORIP destino, el secuencial de matriculas requerido. Modificar en la ORIP destino, el dato de ultimo folio, para que no se utilicen los números de matricula reservados en esa ORIP.

Regla de Negocio 3. La creación de nuevos número de matrícula en el destino seguirá las mismas normas de creación de SIR: 1. Se verifica en la tabla SIR\_NE\_MAT\_NO\_GRABADA, que no exista un numero de matricula disponible para la Orip correspondiente a la transacción en curso. Si no se tiene una matrícula en SIR\_NE\_MAT\_NO\_GRABADA, se genera un nuevo valor con base en la secuencia almacenada en la tabla SIR\_NE\_CIRCULO atributo CRCL\_LAST\_ID\_FOLIO.

Para los ciudadanos asociados a las anotaciones de los folios a trasladar, se debe asignar un nuevo secuencial de ID\_CIUDADANO, para los folios trasladados en la ORIP destino Regla de Negocio 4. Se verificará con la llave tipo\_documento, numero\_documento, con excepción del tipo SE- Secuencia, la existencia del ciudadano en el destino. Si se obtiene coincidencia en tipo\_documento, numero\_documento, se verificará el nombre completo (nombre, apellido1, apellido2) y si se obtiene 100% de coincidencia en la comparación se utilizará el ID\_CIUDADANO del destino; de lo contrario se creará un nuevo

ID\_CIUDADANO en destino. Esta regla será informada en la interface de usuario como una información al momento de tomar la matrícula para traslado. Si la dupla tipo\_documento, numero\_documento es coincidente, pero el nombre completo NO, se creará un nuevo ID\_CIUDADANO con la regla de creación para secuencia en el destino.

Mensaje Informativo: Se creará un nuevo ID\_CIUDADANO en la Orip de destino para la anotación: xx

Mensaje Informativo: NO creará un nuevo ID\_CIUDADANO en la Orip de destino para la anotación: xx debido a que ya existe el ciudadano registrado con tipo, número de identificación y nombre completo en el destino

Para los folios de matrícula de la ORIP origen, debe trasladarse toda la información del folio (encabezado, dirección, complementación, linderos, anotaciones, personas, salvedades, cancelaciones) y así mismo, debe generarse una copia en medio magnético de la información que contiene cada folio (archivos tipo export (dmp) de cada matrícula).

Anotación: La generación de archivos en formato .dmp para cada matrícula no es adecuada, toda vez que genera procesos administrativos de base de datos que no pueden ser ejecutados por la interface de forma automatizada. Por lo anterior, se adiciona el atributo TRSL\_FOLIO\_XML en la tabla SIR\_NE\_TRASLADO, en donde se almacenará una estructura XML con el 100% de la información del folio origen.

NOTA: La información de la tabla folios derivados, no se trasladará a la ORIP destino, dado que no en todos los casos al trasladar un folio, se trasladan sus segregados. Esta información, en caso de ser requerida en la ORIP destino, debe ser adicionada en los folios trasladados a través del modulo de correcciones.

Regla de Negocio 5. No se traslada el conjunto de folios segregados.

Una vez realizada la copia, se debe proceder a generar nuevos registros en todas las tablas relacionadas con la información del folio a trasladar, asignándole el nuevo número de matrícula del secuencial reservado en la ORIP destino.

Anotación: Ya está contemplada en la regla de negocio 3.

Incluir en la tabla de traslados (histórico), el número de matrícula del folio trasladado de la ORIP origen y el nuevo número de matrícula asignado en la ORIP destino.

Regla de Negocio 6. Se escribe la información de la tabla SIR\_NE\_TRASLADO, incluyendo la estructura XML del folio origen.

Modificar, el estado de los folios y anotaciones a trasladar de la ORIP origen, con el estado “T-Trasladado” y garantizar que no se puedan realizar trámites ni consultas sobre estos folios.

Generar un reporte de las matriculas trasladadas y el nuevo número de matrícula asignada. Solución: Se creara el reporte FOLIOS TRASLADADOS, el cual recibirá el parámetro OripOrigen, OripDestino, FechaInicio, y FechaFinal el cual reportara la información de la tabla SIR\_NE\_TRASLADO; para el set de datos que cumpla el criterio de selección.

El sistema debe permitir realizar consulta del historial de matriculas trasladadas, en los casos que se requiere validar esta información.

Anotación: El reporte FOLIOS TRASLADADOS cumple también este propósito

Generar en las ORIPs destino, el acto administrativo que indique cual es el nuevo numero asignado de los predios trasladados (ORIP).

Anotación: Se trata de un procedimiento administrativo de la Orip, en el que SIR no interviene.

#### TRASLADO DE FOLIOS DE ORIPS QUE OPERAN CON SIR A FOLIO.

Determinar en las ORIPs origen, los números de matricula a trasladar (ORIP).

Anotación: Folio se encuentra fuera del alcance de la operación actualmente, y por tanto esta verificación no es posible

Generar en las ORIPs origen, el acto administrativo relacionado con el traslado de los folios. Incluyendo toda la documentación como son los antecedentes, folio de cartulina (si existe) e impresión simple del folio (ORIP).

Anotación: Se trata de un procedimiento administrativo de la Orip, en el que SIR no interviene

Establecidos los folios a trasladar, los cuales deben estar en estado definitivo (no temporales), se debe proceder a:

Pueden ser objeto de traslado uno o un rango de folios.

Solución: La interface nueva para traslado de folios permite tomar uno a uno los folios de matrícula, y solamente se verificará la Regla de Negocio 1.

Para los folios de matrícula de la ORIP origen, debe trasladarse toda la información del

folio (encabezado, dirección, complementación, linderos, anotaciones, personas, salvedades, cancelaciones) y así mismo, debe generarse una copia en medio magnético de la información que contiene cada folio (archivos tipo export (dmp) de cada matrícula).

Solución: Se adiciona el atributo TRSL\_FOLIO\_XML en la tabla SIR\_NE\_TRASLADO, en donde se almacenara una estructura XML con el 100% de la información del folio origen

NOTA: La información de la tabla folios derivados, no se trasladara a la ORIP destino, dado que no en todos los casos al trasladar un folio, se trasladan sus segregados. Esta información, en caso de ser requerida en la ORIP destino, debe ser adiciona en los folios trasladados a través del modulo de correcciones.

Anotación: No es aplicable a Folio por no tener conexión al sistema folio para intervenir un repositorio externo a SIR.

Incluir en la tabla de trasladados (histórico), el número de matrícula del folio trasladado de la ORIP origen y para el nuevo número de matrícula asignado en la ORIP destino, se incluirá el texto “TRASLADADO A FOLIO”.

Anotación: No es aplicable a Folio por no tener conexión al sistema folio para intervenir un repositorio externo a SIR.

Modificar, el estado de los folios y anotaciones a trasladar de la ORIP origen, con el estado “T-Trasladado” y garantizar que no se puedan realizar trámites ni consultas sobre estos folios.

Solución: Se actualizara la tabla SIR\_NE\_FOLIO en el atributo con el valor

Generar un reporte de las matriculas trasladadas y el nuevo número de matrícula asignada.

Anotación: No es aplicable a Folio por no tener conexión al sistema folio para intervenir un repositorio externo a SIR.

El sistema debe permitir realizar consulta del historial de matriculas trasladadas, en los casos que se requiere validar esta información.

Anotación: No es aplicable a Folio por no tener conexión al sistema folio para intervenir un repositorio externo a SIR.

Generar en las ORIPs destino, el acto administrativo que indique cual es el nuevo numero asignado de los predios trasladados (ORIP).

El ítem III de la Resolución 9177, correspondiente a CONVALIDACION DE TURNOS, no será objeto de desarrollo en este requerimiento.

De igual forma, se adicionaron los siguientes condicionantes, en reunión técnica de la cual te informe telefónicamente:

Buenos Días,

Atentamente envío los puntos que se trataron en la reunión interna para definir el proceso de traslado de folios.

- |  |  |
|--|--|
| <p>1. Se diseñará una sola pantalla Administrativa para el proceso de traslado de folios, Uno o Muchos.</p> <p>2. Cuando se va a trasladar el ciudadano se debe verificar que se encuentre en la Orip destino, se debe validar que el ciudadano sea idéntico al usuario origen. Si no se encuentra en el destino se debe insertar con un id nuevo.</p> <p>3. El consecutivo de matriculas, seguirá las mismas reglas de creación de matriculas de SIR.</p> <p>4. El comentario de cambio de estado se registrará en la tabla SIR.SIR_NE_ESTADO_HISTORIA, en la columna STHS_COMENTARIO.</p> <p>5. No se podrán realizar trámites ni consultas sobre las matriculas que se encuentren en estado trasladado.</p> |  |
|--|--|

Cordialmente,

GUILLERMO ANDRÉS CABRERA MORALES.

También he sostenido dos charlas informales sobre este requerimiento en las que te he informado que el Ingeniero Guillermo Cabrera tiene asignada la construcción del paquete PL!SQL; mientras el Ingeniero Julio Alcazar está a cargo de la construcción de la nueva forma en la interface SIR; por tal motivo, no comparto tu afirmación de no tener información sobre este requerimiento. La terminación de la tarea de desarrollo está programada para el día 21 de diciembre, día en el cual se entregara al líder de calidad para las pruebas correspondientes y la inclusión en el despliegue que le corresponda en turno.

Es de aclarar, que se trata de una funcionalidad completamente nueva para el SIR, toda vez que la que se recibió por parte de ETB, no está desarrollada ni cumple con ninguna de las reglas inscritas en el requerimiento. Tan pronto se complete el proceso de integración de la nueva funcionalidad al despliegue en curso, enviaré información al respecto. También cabe anotar lo sensible que resulta el proceso que se está solicitando, amén de las reglas de negocio que son tocadas en el aplicativo respecto al cambio que se suscita.

### ***DESARROLLO***

Se desarrolló el paquete SIR.PCK\_TRASLADO\_FOLIO encargado de la funcionalidad para trasladar folios entre las ORIP que operan en SIR.

Script Bases de Datos: PACKAGE PCK\_TRASLADO\_FOLIO

Creación De Clases:

NINGUNA (el mismo formato de modificación clases o jsp)

Se modificaron los siguientes archivos:

Se desarrolló el paquete **SIR.PCK\_TRASLADO\_FOLIO** encargado de la funcionalidad para trasladar folios entre las ORIP que operan en SIR.

### ***DESCRIPCIÓN PAQUETE:***

FUNCTION FNC\_SEQ\_ZONA\_REGISTRAL RETURN NUMBER  
FUNCTION FNC\_TURNOS\_ACTIVOS  
FUNCTION FNC\_ID\_ZONA\_REGISTRAL  
FUNCTION FNC\_IS\_BLOQUEO  
FUNCTION FNC\_IS\_TEMPORAL  
FUNCTION FNC\_ID\_CIUDADANO  
FUNCTION FNC\_OBTENER\_XML  
PROCEDURE SP\_TRASLADO\_FOLIOS  
FUNCTION FNC\_SEQ\_ZONA\_REGISTRAL RETURN NUMBER

**Descripción Función FNC\_SEQ\_ZONA\_REGISTRAL RETURN NUMBER.**

Función encargada de retornar la secuencia para la tabla SIR.SIR\_NE\_ZONA\_REGISTRAL.

**Descripción Función FNC\_TURNOS\_ACTIVOS (P\_ID\_MATRICULA IN VARCHAR2) RETURN VARCHAR2.**

Función que permite determinar si la matricula que se pasa como parámetro tiene turnos activos de los procesos Registro y Correcciones. Retorna el turno con el cual la matricula se encuentra en trámite.

**Descripción Función FNC\_ID\_ZONA\_REGISTRAL (P\_ID\_ZONA\_ORIGEN VARCHAR2, P\_CIRCULO\_DESTINO VARCHAR2) RETURN NUMBER.**

Función que se encarga de crear la zona regstral, se verifica primero si esta ya existe si no se crea una nueva y retorna el id de la zona regstral.

**Descripción Función FNC\_IS\_BLOQUEO (P\_ID\_MATRICULA VARCHAR2) RETURN VARCHAR2.**

Función que retorna el turno bloqueante, se envía como parámetro la matricula.

**Descripción FUNCTION FNC\_IS\_TEMPORAL(P\_ID\_MATRICULA VARCHAR2) RETURN NUMBER.**

Función que retorna el valor de la columna flio\_temporal de la tabla SIR.SIR\_NE\_FOLIO, para determinar si se encuentra temporal o definitivo.

**Descripción función FNC\_ID\_CIUDADANO(P\_ID\_CIUDADANO NUMBER, P\_ID\_CIRCULO\_DESTINO VARCHAR2) RETURN NUMBER.**

Función encargada de crear el ciudadano en el círculo destino. Primero se verifica si existe y si es idéntico al ciudadano de origen si cumple esta condición se utiliza el mismo ciudadano de origen sino es idéntico se crea un ciudadano nuevo en el circulo destino. Retorna el id del ciudadano.

**Descripción de la función FNC\_OBTENER\_XML(P\_ID\_MATRICULA VARCHAR2) RETURN CLOB.**

Función que retorna un clob con la información en formato xml del folio que se pasa como parámetro.

**Descripción procedimiento SP\_TRASLADO\_FOLIOS(P\_ID\_FOLIO VARCHAR2, P\_CIRCULO\_ORIGEN VARCHAR2, P\_CIRCULO\_DESTINO VARCHAR2, P\_RESOLUCION VARCHAR2, P\_ID\_USUARIO VARCHAR2, P\_FOLIO\_TRASLADADO OUT VARCHAR2)**

Procedimiento encargado de trasladar el folio se pasan los siguientes parámetros:

**@P\_ID\_FOLIO** Id del folio que se va a trasladar.

**@P\_CIRCULO\_ORIGEN** Id del circulo origen.

**@P\_CIRCULO\_DESTINO** Id del circulo destino.

**@P\_RESOLUCION** Texto con la información de la resolución.

**@P\_ID\_USUARIO** Nombre de usuario que realiza el traslado.

**@P\_FOLIO\_TRASLADADO** Parámetro de salida que retorna el id del folio nuevo.

PACKAGE PCK\_TRASLADO\_FOLIO

Archivos Creados:

[PACKAGE PCK\_TRASLADO\_FOLIO.sql]

create or replace

```

PACKAGE PCK_TRASLADO_FOLIO
AS
*****
NAME: PCK_TRASLADO_FOLIO
PURPOSE: Paquete encargado de la funcionalidad para trasladar folios entre las ORIP que operan en SIR.

REVISIONS:
Ver Date Author Description
-----
1.0 07/12/2010 Guillermo Cabrera 1. Created this package.
*****
```

```

FUNCTION FNC_SEQ_ZONA_REGISTRAL RETURN NUMBER;
FUNCTION FNC_TURNOS_ACTIVOS(P_ID_MATRICULA IN VARCHAR2) RETURN VARCHAR2;
FUNCTION FNC_ID_ZONA_REGISTRAL(P_ID_ZONA_ORIGEN VARCHAR2, P_CIRCULO_DESTINO
VARCHAR2) RETURN NUMBER;
FUNCTION FNC_IS_BLOQUEO(P_ID_MATRICULA VARCHAR2) RETURN VARCHAR2;
FUNCTION FNC_IS_TEMPORAL(P_ID_MATRICULA VARCHAR2) RETURN NUMBER;
FUNCTION FNC_ID_CIUDADANO(P_ID_CIUDADANO NUMBER, P_ID_CIRCULO_DESTINO VARCHAR2)
RETURN NUMBER;
FUNCTION FNC_OBTENER_XML(P_ID_MATRICULA VARCHAR2) RETURN CLOB;
PROCEDURE SP_TRASLADO_FOLIOS(P_ID_FOLIO VARCHAR2, P_CIRCULO_ORIGEN VARCHAR2,
P_CIRCULO_DESTINO VARCHAR2, P_RESOLUCION VARCHAR2, P_ID_USUARIO
VARCHAR2, P_FOLIO_TRASLADADO OUT VARCHAR2);
END PCK_TRASLADO_FOLIO;
```

#### PACKAGE PCK\_TRASLADO\_FOLIO

```

Archivos Creados:
[PACKAGE BODY PCK_TRASLADO_FOLIO.sql]
create or replace
PACKAGE BODY PCK_TRASLADO_FOLIO
AS
*****
NAME: PCK_TRASLADO_FOLIO
PURPOSE: Paquete encargado de la funcionalidad para trasladar folios entre las ORIP que operan en SIR.
```

```

REVISIONS:
Ver Date Author Description
-----
1.0 07/12/2010 Guillermo Cabrera 1. Created this package.
*****
```

```

/*
   FUNCION ENCARGADA DE OBTENER LA SECUENCIA DE LA ZONA REGISTRAL
*/
FUNCTION FNC_SEQ_ZONA_REGISTRAL RETURN NUMBER
IS
      N_ID_ZONAREGISTRAL NUMBER:=0;
BEGIN
      SELECT SIR.SEQ_ZONA_REGISTRAL.NEXTVAL INTO
N_ID_ZONAREGISTRAL
      FROM DUAL;
      RETURN N_ID_ZONAREGISTRAL;
END FNC_SEQ_ZONA_REGISTRAL;

/*
   FUNCION ENCARGADA DE VERIFICAR SI LA MATRICULA QUE SE PASA
COMO PARAMETRO
      TIENE TURNOS ACTIVOS DE REGISTRO O CORRECCIONES
*/
FUNCTION FNC_TURNOS_ACTIVOS(P_ID_MATRICULA IN VARCHAR2) RETURN
VARCHAR2
IS
```

```

CURSOR C_TURNOS_ACTIVOS
IS
SELECT TU.ANIO, TU.ID_CIRCULO, TU.ID_PROCESO,
FROM SIR.SIR_OP_SOLICITUD_FOLIO SF
INNER JOIN SIR.SIR_OP_TURNO TU ON SF.ID_SOLICITUD =
TU.ID_TURNO
WHERE SF.ID_MATRICULA = P_ID_MATRICULA
AND TU.TRNO_FECHA_FIN IS NULL
AND TU.TRNO_DESCRIPCION IS NULL
AND TU.ID_PROCESO IN (6,3);

HAY_TURNOS_ACTIVOS NUMBER:=1;
TURNO_TRAMITE VARCHAR2(100):='0';
D_TURNOS_ACTIVOS C_TURNOS_ACTIVOS%ROWTYPE;
FASE VARCHAR2(200):=NULL;
MAX_FADE NUMBER:=0;
COUNT_CUR NUMBER:=0;
BEGIN
BEGIN
OPEN C_TURNOS_ACTIVOS;
LOOP
FETCH C_TURNOS_ACTIVOS INTO
D_TURNOS_ACTIVOS;
EXIT WHEN
C_TURNOS_ACTIVOS%NOTFOUND OR FASE = 'PENDIENTE';
MAX_FADE:=0;
TURNO_TRAMITE:='0';
BEGIN
SELECT
MAX(TH.id_turno_historia) INTO MAX_FADE
SIR.SIR_OP_TURNO_HISTORIA TH
TH.ANIO=D_TURNOS_ACTIVOS.ANIO
TH.ID_CIRCULO=D_TURNOS_ACTIVOS.ID_CIRCULO
TH.ID_PROCESO=D_TURNOS_ACTIVOS.ID_PROCESO
TH.ID_TURNO=D_TURNOS_ACTIVOS.ID_TURNO
AND TH.trhs_id_fase IN
('REG_CERTIFICADOS_ASOCIADOS','COR_MESA_CONTROL');
EXCEPTION WHEN
NO_DATA_FOUND THEN
MAX_FADE:=0;
END;
IF (MAX_FADE<>0)THEN
SELECT TH.trhs_id_fase INTO FASE
FROM SIR.SIR_OP_TURNO_HISTORIA TH
WHERE TH.ANIO=D_TURNOS_ACTIVOS.ANIO
AND TH.ID_CIRCULO=D_TURNOS_ACTIVOS.ID_CIRCULO
AND TH.ID_PROCESO=D_TURNOS_ACTIVOS.ID_PROCESO
AND TH.ID_TURNO=D_TURNOS_ACTIVOS.ID_TURNO
AND TH.ID_TURNO_HISTORIA = MAX_FADE;

IF (FASE = 'REG_CERTIFICADOS_ASOCIADOS') OR (FASE = 'COR_MESA_CONTROL')THEN
TURNO_TRAMITE:='0';
ELSE
TURNO_TRAMITE:=D_TURNOS_ACTIVOS.ANIO||'-'||D_TURNOS_ACTIVOS.ID_CIRCULO||-
'||D_TURNOS_ACTIVOS.ID_PROCESO||'-'||D_TURNOS_ACTIVOS.ID_TURNO;
RETURN TURNO_TRAMITE;
END IF;
ELSE

```

```

FASE:='PENDIENTE';
TURNO_TRAMITE:=D_TURNOS_ACTIVOS.ANIO||'-'||D_TURNOS_ACTIVOS.ID_CIRCULO||-
'||D_TURNOS_ACTIVOS.ID_PROCESO||'.'||D_TURNOS_ACTIVOS.ID_TURNO;
RETURN TURNO_TRAMITE;
END IF;

        END LOOP;
        COUNT_CUR := C_TURNOS_ACTIVOS%ROWCOUNT;
        CLOSE C_TURNOS_ACTIVOS;
EXCEPTION
    WHEN OTHERS THEN
        TURNO_TRAMITE:='0';
END;
RETURN      TURNO_TRAMITE;
END FNC_TURNOS_ACTIVOS;

/*
FUNCION ENCARGADA DE CREAR LA ZONA REGISTRAL, SI YA EXISTE EN LA BASE DE DATOS
RETORNA EL ID CORRESPONDIENTE.
*/
FUNCTION FNC_ID_ZONA_REGISTRAL(P_ID_ZONA_ORIGEN VARCHAR2,
P_CIRCULO_DESTINO VARCHAR2) RETURN NUMBER
IS
N_ID_ZONA_DESTINO NUMBER:=0;
BEGIN
BEGIN
SELECT ID_ZONA_REGISTRAL INTO N_ID_ZONA_DESTINO
FROM SIR.sir_ne_zona_registral
WHERE (ID_DEPARTAMENTO, ID_MUNICIPIO, ID_VEREDA) IN (SELECT ID_DEPARTAMENTO,
ID_MUNICIPIO, ID_VEREDA FROM SIR.sir_ne_zona_registral
WHERE ID_ZONA_REGISTRAL = P_ID_ZONA_ORIGEN)
AND ID_CIRCULO = P_CIRCULO_DESTINO;
EXCEPTION
WHEN NO_DATA_FOUND THEN
N_ID_ZONA_DESTINO:=0;
END;
IF (N_ID_ZONA_DESTINO=0) THEN
N_ID_ZONA_DESTINO := FNC_SEQ_ZONA_REGISTRAL();
BEGIN
INSERT INTO SIR.SIR_NE_ZONA_REGISTRAL(ID_ZONA_REGISTRAL, ID_CIRCULO,
ID_DEPARTAMENTO, ID_VEREDA, ID_MUNICIPIO,
ZNRG_FECHA_CREACION)
(SELECT
N_ID_ZONA_DESTINO,P_CIRCULO_DESTINO,ID_DEPARTAMENTO, ID_VEREDA, ID_MUNICIPIO, SYSDATE
FROM SIR.SIR_NE_ZONA_REGISTRAL
WHERE ID_ZONA_REGISTRAL = P_ID_ZONA_ORIGEN);
EXCEPTION
WHEN OTHERS THEN
N_ID_ZONA_DESTINO:=0;
RETURN N_ID_ZONA_DESTINO;
END;
ELSE
RETURN N_ID_ZONA_DESTINO;
END IF;
RETURN N_ID_ZONA_DESTINO;
END FNC_ID_ZONA_REGISTRAL;

/*
FUNCION ENCARGADA DE VERIFICAR SI EL FOLIO SE ENCUENTRA BLOQUEADO
*/
FUNCTION FNC_IS_BLOQUEO(P_ID_MATRICULA VARCHAR2) RETURN VARCHAR2
IS
V_TURNO_BLOQUEO VARCHAR2(100):='0';

BEGIN
BEGIN

```

```

SELECT B.BLFL_ID_WORKFLOW INTO V_TURNO_BLOQUEO
FROM SIR.SIR_REL_BLOQUEO_FOLIO B
WHERE B.ID_MATRICULA = P_ID_MATRICULA
AND B.BLFL_FECHA_DESBLOQUEO IS NULL
AND ROWNUM = 1;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        V_TURNO_BLOQUEO := '0';
        RETURN V_TURNO_BLOQUEO;
END;

RETURN V_TURNO_BLOQUEO;
END FNC_IS_BLOQUEO;

/*
FUNCION QUE RETORNA EL ESTADO DEL FOLIO.
*/
FUNCTION FNC_IS_TEMPORAL(P_ID_MATRICULA VARCHAR2) RETURN NUMBER
IS
N_FOLIO_TEMPORAL NUMBER:=0;
BEGIN
    SELECT F.FLIO_DEFINITIVO INTO N_FOLIO_TEMPORAL
    FROM SIR.SIR_NE_FOLIO F
    WHERE F.ID_MATRICULA = P_ID_MATRICULA;

    RETURN N_FOLIO_TEMPORAL;
END FNC_IS_TEMPORAL;

/*
FUNCION ENCARGADA DE CREAR EL CIUDADANO EN EL CIRCULO DESTINO
SE VERIFICA SI YA EXISTE Y SI ES IDENTICO AL ORIGEN, SI CUMPLE ESTA CONDICION
SE UTILIZA ESE CIUDADANO SINO ES IDENTICO SE CREA UN CIUDADANO NUEVO.
*/
FUNCTION FNC_ID_CIUDADANO(P_ID_CIUDADANO NUMBER, P_ID_CIRCULO_DESTINO VARCHAR2)
RETURN NUMBER
IS
CURSOR C_CIUDADANO(P_ID_CIUDADANO VARCHAR2)
IS
SELECT ID_CIUDADANO, CDDN_DOCUMENTO, CDDN_NOMBRE, CDDN_TIPO_DOCUMENTO,
       CDDN_APELLIDO1, CDDN_APELLIDO2, CDDN_TELEFONO, CDDN_SOLICITANTE, ID_CIRCULO
FROM SIR.SIR_NE_CIUDADANO
WHERE ID_CIUDADANO = P_ID_CIUDADANO;

D_CIUDADANO C_CIUDADANO%ROWTYPE;

N_ID_CIUDADANO NUMBER:=0;
N_NOMBRE VARCHAR2(500);
N_APELLIDO1 VARCHAR2(500);
N_APELLIDO2 VARCHAR2(500);

BEGIN
    N_ID_CIUDADANO :=0;
    OPEN C_CIUDADANO(P_ID_CIUDADANO);
    LOOP FETCH C_CIUDADANO INTO D_CIUDADANO;
        EXIT WHEN C_CIUDADANO%NOTFOUND;

        IF (D_CIUDADANO.CDDN_TIPO_DOCUMENTO='PS' OR
D_CIUDADANO.CDDN_TIPO_DOCUMENTO='CC' OR D_CIUDADANO.CDDN_TIPO_DOCUMENTO='NT' OR
D_CIUDADANO.CDDN_TIPO_DOCUMENTO='TI' OR D_CIUDADANO.CDDN_TIPO_DOCUMENTO
= 'CE' OR D_CIUDADANO.CDDN_TIPO_DOCUMENTO = 'NIT') THEN

            BEGIN
                SELECT ID_CIUDADANO, CDDN_NOMBRE, CDDN_APELLIDO1, CDDN_APELLIDO2
                INTO N_ID_CIUDADANO, N_NOMBRE, N_APELLIDO1,N_APELLIDO2
                FROM SIR.SIR_NE_CIUDADANO

```

```

    WHERE ID_CIRCULO = P_ID_CIRCULO_DESTINO
    AND CDDN_DOCUMENTO = D_CIUDADANO.CDDN_DOCUMENTO
    AND CDDN_TIPO_DOCUMENTO = D_CIUDADANO.CDDN_TIPO_DOCUMENTO
    AND NVL(CDDN_SOLICITANTE,0)=NVL(D_CIUDADANO.CDDN_SOLICITANTE,0);

    IF (NVL(N_NOMBRE,0) = NVL(D_CIUDADANO.CDDN_NOMBRE,0) AND
    NVL(N_APELLIDO1,0)=NVL(D_CIUDADANO.CDDN_APELLIDO1,0)
        AND NVL(N_APELLIDO2,0)=NVL(D_CIUDADANO.CDDN_APELLIDO2,0)) THEN
        RETURN N_ID_CIUDADANO;
    ELSE
        SELECT SIR.SEC_SIR_NE_CIUDADANO.NEXTVAL INTO N_ID_CIUDADANO
        FROM DUAL;

        INSERT INTO SIR.SIR_NE_CIUDADANO (ID_CIUDADANO, CDDN_DOCUMENTO,
CDDN_NOMBRE,
                           CDDN_TIPO_DOCUMENTO, CDDN_APELLIDO1, CDDN_APELLIDO2,
CDDN_TELEFONO,
                           CDDN_FECHA_CREACION, CDDN_SOLICITANTE, ID_CIRCULO)

VALUES(N_ID_CIUDADANO,N_ID_CIUDADANO,D_CIUDADANO.CDDN_NOMBRE,D_CIUDADANO.CDDN_TIPO_
DOCUMENTO,D_CIUDADANO.CDDN_APELLIDO1,
       D_CIUDADANO.CDDN_APELLIDO2, D_CIUDADANO.CDDN_TELEFONO, SYSDATE,
D_CIUDADANO.CDDN_SOLICITANTE, P_ID_CIRCULO_DESTINO);
        RETURN N_ID_CIUDADANO;
    END IF;

    EXCEPTION WHEN NO_DATA_FOUND THEN
        N_ID_CIUDADANO:=0;
    END;

    IF (N_ID_CIUDADANO<>0 ) THEN

        RETURN N_ID_CIUDADANO;
    ELSE
        SELECT SIR.SEC_SIR_NE_CIUDADANO.NEXTVAL INTO N_ID_CIUDADANO
        FROM DUAL;

        BEGIN
            INSERT INTO SIR.SIR_NE_CIUDADANO (ID_CIUDADANO, CDDN_DOCUMENTO,
CDDN_NOMBRE,
                           CDDN_TIPO_DOCUMENTO, CDDN_APELLIDO1, CDDN_APELLIDO2,
CDDN_TELEFONO,
                           CDDN_FECHA_CREACION, CDDN_SOLICITANTE, ID_CIRCULO)

VALUES(N_ID_CIUDADANO,D_CIUDADANO.CDDN_DOCUMENTO,D_CIUDADANO.CDDN_NOMBRE,D_CIUDAD
ANO.CDDN_TIPO_DOCUMENTO,D_CIUDADANO.CDDN_APELLIDO1,
       D_CIUDADANO.CDDN_APELLIDO2, D_CIUDADANO.CDDN_TELEFONO, SYSDATE,
D_CIUDADANO.CDDN_SOLICITANTE, P_ID_CIRCULO_DESTINO);
            RETURN N_ID_CIUDADANO;
        EXCEPTION
            WHEN OTHERS THEN
                N_ID_CIUDADANO:=0;
                RETURN N_ID_CIUDADANO;
            END;
        END IF;

    ELSIF (D_CIUDADANO.CDDN_TIPO_DOCUMENTO = 'SE') THEN

        SELECT SIR.SEC_SIR_NE_CIUDADANO.NEXTVAL INTO N_ID_CIUDADANO
        FROM DUAL;

        INSERT INTO SIR.SIR_NE_CIUDADANO (ID_CIUDADANO, CDDN_DOCUMENTO,
CDDN_NOMBRE,

```

```

        CDDN_TIPO_DOCUMENTO, CDDN_APELLIDO1, CDDN_APELLIDO2,
CDDN_TELEFONO,
        CDDN_FECHA_CREACION, CDDN_SOLICITANTE, ID_CIRCULO)

VALUES(N_ID_CIUDADANO,N_ID_CIUDADANO,D_CIUDADANO.CDDN_NOMBRE,D_CIUDADANO.CDDN_TIPO_
DOCUMENTO,D_CIUDADANO.CDDN_APELLIDO1,
        D_CIUDADANO.CDDN_APELLIDO2, D_CIUDADANO.CDDN_TELEFONO, SYSDATE,
D_CIUDADANO.CDDN_SOLICITANTE, P_ID_CIRCULO_DESTINO);

        RETURN N_ID_CIUDADANO;
    ELSE
        RAISE_APPLICATION_ERROR(-20001,'TIPO DOCUMENTO NO VALIDO PARA SIR:
'||D_CIUDADANO.CDDN_TIPO_DOCUMENTO);
        NULL;
    END IF;
END LOOP;
CLOSE C_CIUDADANO;
RETURN N_ID_CIUDADANO;
END FNC_ID_CIUDADANO;

/*
FUNCION ENCARGADA DE RETORNAR EL XML DEL FOLIO ORIGEN
ES DECIR UNA COPIA EN XML DE COMO SE TRASLADO EL FOLIO.
*/
FUNCTION FNC_OBTENER_XML(P_ID_MATRICULA VARCHAR2) RETURN CLOB
IS
HCTX DBMS_XMLGEN.CTXHANDLE;
XML CLOB;
SECUENCIA NUMBER;
CAD VARCHAR2(4000);
V_ID_MATRICULA VARCHAR2(100);
BEGIN
    V_ID_MATRICULA:=""||P_ID_MATRICULA||""";
    CAD:= 'select f.* ,cursor(select an.* , cursor(select ac.* from sir.sir_ne_anotacion_ciudadano ac
        where ac.id_matricula = an.id_matricula
        and ac.id_anotacion = an.id_anotacion
        order by to_number(ac.id_anotacion)) ANOTA_CIUDADANO,
        cursor(select ca.* from sir.sir_ne_rel_cancelacion ca
        where ca.id_matricula = an.id_matricula
        and ca.id_anotacion = an.id_anotacion
        order by to_number(ca.id_anotacion)) CANCELACION,
        cursor(select sa.* from sir.sir_ne_salvedad_an sa
        where sa.id_matricula = an.id_matricula
        and sa.id_anotacion = an.id_anotacion
        order by to_number(sa.id_anotacion)) SALVEDAD_ANOTACION
        from sir.sir_ne_anotacion an
        where an.id_matricula=f.id_matricula
        order by to_number(an.ntcn_orden)asc) ANOTACION,
        cursor(select d.* from sir.sir_ne_direccion d
        where d.id_matricula = f.id_matricula
        order by to_number(d.id_direccion)) DIRECCION,
        cursor(select sf.* from sir.sir_ne_salvedad_fo sf
        where sf.id_matricula=f.id_matricula
        order by to_number(sf.id_salvedad_fo)) SALVEDAD_FOLIO
        from sir.sir_ne_folio f
        where f.id_matricula = ''||V_ID_MATRICULA||''';

HCTX:= DBMS_XMLGEN.newcontext(CAD);
dbms_xmlgen.setrowsettag(HCTX, 'FOLIO');
DBMS_XMLGEN.setrowtag(HCTX, 'FOLIO_ROW');
XML := dbms_xmlgen.getxml(HCTX);
DBMS_XMLGEN.CLOSECONTEXT(HCTX);
RETURN XML;
END FNC_OBTENER_XML;

```

```

/*
PROCEDIMIENTO ENCARGADO DE TRASLADAR EL FOLIO SE
PASAN LOS SIGUIENTES PARAMETROS:

@P_ID_FOLIO ID DEL FOLIO QUE SE VA A TRASLADAR
@P_CIRCULO_ORIGEN ID DEL CIRCULO ORIGEN
@P_CIRCULO_DESTINO ID DEL CIRCULO DESTINO
@P_RESOLUCION TEXTO CON LA INFORMACION DE LA RESOLUCIÓN
@P_ID_USUARIO NOMBRE DE USUARIO EL CUAL REALIZA EL TRASLADO
@P_FOLIO_TRASLADADO PARAMETRO OUT QUE RETORNA EL ID DEL FOLIO NUEVO.
*/
PROCEDURE SP_TRNSLADO_FOLIOS(P_ID_FOLIO VARCHAR2, P_CIRCULO_ORIGEN
VARCHAR2, P_CIRCULO_DESTINO VARCHAR2, P_RESOLUCION VARCHAR2, P_ID_USUARIO
VARCHAR2, P_FOLIO_TRNSLADADO OUT VARCHAR2)
IS

CURSOR C_ANOTACIONES(P_ID_MATRICULA VARCHAR2)
IS
SELECT ID_ANOTACION, ID_MATRICULA, NTCN_COMENTARIO, NTCN_FECHA_REGISTRO,
NTCN_VALOR, ID_DOCUMENTO, ID_NATURALEZA_JURIDICA,
ID_TIPO_ANOTACION, NTCN_FECHA_RADICACION, NTCN_RADICACION, ID_ESTADO_AN,
NTCN_ORDEN, NTCN_ESPECIFICACION, NTCN_FECHA_CREACION,
NTCN_LAST_ID_SALVEDAD, ID_USUARIO, NTCN_LINK, NTCN_ORDEN_LPAD,
ID_USUARIO_TMP, NTCN_ID_ANOTA_MODIFICADA, ID_DATOS_ANT_SISTEMA,
NTCN_USUARIO_ANT
FROM SIR.SIR_NE_ANOTACION
WHERE ID_MATRICULA = P_ID_MATRICULA
ORDER BY TO_NUMBER(NTCN_ORDEN);

CURSOR C_ANOTA_CIUDADANOS(P_ID_MATRICULA VARCHAR2, P_ID_ANOTACION VARCHAR2)
IS
SELECT ID_ANOTACION, ID_CIUDADANO, ID_MATRICULA, NTCD_DESCRIPCION,
NTCD_PROPIETARIO,
NTCD_ROL_PERSONA, NTCD_FECHA_CREACION, NTCD_PORCENTAJE, ID_CIRCULO,
ID_NANOTACION,
ID_NCIUDADANO
FROM SIR.SIR_NE_ANOTACION_CIUDADANO
WHERE ID_MATRICULA = P_ID_MATRICULA
AND ID_ANOTACION = P_ID_ANOTACION;

CURSOR C_CANCELACIONES(P_ID_MATRICULA VARCHAR2, P_ID_ANOTACION VARCHAR2)
IS
SELECT CA.ID_ANOTACION, CA.ID_ANOTACION1, CA.ID_MATRICULA, CA.CNCL_DESCRIPCION,
CA.CNCL_FECHA_CREACION
FROM SIR.SIR_NE_REL_CANCELACION CA
WHERE CA.ID_MATRICULA = P_ID_MATRICULA
AND CA.ID_ANOTACION = P_ID_ANOTACION;

CURSOR C_SALVEDAD_AN(P_ID_MATRICULA VARCHAR2, P_ID_ANOTACION VARCHAR2)
IS
SELECT SA.ID_ANOTACION, SA.ID_MATRICULA, SA.ID_SALVEDAD_AN, SA.SLAN_DESCRIPCION,
SA.SLAN_FECHA_CREACION, SA.ID_USUARIO, SA.ID_USUARIO_TMP, SA.SLAN_RADICACION,
SA.SLAN_USUARIO_ANT
FROM SIR.SIR_NE_SALVEDAD_AN SA
WHERE SA.ID_MATRICULA = P_ID_MATRICULA
AND SA.ID_ANOTACION = P_ID_ANOTACION;

D_ANOTACIONES C_ANOTACIONES%ROWTYPE;
D_ANOTA_CIUDADANOS C_ANOTA_CIUDADANOS%ROWTYPE;
D_CANCELACIONES C_CANCELACIONES%ROWTYPE;
D_SALVEDAD_AN C_SALVEDAD_AN%ROWTYPE;

N_FOLIO_INICIAL NUMBER:=0;

```

```

N_FOLIO_FINAL NUMBER:=0;
N_TURNOS_ACTIVOS VARCHAR2(100);
N_FOLIO_TEMPORAL NUMBER:=0;
V_TURNO_BLOQUEO VARCHAR2(100):='0';
C_ESTADO_FOLIO CHAR:=NULL;
N_FOLIO_EXISTE VARCHAR2(100):='0';
N_LAST_IDESTADO NUMBER:=0;
V_COMENTARIO_ANTERIOR VARCHAR2(512);
N_ID_USUARIO NUMBER:=0;

EXCEP_TURNOS_ACTIVOS EXCEPTION;
EXCEP_FOLIO_BLOQUEADO EXCEPTION;
EXCEP_FOLIO_TMP EXCEPTION;
EXCEP_ZONA_REGISTRAL EXCEPTION;
EXCEP_ANOTA_CIUDADANO EXCEPTION;
EXCEP_FOLIO_NOEXISTE EXCEPTION;
EXCEP_FOLIO_TRASLADADO EXCEPTION;

N_ID_ZONA_REGISTRAL NUMBER:=0;
N_IS_ZONA_REGISTRAL NUMBER:=0;
N_LAST_ID_FOLIO NUMBER:=0;
SEQ_ID_DOC NUMBER:=0;
SEQ_ID_COMPLE NUMBER:=0;
N_ID_CIUDADANO NUMBER:=0;

V_ID_FOLIO VARCHAR2(100);
N_ID_DOCUMENTO NUMBER:=0;
N_ID_COMPLEMENTACION NUMBER:=0;
N_ID_DOC_ANOTACION NUMBER:=0;

N_ID_TRASLADO NUMBER:=0;

ERR_NUM NUMBER;
ERR_MSJ VARCHAR2(500);

FOLIO_XML CLOB;

V_COUNT_ID_MATRICULA NUMBER(2);

BEGIN
    SAVEPOINT TRANSACTION_TRASLADO;
    BEGIN
        SELECT NVL(F.ID_DOCUMENTO,0), NVL(F.ID_COMPLEMENTACION,0), F.ID_ESTADO,
        F.ID_MATRICULA,
        NVL(FLIO_LAST_ID_ESTADO_HISTORIA,0), SUBSTR(FLIO_COMENTARIO,1,512)
        INTO N_ID_DOCUMENTO, N_ID_COMPLEMENTACION,C_ESTADO_FOLIO,N_FOLIO_EXISTE,
        N_LAST_IDESTADO,V_COMENTARIO_ANTERIOR
        FROM SIR.SIR_NE_FOLIO F
        WHERE F.ID_MATRICULA = P_ID_FOLIO
        AND F.ID_CIRCULO = P_CIRCULO_ORIGEN;
    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            N_FOLIO_EXISTE:='0';
    END;

    IF (N_FOLIO_EXISTE='0') THEN
        RAISE EXCEP_FOLIO_NOEXISTE;
    ELSE
        N_TURNOS_ACTIVOS:=FNC_TURNOS_ACTIVOS(P_ID_FOLIO);
        IF (N_TURNOS_ACTIVOS<>'0') THEN
            RAISE EXCEP_TURNOS_ACTIVOS;
        ELSE
            IF (C_ESTADO_FOLIO='T') THEN
                RAISE EXCEP_FOLIO_TRASLADADO;
        END IF;
    END IF;
END;

```

```

    ELSE
        V_TURNO_BLOQUEO:=FNC_IS_BLOQUEO(P_ID_FOLIO);
        IF (V_TURNO_BLOQUEO<>'0') THEN
            RAISE EXCEP_FOLIO_BLOQUEADO;
        ELSE
            N_FOLIO_TEMPORAL:=FNC_IS_TEMPORAL(P_ID_FOLIO);
            IF(N_FOLIO_TEMPORAL=0) THEN
                RAISE EXCEP_FOLIO_TMP;
            ELSE
                SELECT F.ID_ZONA_REGISTRAL INTO N_ID_ZONA_REGISTRAL
                FROM SIR.SIR_NE_FOLIO F
                WHERE F.ID_MATRICULA =P_ID_FOLIO;

                N_IS_ZONA_REGISTRAL:=
                FNC_ID_ZONA_REGISTRAL(N_ID_ZONA_REGISTRAL,P_CIRCULO_DESTINO);

                IF (N_IS_ZONA_REGISTRAL = 0) THEN
                    RAISE EXCEP_ZONA_REGISTRAL;
                ELSE
                    /******REGLA TOMADA DEL PROCEDURE SG
                    /* Se verifica primero que haya primero un n??A?mero de matr??A?la disponible
                    en SIR_NE_MAT_NO_GRABADA
                    */

                    SELECT COUNT(ID_MAT_NO_GRABADA)
                    INTO N_LAST_ID_FOLIO
                    FROM SIR.SIR_NE_MAT_NO_GRABADA
                    WHERE ID_CIRCULO = P_CIRCULO_DESTINO
                    AND ROWNUM = 1;
                    /*
                     Si existe un folio, se toma ese numero de folio
                    */
                    IF V_ID_FOLIO <> 0 THEN
                        BEGIN
                            SELECT NVL(ID_MAT_NO_GRABADA,0)
                            INTO N_LAST_ID_FOLIO
                            FROM SIR.SIR_NE_MAT_NO_GRABADA
                            WHERE ID_CIRCULO = P_CIRCULO_DESTINO
                            AND ROWNUM = 1
                            ORDER BY ID_MAT_NO_GRABADA
                            FOR UPDATE;
                        EXCEPTION
                            WHEN NO_DATA_FOUND THEN
                                N_LAST_ID_FOLIO:=0;
                        END;
                        /*
                         Se verifica primero que la matricula no se haya utilizado primero
                        */
                        SELECT COUNT(ID_MATRICULA)
                        INTO V_COUNT_ID_MATRICULA
                        FROM SIR.SIR_NE_FOLIO
                        WHERE ID_MATRICULA = P_CIRCULO_DESTINO || '-' ||
                        TO_CHAR(N_LAST_ID_FOLIO);

                        /*
                         Si se utilizo la matricula ya, se usa el secuencial
                        */
                        IF V_COUNT_ID_MATRICULA <> 0 THEN
                            /*
                             Se elimina el numero de matricula
                            */
                            DELETE FROM SIR.SIR_NE_MAT_NO_GRABADA
                            WHERE ID_CIRCULO = P_CIRCULO_DESTINO
                            AND ID_MAT_NO_GRABADA = N_LAST_ID_FOLIO;
                END;

```

```

/*
Se obtiene el nuevo id del folio derivado
*/
SELECT CRCL_LAST_ID_FOLIO+1
INTO N_LAST_ID_FOLIO
FROM SIR.SIR_NE_CIRCULO
WHERE ID_CIRCULO = P_CIRCULO_DESTINO;

/*
Se actualiza el nuevo id
*/
UPDATE SIR.SIR_NE_CIRCULO
SET CRCL_LAST_ID_FOLIO = N_LAST_ID_FOLIO
WHERE ID_CIRCULO = P_CIRCULO_DESTINO;
END IF;
--DBMS_OUTPUT.PUT_LINE('ID_MAT_NO_GRABADA'||V_ID_FOLIO);
/*
Se elimina el numero de matricula
*/
DELETE FROM SIR.SIR_NE_MAT_NO_GRABADA
WHERE ID_CIRCULO = P_CIRCULO_DESTINO
AND ID_MAT_NO_GRABADA = N_LAST_ID_FOLIO;
ELSE
-- DBMS_OUTPUT.PUT_LINE(' NO Entro a mat no grabada');
/*
Se obtiene el nuevo id del folio derivado
*/
SELECT CRCL_LAST_ID_FOLIO+1
INTO N_LAST_ID_FOLIO
FROM SIR.SIR_NE_CIRCULO
WHERE ID_CIRCULO = P_CIRCULO_DESTINO;
/*
Se actualiza el nuevo id
*/
UPDATE SIR_NE_CIRCULO
SET CRCL_LAST_ID_FOLIO = N_LAST_ID_FOLIO
WHERE ID_CIRCULO = P_CIRCULO_DESTINO;
END IF;

/*FIN TOMADA DEL PROCEDURE SG*/

/*****************/
/*SELECT CI.CRCL_LAST_ID_FOLIO+1 INTO N_LAST_ID_FOLIO
FROM SIR.SIR_NE_CIRCULO CI
WHERE CI.ID_CIRCULO = P_CIRCULO_DESTINO;

UPDATE SIR.SIR_NE_CIRCULO CI
SET CI.CRCL_LAST_ID_FOLIO = N_LAST_ID_FOLIO
WHERE CI.ID_CIRCULO = P_CIRCULO_DESTINO;*/

IF(N_ID_DOCUMENTO<>0)THEN
    SELECT SIR.SEC_SIR_NE_DOCUMENTO.NEXTVAL INTO SEQ_ID_DOC FROM DUAL;
    INSERT INTO SIR.SIR_NE_DOCUMENTO
    (ID_DOCUMENTO,DCMN_FECHA,DCMN_NUMERO, ID_OFICINA_ORIGEN, ID_TIPO_DOCUMENTO,
     DCMN_FECHA_CREACION,DCMN_COMENTARIO,
     DCMN_OFICINA_INTERNACIONAL, ID_CIRCULO)
    (SELECT SEQ_ID_DOC, D.DCMN_FECHA, D.DCMN_NUMERO,
     D.ID_OFICINA_ORIGEN, D.ID_TIPO_DOCUMENTO, D.DCMN_FECHA_CREACION,
     D.DCMN_COMENTARIO, D.DCMN_OFICINA_INTERNACIONAL,
     P_CIRCULO_DESTINO
      FROM SIR.SIR_NE_DOCUMENTO D
      WHERE D.ID_DOCUMENTO=N_ID_DOCUMENTO);

```

```

    ELSE
        SEQ_ID_DOC:=NULL;
    END IF;

    IF(N_ID_COMPLEMENTACION<>0) THEN
        SELECT SIR.SIR_NE_COMPLEMENTACION.NEXTVAL INTO SEQ_ID_COMPLE
    FROM DUAL;
        INSERT INTO SIR.SIR_NE_COMPLEMENTACION
        (ID_COMPLEMENTACION,CMPL_DESCRIPCION, CMPL_FECHA_CREACION, CMPL_DESC_CONFLICTIVA)
        (SELECT SEQ_ID_COMPLE, C.CMPL_DESCRIPCION,
        C.CMPL_FECHA_CREACION, C.CMPL_DESC_CONFLICTIVA
        FROM SIR.SIR_NE_COMPLEMENTACION C
        WHERE C.ID_COMPLEMENTACION=N_ID_COMPLEMENTACION);
    ELSE
        SEQ_ID_COMPLE:=NULL;
    END IF;

    BEGIN
        INSERT INTO SIR.SIR_NE_FOLIO (ID_MATRICULA, FLIO_COD_CATASTRAL,
        ID_ESTADO, ID_TIPO_PREDIO, ID_ZONA_REGISTRAL, ID_COMPLEMENTACION, FLIO_FECHA_APERTURA,
        FLIO_DEFINITIVO, FLIO_LINDERO, FLIO_LAST_ID_ANOTACION,
        FLIO_LAST_ID_DIRECCION, FLIO_LAST_ID_SALVEDAD, FLIO_FECHA_CREACION,
        FLIO_COD_CATASTRAL_ANT, ID_DOCUMENTO, FLIO_RADICACION,
        FLIO_USUARIO_GRAB, FLIO_LAST_ID_ANOTACION_TMP, FLIO_LAST_ID_DIRECCION_TMP,
        FLIO_LAST_ID_SALVEDAD_TMP, FLIO_INCONSISTENTE, ID_USUARIO,
        FLIO_COMENTARIO, FLIO_ORDEN_LPAD,
        FLIO_LAST_ID_ESTADO_HISTORIA, FLIO_NOMBRE_LOTE, FLIO_USUARIO_ANT,
        FLIO_DIRECTO, ID_CIRCULO)
        (SELECT P_CIRCULO_DESTINO||'-'||N_LAST_ID_FOLIO, FLIO_COD_CATASTRAL,
        ID_ESTADO, ID_TIPO_PREDIO, N_IS_ZONA_REGISTRAL, SEQ_ID_COMPLE, FLIO_FECHA_APERTURA,
        FLIO_DEFINITIVO, FLIO_LINDERO, FLIO_LAST_ID_ANOTACION,
        FLIO_LAST_ID_DIRECCION, FLIO_LAST_ID_SALVEDAD, FLIO_FECHA_CREACION,
        FLIO_COD_CATASTRAL_ANT, SEQ_ID_DOC, FLIO_RADICACION,
        FLIO_USUARIO_GRAB, FLIO_LAST_ID_ANOTACION_TMP, FLIO_LAST_ID_DIRECCION_TMP,
        FLIO_LAST_ID_SALVEDAD_TMP, FLIO_INCONSISTENTE, ID_USUARIO,
        FLIO_COMENTARIO, FLIO_ORDEN_LPAD,
        FLIO_LAST_ID_ESTADO_HISTORIA, FLIO_NOMBRE_LOTE, FLIO_USUARIO_ANT,
        FLIO_DIRECTO, P_CIRCULO_DESTINO
        FROM SIR.SIR_NE_FOLIO
        WHERE ID_MATRICULA = P_ID_FOLIO);
    EXCEPTION
        WHEN OTHERS THEN
            ERR_NUM:=SQLCODE;
            ERR_MSJ:=SQLERRM;
            DBMS_OUTPUT.put_line('ERROR: ' || ERR_NUM || ERR_MSJ);
            RAISE;
    END;

    INSERT INTO SIR.SIR_NE_DIRECCION(ID_DIRECCION, ID_MATRICULA,
    DRCC_ESPECIFICACION, ID_EJE,
                ID_EJE1, DRCC_VALOR_EJE, DRCC_VALOR_EJE1,
    DRCC_FECHA_CREACION,
                ID_USUARIO, ID_USUARIO_TMP, DRCC_USUARIO_ANT,
    ID_CIRCULO)
    (SELECT D.ID_DIRECCION, P_CIRCULO_DESTINO||'-'||N_LAST_ID_FOLIO,
    D.DRCC_ESPECIFICACION, D.ID_EJE,
                D.ID_EJE1, D.DRCC_VALOR_EJE, D.DRCC_VALOR_EJE1,
    D.DRCC_FECHA_CREACION,
                D.ID_USUARIO, D.ID_USUARIO_TMP, D.DRCC_USUARIO_ANT,
    P_CIRCULO_DESTINO
    FROM SIR.SIR_NE_DIRECCION D
    WHERE D.ID_MATRICULA = P_ID_FOLIO);

```

```

        INSERT INTO SIR.SIR_NE_SALVEDAD_FO(ID_MATRICULA, ID_SALVEDAD_FO,
SLFO_DESCRIPCION, SFLO_FECHA_CREACION,
ID_USUARIO, ID_USUARIO_TMP, SFLO_RADICACION,
SFLO_USUARIO_ANT)
        (SELECT P_CIRCULO_DESTINO||'-'||N_LAST_ID_FOLIO, SF.ID_SALVEDAD_FO,
SF.SLFO_DESCRIPCION, SF.SFLO_FECHA_CREACION,
SF.ID_USUARIO, SF.ID_USUARIO_TMP, SF.SFLO_RADICACION,
SF.SFLO_USUARIO_ANT
         FROM SIR.SIR_NE_SALVEDAD_FO SF
        WHERE SF.ID_MATRICULA=P_ID_FOLIO);

        P_FOLIO_TRASLADADO:=P_CIRCULO_DESTINO||'-'||N_LAST_ID_FOLIO;

        SELECT SIR.SEC_SIR_NE_TRASLADO.NEXTVAL INTO N_ID_TRASLADO FROM DUAL;

        BEGIN
        SELECT ID_USUARIO INTO N_ID_USUARIO
        FROM SIR.SIR_OP_USUARIO
        WHERE SRIO_NOMBRE_DE_USUARIO = P_ID_USUARIO;
        EXCEPTION
        WHEN NO_DATA_FOUND THEN
        N_ID_USUARIO:=NULL;
        END;

        FOLIO_XML:=NULL;
        FOLIO_XML:=FNC_OBTENER_XML(P_ID_FOLIO);

        INSERT INTO SIR.SIR_NE_TRASLADO
        (ID_TRASLADO, ID_MATRICULA_ORIGEN, ID_MATRICULA_DESTINO,
         FECHA_TRASLADO, TRSL_RESOLUCION, TRSL_FOLIO_XML)
        VALUES(N_ID_TRASLADO, P_ID_FOLIO, P_CIRCULO_DESTINO||'-'||N_LAST_ID_FOLIO,
         SYSDATE, P_RESOLUCION, FOLIO_XML);

        INSERT INTO SIR.SIR_NE_ESTADO_HISTORIA(ID_ESTADO_HISTORIA,
        ID_MATRICULA, ID_ESTADO,
        STHS_FECHA_CREACION,
        ID_USUARIO, STHS_FECHA_REGISTRO, ID_ESTADO1,
        STHS_COMENTARIO, STHS_COMENTARIO_ANTERIOR)
        VALUES(N_LAST_IDESTADO+1, P_ID_FOLIO, 'T', SYSDATE, N_ID_USUARIO, SYSDATE,
        C_ESTADO_FOLIO,
        P_RESOLUCION, V_COMENTARIO_ANTERIOR);

        UPDATE SIR.SIR_NE_FOLIO F
        SET F.ID_ESTADO = 'T', FLIO_LAST_ID_ESTADO_HISTORIA = N_LAST_IDESTADO+1
        WHERE F.ID_MATRICULA = P_ID_FOLIO;

        OPEN C_ANOTACIONES(P_ID_FOLIO);
        LOOP FETCH C_ANOTACIONES INTO D_ANOTACIONES;
        EXIT WHEN C_ANOTACIONES%NOTFOUND;

        SELECT SIR.SEC_SIR_NE_DOCUMENTO.NEXTVAL INTO SEQ_ID_DOC FROM
DUAL;

        SELECT AN.ID_DOCUMENTO
        INTO N_ID_DOC_ANOTACION
        FROM SIR.SIR_NE_ANOTACION AN
        WHERE AN.ID_MATRICULA = D_ANOTACIONES.ID_MATRICULA
        AND AN.ID_ANOTACION = D_ANOTACIONES.ID_ANOTACION;

```

```

    INSERT INTO SIR.SIR_NE_DOCUMENTO
    (ID_DOCUMENTO, DCMN_FECHA, DCMN_NUMERO, ID_OFICINA_ORIGEN, ID_TIPO_DOCUMENTO,
     DCMN_FECHA_CREACION, DCMN_COMENTARIO,
     DCMN_OFICINA_INTERNACIONAL, ID_CIRCULO)
    (SELECT SEQ_ID_DOC, D.DCMN_FECHA, D.DCMN_NUMERO,
    D.ID_OFICINA_ORIGEN, D.ID_TIPO_DOCUMENTO, D.DCMN_FECHA_CREACION,
    D.DCMN_COMENTARIO, D.DCMN_OFICINA_INTERNACIONAL,
    P_CIRCULO_DESTINO
     FROM SIR.SIR_NE_DOCUMENTO D
     WHERE D.ID_DOCUMENTO=N_ID_DOC_ANOTACION);

    INSERT INTO SIR.SIR_NE_ANOTACION (ID_ANOTACION, ID_MATRICULA,
    NTCN_COMENTARIO, NTCN_FECHA_REGISTRO, NTCN_VALOR, ID_DOCUMENTO,
    ID_NATURALEZA_JURIDICA,
    ID_TIPO_ANOTACION, NTCN_FECHA_RADICACION, NTCN_RADICACION,
    ID_ESTADO_AN, NTCN_ORDEN, NTCN_ESPECIFICACION, NTCN_FECHA_CREACION,
    NTCN_LAST_ID_SALVEDAD, ID_USUARIO, NTCN_LINK, NTCN_ORDEN_LPAD,
    ID_USUARIO_TMP, NTCN_ID_ANOTA_MODIFICADA, ID_DATOS_ANT_SISTEMA,
    NTCN_USUARIO_ANT, ID_CIRCULO)
    VALUES(D_ANOTACIONES.ID_ANOTACION, P_CIRCULO_DESTINO||'-
    '||IN_LAST_ID_FOLIO, D_ANOTACIONES.NTCN_COMENTARIO,
    D_ANOTACIONES.NTCN_FECHA_REGISTRO, D_ANOTACIONES.NTCN_VALOR, SEQ_ID_DOC, D_ANOTACIONE
    S.ID_NATURALEZA_JURIDICA, D_ANOTACIONES.ID_TIPO_ANOTACION,
    D_ANOTACIONES.NTCN_FECHA_RADICACION,
    D_ANOTACIONES.NTCN_RADICACION, D_ANOTACIONES.ID_ESTADO_AN, D_ANOTACIONES.NTCN_ORDEN,
    D_ANOTACIONES.NTCN_ESPECIFICACION, D_ANOTACIONES.NTCN_FECHA_CREACION,
    D_ANOTACIONES.NTCN_LAST_ID_SALVEDAD,
    D_ANOTACIONES.ID_USUARIO, D_ANOTACIONES.NTCN_LINK,
    D_ANOTACIONES.NTCN_ORDEN_LPAD, D_ANOTACIONES.ID_USUARIO_TMP,
    D_ANOTACIONES.NTCN_ID_ANOTA_MODIFICADA, D_ANOTACIONES.ID_DATOS_ANT_SISTEMA,
    D_ANOTACIONES.NTCN_USUARIO_ANT, P_CIRCULO_DESTINO);

    OPEN
    C_ANOTA_CIUDADANOS(D_ANOTACIONES.ID_MATRICULA, D_ANOTACIONES.ID_ANOTACION);
    LOOP FETCH C_ANOTA_CIUDADANOS INTO D_ANOTA_CIUDADANOS;
        EXIT WHEN C_ANOTA_CIUDADANOS%NOTFOUND;

    N_ID_CIUDADANO:=FNC_ID_CIUDADANO(D_ANOTA_CIUDADANOS.ID_CIUDADANO, P_CIRCULO_DESTINO);
    IF (N_ID_CIUDADANO=0)THEN
        RAISE EXCEP_ANOTA_CIUDADANO;
    ELSE
        INSERT INTO SIR.SIR_NE_ANOTACION_CIUDADANO(ID_ANOTACION,
        ID_CIUDADANO, ID_MATRICULA,
        NTCD_DESCRIPCION, NTCD_PROPIETARIO,
        NTCD_ROL_PERSONA, NTCD_FECHA_CREACION,
        NTCD_PORCENTAJE, ID_CIRCULO)
        VALUES(D_ANOTA_CIUDADANOS.ID_ANOTACION,
        N_ID_CIUDADANO, P_CIRCULO_DESTINO||'-'||IN_LAST_ID_FOLIO,
        D_ANOTA_CIUDADANOS.NTCD_DESCRIPCION,
        D_ANOTA_CIUDADANOS.NTCD_PROPIETARIO,
        D_ANOTA_CIUDADANOS.NTCD_ROL_PERSONA, SYSDATE,
        D_ANOTA_CIUDADANOS.NTCD_PORCENTAJE,
        P_CIRCULO_DESTINO);
    END IF;
    END LOOP;
    CLOSE C_ANOTA_CIUDADANOS;

    OPEN
    C_CANCELACIONES(D_ANOTACIONES.ID_MATRICULA, D_ANOTACIONES.ID_ANOTACION);
    LOOP FETCH C_CANCELACIONES INTO D_CANCELACIONES;
        EXIT WHEN C_CANCELACIONES%NOTFOUND;

```

```

    INSERT INTO SIR.SIR_NE_REL_CANCELACION(ID_ANOTACION,
                                           ID_ANOTACION1, ID_MATRICULA, CNCL_DESCRIPCION,
                                           CNCL_FECHA_CREACION)
        VALUES(D_CANCELACIONES.ID_ANOTACION,
               D_CANCELACIONES.ID_ANOTACION1, P_CIRCULO_DESTINO||'-'||N_LAST_ID_FOLIO,
               D_CANCELACIONES.CNCL_DESCRIPCION,
               D_CANCELACIONES.CNCL_FECHA_CREACION);

        END LOOP;
        CLOSE C_CANCELACIONES;

        OPEN
        C_SALVEDAD_AN(D_ANOTACIONES.ID_MATRICULA,D_ANOTACIONES.ID_ANOTACION);
        LOOP FETCH C_SALVEDAD_AN INTO D_SALVEDAD_AN;
            EXIT WHEN C_SALVEDAD_AN%NOTFOUND;

            INSERT INTO SIR.SIR_NE_SALVEDAD_AN(ID_ANOTACION,
                                               ID_MATRICULA,
                                               ID_SALVEDAD_AN, SLAN_DESCRIPCION,
                                               SLAN_FECHA_CREACION, ID_USUARIO, ID_USUARIO_TMP,
                                               SLAN_RADICACION, SLAN_USUARIO_ANT)
                VALUES(D_SALVEDAD_AN.ID_ANOTACION, P_CIRCULO_DESTINO||'-
'||N_LAST_ID_FOLIO,
                       D_SALVEDAD_AN.ID_SALVEDAD_AN,
                       D_SALVEDAD_AN.SLAN_DESCRIPCION,
                       D_SALVEDAD_AN.SLAN_FECHA_CREACION,
                       D_SALVEDAD_AN.ID_USUARIO,
                       D_SALVEDAD_AN.ID_USUARIO_TMP,
                       D_SALVEDAD_AN.SLAN_RADICACION, D_SALVEDAD_AN.SLAN_USUARIO_ANT);
            END LOOP;
            CLOSE C_SALVEDAD_AN;
        END LOOP;
        CLOSE C_ANOTACIONES;
        END IF;
        END IF;
        END IF;
        END IF;
        END IF;
        END IF;
        COMMIT;
EXCEPTION WHEN EXCEP_TURNOS_ACTIVOS THEN
    ROLLBACK TO TRANSACTION_TRASLADO;
    ERR_NUM:=SQLCODE;
    ERR_MSJ:=SQLERRM;
    RAISE_APPLICATION_ERROR(-20002,'NO SE PUEDE REALIZAR EL TRASLADO EL FOLIO
TIENE TURNOS EN TRÁMITE: '||P_ID_FOLIO||' TURNO: '||N_TURNOS_ACTIVOS);
WHEN EXCEP_FOLIO_BLOQUEADO THEN
    ROLLBACK TO TRANSACTION_TRASLADO;
    ERR_NUM:=SQLCODE;
    ERR_MSJ:=SQLERRM;
    RAISE_APPLICATION_ERROR(-20003,'FOLIO BLOQUEADO ');
WHEN EXCEP_FOLIO_TMP THEN
    ROLLBACK TO TRANSACTION_TRASLADO;
    ERR_NUM:=SQLCODE;
    ERR_MSJ:=SQLERRM;
    RAISE_APPLICATION_ERROR(-20004,'FOLIO EN ESTADO TEMPORAL ');
WHEN EXCEP_ZONA_REGISTRAL THEN
    ROLLBACK TO TRANSACTION_TRASLADO;
    ERR_NUM:=SQLCODE;
    ERR_MSJ:=SQLERRM;
    RAISE_APPLICATION_ERROR(-20005,'NO SE PUEDE CREAR LA ZONA REGISTRAL ');
WHEN EXCEP_ANOTA_CIUDADANO THEN
    ROLLBACK TO TRANSACTION_TRASLADO;
    ERR_NUM:=SQLCODE;

```

```

ERR_MSJ:=SQLERRM;
RAISE_APPLICATION_ERROR(-20006,'NO SE PUEDE CREAR EL CIUDADANO ');
WHEN EXCEP_FOLIO_NOEXISTE THEN
ROLLBACK TO TRANSACTION_TRASLADO;
ERR_NUM:=SQLCODE;
ERR_MSJ:=SQLERRM;
RAISE_APPLICATION_ERROR(-20007,'FOLIO NO EXISTE ');
WHEN EXCEP_FOLIO_TRASLADADO THEN
ROLLBACK TO TRANSACTION_TRASLADO;
ERR_NUM:=SQLCODE;
ERR_MSJ:=SQLERRM;
RAISE_APPLICATION_ERROR(-20008,'FOLIO EN ESTADO TRASLADADO ');
WHEN OTHERS THEN
ROLLBACK TO TRANSACTION_TRASLADO;
ERR_NUM:=SQLCODE;
ERR_MSJ:=SQLERRM;
RAISE;
END SP_TRASLADO_FOLIOS;
END PCK_TRASLADO_FOLIO;

```

## DESPLIEGUE – 2.6.5

30/12/2010

### 003 Mensajes de error en el sistema (Informativo)

#### **REQUERIMIENTO**

Mensajes de error en el sistema: claridad de los mensajes

SIR-312.Existen en ocasiones mensajes de error que no son claros para el usuario. Se solicita que se cambien estos mensajes por otros más comprensibles para el usuario.

#### SEGUIMIENTO ADA:

Especificar los siguiente: Proceso: Fase: Mensaje actual: Mensaje deseado:

#### OBSERVACIONES SNR:

Este requerimiento fue escalado a desarrollo extractándolo de las actas de revisión de SIR en producción, las cuales fueron realizadas por personal de la Delegada para el registro y no especificaron con turnos proceso y fase, por lo tanto no es posible suministrar la información como lo solicito ADA. Se sugiere validar si hay casos escalados en GLPI sobre este tema y si no hay ninguno que aclare el requerimiento se debe CERRAR.

Crear un informe detallado donde se presente un informe de los actuales mensajes que no son lo suficientemente explícitos para el usuario final en una tabla con los siguientes datos: Proceso y Fase (donde se genera el mensaje), Mensaje No Explícito (es decir, mensaje que actualmente presenta el sistema), Mensaje Explícito (es decir, mensaje que se desea presente el sistema).

El 19 de agosto a través de correo electrónico enviado por los funcionarios YAMILE MORENO MELO y JOSE MIGUEL GALVIS LEAL; se respondió lo siguiente a la solicitud de información hecha por parte de desarrollo.

Se debe validar si hay casos escalados en GLPI, desde la fecha de solicitud de este requerimiento, con base en ellos realizar los ajustes necesarios.

### **DESARROLLO**

Con base a la respuesta de los funcionarios de la SNR con respecto a este requerimiento, se procedió averiguar con los agentes de mesa de ayuda y su coordinador, si desde el tiempo que se solicito este requerimiento, no se ha reportado o detectado algún incidente que haga referencia a este caso en GLPI.

Dado la anterior información se procede a cerrar el caso.

### **013 Certificados Masivos Exentos Impresión de Recibos**

#### **REQUERIMIENTO**

##### **Certificados Masivos Exentos: Impresión de Recibo.**

SIR-251

Como quiera que no se exige recibo de caja individual para cada certificado exento, expedido a partir de una solicitud masiva, sino un recibo para la solicitud masiva, es necesario, como mecanismo de control, que el sistema exija como campo obligatorio en la Radicación de la solicitud masiva, el tipo, número, fecha y oficina de origen del documento que sirve de soporte para la expedición de los certificados tipo exento. Esta información también debe salir impresa en el recibo de caja y se debe visualizar en el historial del turno masivo.

Igualmente, es necesario que se permita asignar e imprimir un secuencial de recibo para los turnos de certificado masivo exento, que no genere conflicto con los secuenciales de recibos de otros procesos, para aquellos casos en que el mismo funcionario ejecuta varios roles dentro de la ORIP.

#### **SEGUIMIENTO ADA:**

Se requiere dar mayor claridad al proceso, especificando este de manera similar al proceso de devoluciones.

#### **OBSERVACIONES SNR:**

Se considera que la solicitud es clara.

### **DESARROLLO**

Se modificaron los siguientes archivos:

Se modifco el archivo xml de la persistencia **gov.sir.impl**

#### **ARCHIVO JSP:**

#### **WEB-INF.jspSeguro.certificados\_masivos. SOLICITUD\_CERTIFICADOS\_MASIVOS.jsp**

##### **Código Insertado Línea 10:**

```
<%--  
 * @author : Julio Alcázar Rivas  
 * @change : Nuevo import de oficinas  
 * Caso Mantis : 000941  
--%>  
%@page import="gov.sir.core.web.acciones.comun.AWOficinas"%>
```

##### **Código Insertado Línea 84:**

```
<%--  
 * @author : Julio Alcázar Rivas  
 * @change : Funciones javascript para cargar las oficinas por ubicación geográfica dependientes al
```

```

documento de la solicitud.
 * Caso Mantis : 000941
--%>
<script type="text/javascript">
var constancia;
function cambiarAccion(text) {
    document.BUSCAR.<%= WebKeys.ACCION %>.value = text;
}

function oficinas(nombre,valor,dimensiones){
//document.BUSCAR.ACCION.value='<%=CSolicitudRegistro.PRESERVAR_INFO%>';
var idDepto =
document.getElementById('<%=CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_DEPTO%>').value;
var idMunic =
document.getElementById('<%=CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_MUNIC%>').value;
var idVereda =
document.getElementById('<%=CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_VEREDA%>').value;
document.getElementById('tipo_oficina').value=valor+"_ID_TIPO";
document.getElementById('tipo_nom_oficina').value=valor+"_TIPO";
document.getElementById('numero_oficina').value=valor+"_NUM";
document.getElementById('id_oficina').value=valor+"_ID_OFICINA";

popup=window.open(nombre+'?<%=AWOficinas.ID_DEPTO%>='+idDepto+'&<%=AWOficinas.ID_MUNIC%>='+idMunic+'&<%=AWOficinas.ID_VEREDA%>='+idVereda,valor,dimensiones);
popup.focus();
}

function locacion(nombre,valor,dimensiones){
document.getElementById('id_depto').value=valor+"_ID_DEPTO";
document.getElementById('nom_Depto').value=valor+"_NOM_DEPTO";
document.getElementById('id_munic').value=valor+"_ID_MUNIC";
document.getElementById('nom_munic').value=valor+"_NOM_MUNIC";
document.getElementById('id_vereda').value=valor+"_ID_VEREDA";
document.getElementById('nom_vereda').value=valor+"_NOM_VEREDA";
popup=window.open(nombre,valor,dimensiones);
popup.focus();
}

function locacion1(nombre,valor,dimensiones){
document.getElementById('id_depto').value=valor+"_ID_DEPTO";
document.getElementById('nom_Depto').value=valor+"_NOM_DEPTO";
document.getElementById('id_munic').value=valor+"_ID_MUNIC";
document.getElementById('nom_munic').value=valor+"_NOM_MUNIC";
document.getElementById('id_vereda').value=valor+"_ID_VEREDA";
document.getElementById('nom_vereda').value=valor+"_NOM_VEREDA";
popup=window.open(nombre,valor,dimensiones);
popup.focus();
}
</script>

```

#### Código Insertado Línea 336:

```

<%--
 * @author : Julio Alcázar Rivas
 * @change : Se agrega el formulario para el ingreso del documento de la solicitud
 * Caso Mantis : 000941
--%>

```

```

<br>
<hr class="linehorizontal">
<form name="SOLICITUD" method="post"
action="turnoLiquidacionCertificadosMasivos.do">
    <input type="hidden" name="ACCION" value="LIQUIDAR">
    <input name="Dept" type="hidden" id="id_dept" value="">
    <input name="Dept" type="hidden" id="nom_Dept" value="">
    <input name="Mpio" type="hidden" id="id_munic" value="">
    <input name="Mpio" type="hidden" id="nom_munic" value="">
    <input name="Ver" type="hidden" id="id_vereda" value="">
    <input name="Ver" type="hidden" id="nom_vereda" value="">
    <table width="100%" border="0" cellpadding="0" cellspacing="0">
        <tr>
            <td width="12"></td>
            <td class="bgnsub">Encabezado del Documento</td>
            <td width="16" class="bgnsub"></td>
            <td width="15"></td>
        </tr>
        </table>
        <table width="100%" class="camposform">
            <tr>
                <td width="20"></td>
                <td>Datos B&aacute;sicos</td>
            </tr>
            <tr>
                <td>&nbsp;</td>
                <td>
                    <table width="100%" class="camposform">
                        <tr>
                            <td>Tipo</td>
                            <td>
                                <% try {
                                    textHelper.setNombre(CSolicitudRegistro.ID_TIPO_ENCABEZADO);
                                    textHelper.setCssClase("camposformtext");
                                    textHelper.setFuncion("onchange=\"javascript:cambiarValorTipoDocumento(this.value);");
                                    cambiarAccionForm('LIQUIDAR_REGISTRO');\"");
                                    textHelper.setId(CSolicitudRegistro.ID_TIPO_ENCABEZADO);
                                    textHelper.setEditable(true);
                                    textHelper.render(request,out);
                                    textHelper.setFuncion("");
                                } catch(HelperException re){
                                    out.println("ERROR " + re.getMessage());
                                }
                                %>
                            </td>
                            <td>
                                <% try {
                                    List tiposDoc = (List)
                                    session.getServletContext().getAttribute(WebKeys.LISTA_TIPOS_DOCUMENTO);
                                    ListaElementoHelper tipoEncabezado= new ListaElementoHelper();
                                }
                                %>
                            </td>
                        </tr>
                    </table>
                </td>
            </tr>
        </table>
    </form>

```

```

        tipoEncabezado.setCampoOrdenamiento(ListaElementoHelper.ORDENAMIENTO_POR_ID);

        tipoEncabezado.setFuncion("onchange=getElementById('ID_TIPO_ENCABEZADO').value=
his.value;");
        tipoEncabezado.setOrdenar(false);
        tipoEncabezado.setTipos(tiposDoc);

        tipoEncabezado.setNombre(CSolicitudRegistro.TIPO_ENCABEZADO);
                tipoEncabezado.setCssClase("camposformtext");
                tipoEncabezado.setId(CSolicitudRegistro.TIPO_ENCABEZADO);
        tipoEncabezado.render(request,out);
    }catch(HelperException re){
        out.println("ERROR " + re.getMessage());
    }
    %>
</td>
<script>

        document.getElementById('ID_TIPO_ENCABEZADO').value=document.getElementById('TI
PO_ENCABEZADO').value;

        if(document.getElementById('ID_TIPO_ENCABEZADO').value=='SIN_SELECCIONAR'){
            document.getElementById('ID_TIPO_ENCABEZADO').value="";
        }
</script>
<td>N&uacute;mero</td>
<td>
<% try {
        textHelper.setNombre(CSolicitudRegistro.ID_ENCABEZADO);
textHelper.setCssClase("camposformtext");
textHelper.setId(CSolicitudRegistro.ID_ENCABEZADO);
        textHelper.render(request,out);
    }catch(HelperException re){
        out.println("ERROR " + re.getMessage());
    }
    %>
</td>
<td>Fecha</td>
<td>
<table border="0" cellpadding="0" cellspacing="0">
<tr>
<td>
<% try {
        textHelper.setNombre(CSolicitudRegistro.CALENDAR);
textHelper.setCssClase("camposformtext");
textHelper.setId(CSolicitudRegistro.CALENDAR);
        textHelper.render(request,out);
    }catch(HelperException re){
        out.println("ERROR " + re.getMessage());
    }
    %>
</td>
<td>
<a href="javascript:NewCal('calendar','ddmmmyyyy',true,24)">')">
</a>
</td>

```

```

        </tr>
        </table>
        </td>
    </tr>
    </table>
</td>
</tr>
<tr>
    <td>
</td>
    <td>Oficina de Procedencia</td>
</tr>
<!-- EL HELPER DE OFICINA EMPIEZA ACA --&gt;
&lt;jsp:include page="../registro/HELPER_OFICINAS.jsp" flush="true" /&gt;
<!-- EL HELPER DE OFICINA TERMINA ACA --&gt;
&lt;/table&gt;

<table width="100%" border="0" cellpadding="0" cellspacing="0"&gt;
</pre>

```

### WEB-INF.jspSeguro.certificados\_masivos.INFO\_SOLICITUD\_TURNO\_CERTIFICADO.jsp

#### Código Insertado Línea 1:

```

<%--
 * @author : Julio Alcázar Rivas
 * @change : Nuevo Import Calendar
 * Caso Mantis : 000941
--%>
%@page import="java.util.Calendar"%>

```

#### Código Insertado Línea 55:

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Se agrega el documento a la información de la solicitud
 * Caso Mantis : 000941
*/
Documento documento = solicitud.getDocumento();
String fechaHoy = null;
if(documento != null){
    Calendar hoy = Calendar.getInstance();
    hoy.setTime(documento.getFecha());
    fechaHoy=hoy.get(Calendar.DAY_OF_MONTH)+"/"+(hoy.get(Calendar.MONTH)+1)+"/"+hoy.get(Calendar.YEAR);
}

```

#### Código Insertado Línea 252:

```

<br>
<%--
 * @author : Julio Alcázar Rivas
 * @change : Se muestra el documento en la información de la solicitud
 * Caso Mantis : 000941
--%>





```

```

        </td>
        <td width="16" class="bgnsub">
        </td>
        <td width="16" class="bgnsub">
        </td>
        <td width="15">
        </td>
    </tr>
    </table>
<br>
    <table width="100%" border="0" cellpadding="0" cellspacing="2"
        class="camposform">
        <tr>
            <td width="20"></td>
            <td align="left">Datos B&acute;sicos</td>
        </tr>
        <tr>
            <td width="10%" align="right">Tipo</td>
            <td width="23%" class="campositem"><%=

documento.getTipoDocumento().getNombre() != null ? documento.getTipoDocumento().getNombre() :
"nbsp;"%></td>
            <td width="10%" align="right">Numero</td>
            <td width="23%" class="campositem"><%= documento.getNumero() != null ?
documento.getNumero() : "nbsp;"%></td>
            <td width="10%" align="right">Fecha</td>
            <td width="23%" class="campositem"><%= fechaHoy != null ? fechaHoy :
"nbsp;"%></td>
        </tr>
    </table>
    <table width="100%" border="0" cellpadding="0" cellspacing="2"
        class="camposform">
        <tr>
            <td width="20"></td>
            <td align="left">Oficina de Procedencia</td>
        </tr>
        <%
        if (documento.getOficinaInternacional() == null){
            %>
            <tr>
                <td width="10%" align="right">Departamento</td>
                <td width="23%" class="campositem"><%=

documento.getOficinaOrigen().getVereda().getMunicipio().getDepartamento().getNombre() != null ?
documento.getOficinaOrigen().getVereda().getMunicipio().getDepartamento().getNombre() :
"nbsp;"%></td>
                <td width="10%" align="right">Municipio</td>
                <td width="23%" class="campositem"><%=
```

```

documento.getOficinaOrigen().getVereda().getMunicipio().getNombre() != null ?
    documento.getOficinaOrigen().getVereda().getMunicipio().getNombre() :
"&nbsp;"%></td>
</tr>
<tr>
    <td width="10%" align="right">Tipo de Oficina Origen</td>
    <td width="23%" class="campositem"><%=
documento.getOficinaOrigen().getTipoOficina().getNombre() != null ?
    documento.getOficinaOrigen().getTipoOficina().getNombre() : "&nbsp;"%></td>
    <td width="10%" align="right">Nombre</td>
    <td width="23%" class="campositem"><%=
documento.getOficinaOrigen().getNombre() != null ?
    documento.getOficinaOrigen().getNombre() : "&nbsp;"%></td>
</tr>

<% } else{ %>
<tr>
    <td width="10%" align="right">Oficina ubicaci&on Internacional</td>
    <td width="23%" class="campositem"><%= documento.getOficinaInternacional() != null ?
        documento.getOficinaInternacional() : "&nbsp;"%></td>
    </tr>
<% }%>
</table>

<% }%>
</td>

```

#### WEB-INF.jspSeguro.administracion.TURNO\_CERTIFICADOS\_DETALLES\_VISTA.jsp

##### Código Insertado Línea 252:

```

<%--
 * @author : Julio Alcázar Rivas
 * @change : Nuevos Import gov.sir.core.util.DateFormatUtil
 *           gov.sir.core.negocio.modelo.constantes.CTipoTarifa
 * Caso Mantis : 000941
--%>
<%@page import="gov.sir.core.util.DateFormatUtil" %>
%@page import="gov.sir.core.negocio.modelo.constantes.CTipoTarifa" %

```

##### Código Insertado Línea 341:

```

/**
 * @author : Julio Alcázar Rivas
 * @change : Se agrega la información del documento de SolicitudCertificadoMasivo a los
detalles del turno
 * Caso Mantis : 000941
*/
String tipoDocumento="&nbsp;";
String numDocumento="&nbsp;";
String fechaDocumento="&nbsp;";
String comentarioDocumento ="&nbsp;";
String oficinaInternacional="&nbsp;";
String departamento="&nbsp;";
String municipio="&nbsp;";
String vereda="&nbsp;";
String comentario="&nbsp;";
String tipoOficina="&nbsp;";
String nombreOficina="&nbsp;";
String numOficina="&nbsp;";
boolean isInternacional=false;
SolicitudCertificadoMasivo solCertMasi = null;
if(solicitud instanceof SolicitudCertificadoMasivo){

```

```

solCertMasi = (SolicitudCertificadoMasivo)turno.getSolicitud();
//TIPO DOCUMENTO

if(solCertMasi != null &&
    solCertMasi.getDocumento() != null &&
    solCertMasi.getDocumento().getTipoDocumento() != null &&
    solCertMasi.getDocumento().getTipoDocumento().getNombre()!=null){
    tipoDocumento=solCertMasi.getDocumento().getTipoDocumento().getNombre();
}
//NUM DOCUMENTO

if(solCertMasi != null && solCertMasi.getDocumento() != null &&
    solCertMasi.getDocumento().getNumero()!=null){
    numDocumento=solCertMasi.getDocumento().getNumero();
}
//FECHA DOCUMENTO

if(solCertMasi != null && solCertMasi.getDocumento() != null &&
    solCertMasi.getDocumento().getFecha()!=null){

fechaDocumento = DateFormatUtil.format(solCertMasi.getDocumento().getFecha());
}

if(solCertMasi != null && solCertMasi.getDocumento() != null &&
solCertMasi.getDocument().getComentario() != null){
    comentarioDocumento = solCertMasi.getDocument().getComentario();
}
//OFICINA

//OFICINA INTERNACIONAL

if(solCertMasi != null && solCertMasi.getDocumento() != null &&
    solCertMasi.getDocumento().getOficinaInternacional()!=null &&
    !solCertMasi.getDocumento().getOficinaInternacional().equals("") ){
    isInternacional=true;
    oficinaInternacional=solCertMasi.getDocumento().getOficinaInternacional();
}
//OFICINA NACIONAL
//DEPARTAMENTO, MUNICIPIO, VEREDA

if(!isInternacional){
    if(solCertMasi != null && solCertMasi.getDocumento() != null &&
        solCertMasi.getDocumento().getOficinaOrigen() != null &&
        solCertMasi.getDocumento().getOficinaOrigen().getVereda()!=null){
            gov.sir.core.negocio.modelo.Vereda
auxVereda=solCertMasi.getDocumento().getOficinaOrigen().getVereda();
            if(auxVereda.getMunicipio() != null &&
                auxVereda.getMunicipio().getDepartamento() != null &&
auxVereda.getMunicipio().getDepartamento().getNombre() != null){

departamento=auxVereda.getMunicipio().getDepartamento().getNombre();
            }
            if(auxVereda.getMunicipio() != null &&
                auxVereda.getMunicipio().getNombre() != null){
                    municipio=auxVereda.getMunicipio().getNombre();
            }
            if(auxVereda.getNombre() != null){
                vereda=auxVereda.getNombre();
            }
        }
    }
}

```

```

        }

    boolean isMigrado = false;
    if (turno.getDescripcion() != null && turno.getDescripcion().equals(WebKeys.MIGRACION))
    {
        isMigrado = true;
        if (solCertMasi.getDocumento().getComentario() != null) {
            comentario = solCertMasi.getDocumento().getComentario();
        }
    }

//TIPO OFICINA

if(!isInternacional){
    if(solCertMasi != null && solCertMasi.getDocumento() != null &&
       solCertMasi.getDocumento().getOficinaOrigen() != null &&
       solCertMasi.getDocumento().getOficinaOrigen().getTipoOficina() != null &&

       solCertMasi.getDocumento().getOficinaOrigen().getTipoOficina().getNombre()!=null) {
        tipoOficina=solCertMasi.getDocumento().getOficinaOrigen().getTipoOficina().getNombre();
        nombreOficina=solCertMasi.getDocumento().getOficinaOrigen().getNombre();
    }
}

//NUM OFICINA

if(!isInternacional){
    if((solCertMasi != null &&      solCertMasi.getDocumento() != null &&
       solCertMasi.getDocumento().getOficinaOrigen() != null &&
       solCertMasi.getDocumento().getOficinaOrigen().getNumero()!=null)) {

        numOficina=solCertMasi.getDocumento().getOficinaOrigen().getNumero();
        nombreOficina=solCertMasi.getDocumento().getOficinaOrigen().getNombre();
    }
}
}

```

#### Código Insertado Línea 1537:

```

<%--
 * @author : Julio Alcázar Rivas
 * @change : Se imprimen los datos del Documento de la SolicitudCertificadoMasivo Exento
 * Caso Mantis : 000941
--%>
<%
if (solCertMasi != null && solCertMasi.getDocumento() != null &&
((LiquidacionTurnoCertificadoMasivo)solCertMasi.getLiquidaciones().get(liquidaciones.size() -
1)).getTipoTarifa().equals(CTipoTarifa.EXENTO )){

    %>
    <table width="100%" border="0" cellpadding="0" cellspacing="0">
    <tr>

        <td class="bgnsup">Encabezado del Documento </td>
        <td width="16" class="bgnsup"></td>
        <td width="15"></td>
        </tr>
    </table>
    <table width="100%" class="camposform">

```

```

<tr>

    <td>Datos B&aacute;sicos </td>
</tr>
<tr>

    <td><table width="100%" class="camposform">
        <tr>
            <td>Tipo</td>
            <td class="campositem"><%=tipoDocumento!=null?tipoDocumento:"&nbsp;"%></td>
            <td width="15%">&nbsp;</td>

            <td>N&uacute;mero</td>
            <td class="campositem"><%=numDocumento!=null?numDocumento:"&nbsp;"%></td>
            <td width="15%">&nbsp;</td>

            <td>Fecha</td>
            <td
                class="campositem"><%=fechaDocumento!=null?fechaDocumento:"&nbsp;"%></td>
            <td width="15%">&nbsp;</td>

            <td>Comentario</td>
            <td
                class="campositem"><%=comentarioDocumento!=null?comentarioDocumento:"&nbsp;"%></td>
        </tr>
    </table></td>
</tr>
<tr>

    <td>Oficina de Procedencia </td>
</tr>

<tr>

    <td>
        <%if(!isInternacional){%>
            <table width="100%" class="camposform">
                <tr>
                    <td>Departamento</td>
                    <td class="campositem">
                        <%=departamento!=null?departamento:"&nbsp;"%>
                    </td>
                </tr>
                <tr>
                    <td>Municipio</td>
                    <td class="campositem">
                        <%=municipio!=null?municipio:"&nbsp;"%>
                    </td>
                </tr>
            </table>
        <%else{%
            <table width="100%" class="camposform">
                <tr>
                    <td>Tipo</td>
                    <td class="campositem">
                        <%=tipoOficina!=null?tipoOficina:"&nbsp;"%>
                    </td>
                    <td>N&uacute;mero</td>
                    <td class="campositem">

```

```

<%=numOficina!=null?numOficina:"&nbsp;"%>

        </td>
    </tr>
    <tr>
        <td>Nombre</td>
        <td class="campositem">
<%=nombreOficina!=null?nombreOficina:"&nbsp;"%></td>
        </tr>
    </table>
<% }else{ %>
                <table width="100%" class="camposform">
                    <tr>
                        <td>Oficina internacional</td>
                        <td class="campositem">
                            <%=oficinaInternacional%>
                        </td>
                    </tr>
                </table>
                <% }%>
            </td>
        </tr>
    </table>
    <br>
    <br>
    <br>
    <%
    }
    %>

```

**CLASE:**

**gov.sir.core.web.acciones.certificadosmasivos.AWLiquidacionCertificadosMasivos.java**

**Método:**

**private EvnCertificadoMasivo liquidar(HttpServletRequest request) throws  
AccionWebException{**

**Código Insertado Línea 627:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Se agrega el documento a la información de la solicitud
 * Caso Mantis : 000941
 */boolean isInternacional = false;

```

```

String TOficina= request.getParameter(WebKeys.TIPO_OFICINA_I_N);
if(TOficina.equalsIgnoreCase(WebKeys.TIPO_OFICINA_INTERNAZIONAL)){
    isInternacional=true;
}

```

```

String tipo_encabezado = request.getParameter(CSolicitudRegistro.TIPO_ENCABEZADO);
String id_encabezado = request.getParameter(CSolicitudRegistro.ID_ENCABEZADO);
String fecha = request.getParameter(CSolicitudRegistro.CALENDAR);

```

```

session.setAttribute(CSolicitudRegistro.TIPO_ENCABEZADO, tipo_encabezado);
session.setAttribute(CSolicitudRegistro.ID_ENCABEZADO, id_encabezado);
session.setAttribute(CSolicitudRegistro.CALENDAR, fecha);

```

```

String nomOficinaInternacional="";
String nomDepto="";
String idDepto="";
String nomMunic="";

```

```

String idMunic="";
String nomVereda="";
String idVereda="";
String tipo_oficina="";
String numero_oficina="";
String id_oficina="";
if(isInternacional){
    nomOficinaInternacional = request.getParameter(WebKeys.TIPO_OFICINA_INTERNACIONAL);
    session.setAttribute(WebKeys.TIPO_OFICINA_INTERNACIONAL,
    nomOficinaInternacional);
} else {
    nomDept = request.getParameter(CFolio.ANOTACION_OFICINA_DOCUMENTO_NOM_DEPTO);
    idDept = request.getParameter(CFolio.ANOTACION_OFICINA_DOCUMENTO_ID_DEPTO);
    session.setAttribute(CFolio.ANOTACION_OFICINA_DOCUMENTO_NOM_DEPTO, nomDept);

    session.setAttribute(CFolio.ANOTACION_OFICINA_DOCUMENTO_ID_DEPTO, idDept);

    nomMunic = request.getParameter(CFolio.ANOTACION_OFICINA_DOCUMENTO_NOM_MUNIC);
    idMunic = request.getParameter(CFolio.ANOTACION_OFICINA_DOCUMENTO_ID_MUNIC);

    session.setAttribute(CFolio.ANOTACION_OFICINA_DOCUMENTO_NOM_MUNIC,
    nomMunic);

    session.setAttribute(CFolio.ANOTACION_OFICINA_DOCUMENTO_ID_MUNIC, idMunic);

    nomVereda =
    request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_NOM_VEREDA);
    idVereda =
    request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_VEREDA);

    session.setAttribute(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_NOM_VEREDA,
    nomVereda);

    session.setAttribute(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_VEREDA,
    idVereda);

    if( idVereda == null || nomDept.equals("") || nomMunic.equals("") ) {
        idVereda = ValidacionDatosOficinaOrigen.obtenerVereda((List)
        request.getSession().getServletContext().getAttribute(WebKeys.LISTA_DEPARTAMENTOS), idDept,
        idMunic);
    }

    tipo_oficina =
    request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_TIPO);
    numero_oficina =
    request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_NUM);
    id_oficina =
        request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_OFICINA);

    session.setAttribute(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_TIPO,
    tipo_oficina);

    session.setAttribute(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_NUM,
    numero_oficina);

    session.setAttribute(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_OFICINA,
    id_oficina);
}

```

```

        }

        if((tipo_encabezado == null) || tipo_encabezado.equals("SIN_SELECCIONAR")) {
            exception.addError("El tipo de encabezado es inválido");
        }

        Calendar fechaDocumento = darFecha(fecha);

        if(fechaDocumento == null) {
            exception.addError("La fecha del documento es invalida");
        }

        if((fecha == null) || fecha.trim().equals("")) {
            exception.addError("La fecha del encabezado es inválida");
        }

        if((nomOficinaInternacional!=null && nomOficinaInternacional.length()>0 ) ||
        (idVereda!=null && idVereda.length()>0)|||
        (idDepto!=null && idDepto.length()>0)|||
        (idMunic!=null && idMunic.length()>0)|||
        (id_oficina!=null&&id_oficina.length()>0))|


        if(isInternacional){
        if(nomOficinaInternacional == null) || nomOficinaInternacional.trim().equals("")) {
            exception.addError("El nombre de la oficina internacional es inválido");
        }
        }else{

        if((idVereda == null) || idVereda.trim().equals("")) {
            .addError("La vereda o ciudad es inválida");
        }

        if((idDepto == null) || idDepto.trim().equals("")) {
            exception.addError("El departamento es inválido");
        }

        if((idMunic == null) || idMunic.trim().equals("")) {
            exception.addError("El municipio es inválido");
        }

        if((id_oficina == null) || id_oficina.trim().equals("")) {
            exception.addError("El número de oficina es inválido");
        }
        }
        }

        if(exception.getErrores().size() > 0) {
            exception;
        }

        String valorDepartamento =
            request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_DEPTO);
        String valorMunicipio =
            request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_MUNICIPIO);
    }
}

```

```

String valorVereda =
    request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_VEREDA);
String idOficina =
    request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_OFICINA);
String nomOficina =
    request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_TIPO);
String numOficina =
    request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_NUM);

if(!isInternacional){

    if(nomOficina != null) {
        oficinaOrigen.setNombre(nomOficina);
    }

    if(numOficina != null) {
        oficinaOrigen.setNumero(numOficina);
    }

    oficinaOrigen.setIdOficinaOrigen(idOficina);

    Vereda vereda = new Vereda();
    Municipio municipio = new Municipio();
    Departamento departamento = new Departamento();

    departamento.setNombre(nomDept);
    municipio.setDepartamento(departamento);
    vereda.setMunicipio(municipio);

    if(nomVereda != null) {
        vereda.setNombre(nomVereda);
    }

    vereda.setIdVereda(valorVereda);
    vereda.setIdDepartamento(valorDepartamento);
    vereda.setIdMunicipio(valorMunicipio);

    oficinaOrigen.setVereda(vereda);
}

Documento documento = new Documento();
documento.setFecha(fechaDocumento.getTime());

if((nomOficinaInternacional!=null && nomOficinaInternacional.length()>0 ) ||
(idVereda!=null && idVereda.length()>0) ||
(idDept!=null && idDept.length()>0) ||
(idMunic!=null && idMunic.length()>0) ||
(id_oficina!=null&&id_oficina.length()>0)) {

    if(!isInternacional){
        documento.setOficinaInternacional(null);
        documento.setOficinaOrigen(oficinaOrigen);
    }else{
        documento.setOficinaOrigen(null);
        documento.setOficinaInternacional(nomOficinaInternacional);
    }
}

```

```

        }

        TipoDocumento tipoDoc = new TipoDocumento();
        tipoDoc.setIdTipoDocumento(tipo_encabezado);

        List tiposDocs = (List)
        request.getSession().getServletContext().getAttribute(WebKeys.LISTA_TIPOS_DOCUMENTO);
        Iterator itTiposDocs = tiposDocs.iterator();

        while (itTiposDocs.hasNext()) {
            ElementoLista elemento = (ElementoLista) itTiposDocs.next();

            if (elemento.getId().equals(tipo_encabezado)) {
                tipoDoc.setNombre(elemento.getValor());
            }
        }

        documento.setTipoDocumento(tipoDoc);
        documento.setNumero(id_encabezado);
    }
}

```

**Código Insertado Línea 879:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : SET del documento en la solicitud
 * Caso Mantis : 000941
 */
solicitud.setDocumento(documento);

```

**Código Insertado Línea 1021:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Se agrega el documento a la session
 * Caso Mantis : 000941
 */
if(request.getParameter(CSolicitudRegistro.TIPO_ENCABEZADO) != null){
    String tipo_encabezado = request.getParameter(CSolicitudRegistro.TIPO_ENCABEZADO);
    session.setAttribute(CSolicitudRegistro.TIPO_ENCABEZADO, tipo_encabezado);
}

if(request.getParameter(CSolicitudRegistro.ID_ENCABEZADO) != null ){
    String id_encabezado = request.getParameter(CSolicitudRegistro.ID_ENCABEZADO);
    session.setAttribute(CSolicitudRegistro.ID_ENCABEZADO, id_encabezado);
}

if(request.getParameter(CSolicitudRegistro.CALENDAR) != null ){
    String fecha = request.getParameter(CSolicitudRegistro.CALENDAR);
    session.setAttribute(CSolicitudRegistro.CALENDAR, fecha);
}

if(request.getParameter(WebKeys.TIPO_OFICINA_INTERNACIONAL) != null ){
    String nomOficinaInternacional =
    request.getParameter(WebKeys.TIPO_OFICINA_INTERNACIONAL);
    session.setAttribute(WebKeys.TIPO_OFICINA_INTERNACIONAL, nomOficinaInternacional);
}

if(request.getParameter(CFolio.ANOTACION_OFICINA_DOCUMENTO_NOM_DEPTO) != null ){
    String nomDept = 
```

```

request.getParameter(CFolio.ANOTACION_OFICINA_DOCUMENTO_NOM_DEPTO);
session.setAttribute(CFolio.ANOTACION_OFICINA_DOCUMENTO_NOM_DEPTO, nomDepto);
}

if(request.getParameter(CFolio.ANOTACION_OFICINA_DOCUMENTO_ID_DEPTO) != null ){
String idDepto = request.getParameter(CFolio.ANOTACION_OFICINA_DOCUMENTO_ID_DEPTO);
session.setAttribute(CFolio.ANOTACION_OFICINA_DOCUMENTO_ID_DEPTO, idDepto);
}

if(request.getParameter(CFolio.ANOTACION_OFICINA_DOCUMENTO_NOM_MUNIC) != null ){
String nomMunic =
request.getParameter(CFolio.ANOTACION_OFICINA_DOCUMENTO_NOM_MUNIC);
session.setAttribute(CFolio.ANOTACION_OFICINA_DOCUMENTO_NOM_MUNIC, nomMunic);
}

if(request.getParameter(CFolio.ANOTACION_OFICINA_DOCUMENTO_ID_MUNIC) != null ){
String idMunic =
request.getParameter(CFolio.ANOTACION_OFICINA_DOCUMENTO_ID_MUNIC);
session.setAttribute(CFolio.ANOTACION_OFICINA_DOCUMENTO_ID_MUNIC, idMunic);
}

if(request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_NOM_VEREDA)
!= null ){
String nomVereda =
request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_NOM_VEREDA);
session.setAttribute(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_NOM_VEREDA,
nomVereda);
}

if(request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_VEREDA)
!= null ){
String idVereda =
request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_VEREDA);
session.setAttribute(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_VEREDA,
idVereda);
}

if(request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_TIPO) != null ){
String tipo_oficina =
request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_TIPO);
session.setAttribute(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_TIPO, tipo_oficina);
}

if(request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_NUM) != null ){
String numero_oficina =
request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_NUM);
session.setAttribute(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_NUM,
numero_oficina);
}

if(request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_OFICINA)
!= null ){
String id_oficina
=request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_OFICINA);
session.setAttribute(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_ID_OFICINA,
id_oficina);
}

```

**Código Insertado Línea 1093:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : metodo formato fecha
 * Caso Mantis : 000941
 * @param fechaInterfaz
 * @return Calendar
 */
private Calendar darFecha(String fechaInterfaz) {
    java.util.Date date = null;

    try {
        date = DateFormatUtil.parse(fechaInterfaz);
        if(fechaInterfaz.indexOf("-")!=-1){
            return null;
        }
    } catch (ParseException e) {
        return null;
    } catch (Throwable t) {
        return null;
    }

    Calendar calendar = Calendar.getInstance();
    String[] partido = fechaInterfaz.split("/");
    if(partido.length == 3) {
        int dia = Integer.parseInt(partido[0]);
        int mes = Integer.parseInt(partido[1]) - 1;
        int año = Integer.parseInt(partido[2]);
        calendar.set(año, mes, dia);

        if((calendar.get(Calendar.YEAR) == año) && (calendar.get(Calendar.MONTH) == mes) &&
(calendar.get(Calendar.DAY_OF_MONTH) == dia)) {
            return calendar;
        }
    }
    return null;
}

```

**CLASE:**

gov.sir.core.negocio.acciones.certificadosmasivos. ANCertificadoMasivo.java

**Método:**

private EvnCertificadoMasivo liquidar(HttpServletRequest request) throws  
AccionWebException{

**Código Insertado Línea 254:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Determina si la solicitud es de tipo SolicitudCertificadoMasivo
 * Caso Mantis : 000941
 */
SolicitudCertificadoMasivo solicitudcertmasivo = null;
if( liquidacion.getSolicitud() instanceof SolicitudCertificadoMasivo){
    solicitudcertmasivo = (SolicitudCertificadoMasivo) liquidacion.getSolicitud();
}

```

**Código Insertado Línea 354:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Se existe un documento, se adjunta a la solicitud SolicitudCertificadoMasivo
 * Caso Mantis : 000941

```

```
/*
if(solicitudcertmasivo != null && solicitudcertmasivo.getDocumento()!= null){
    scm.setDocumento(solicitudcertmasivo.getDocumento());
}
```

**CLASE:**

**gov.sir.core.negocio.modelo.SolicitudCertificadoMasivo.java**

**Código Insertado Linea 21:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Nuevo atributo de la clase
 * Caso Mantis : 000941
 */
private Documento documento;
```

**Código Insertado Linea 41:**

```
/* @author : Julio Alcázar Rivas
 * @change : Metodo Get -> documento
 * @param documento
 */
public Documento getDocumento() {
    return documento;
}
```

**Código Insertado Linea 49:**

```
/* @author : Julio Alcázar Rivas
 * @change : Metodo Set -> documento
 * @param documento
 */
public void setDocumento(Documento documento) {
    this.documento = documento;
}
```

**CLASE:**

**gov.sir.core.negocio.modelo.jdogenie.SolicitudCertificadoMasivoEnhanced.java**

**Código Insertado Linea 20:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Nuevo atributo de la clase
 * Caso Mantis : 000941
 */
private DocumentoEnhanced documento;
```

**Código Insertado Linea 45:**

```
/* @author : Julio Alcázar Rivas
 * @change : Metodo Get -> documento
 * @param documento
 */
public DocumentoEnhanced getDocumento() {
    return documento;
}
```

**Código Insertado Linea 53:**

```
/* @author : Julio Alcázar Rivas
 * @change : Metodo Set -> documento
 * @param documento
 */
```

```
public void setDocumento(DocumentoEnhanced enhanced) {
    documento = enhanced;
}
```

**CLASE:**

```
gov.sir.core.negocio.modelo.transferutil.SolicitudCertificadoMasivoTransferCopier.java
```

**Método:**

```
public static org.auriga.core.modelo.TransferObject deepCopy(
    gov.sir.core.negocio.modelo.jdogenie.SolicitudCertificadoMasivoEnhanced enhancedObject,
    java.util.HashMap cache)
```

**Código Insertado Línea 39:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : TransferObject al documento
 * Caso Mantis : 000941
 */
gov.sir.core.negocio.modelo.jdogenie.DocumentoEnhanced documentoEnh = null;
try {
    documentoEnh = enhancedObject.getDocumento();
} catch (Throwable th){}
if(documentoEnh != null)
{
    gov.sir.core.negocio.modelo.Documento objTo =
(gov.sir.core.negocio.modelo.Documento)cache.get(documentoEnh);
    if(objTo == null)
    {
        objTo =
(gov.sir.core.negocio.modelo.Documento)DocumentoTransferCopier.deepCopy(documentoEnh, cache);
    }
    transferObject.setDocumento(objTo);
}
```

**Método:**

```
public static gov.sir.core.negocio.modelo.jdogenie Enhanced deepCopy(
    gov.sir.core.negocio.modelo.SolicitudCertificadoMasivo transferObject, java.util.HashMap
    cache){}
```

**Código Insertado Línea 446:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Enhanced al documento
 * Caso Mantis : 000941
 */
gov.sir.core.negocio.modelo.Documento documento = null;
try {
    documento = (gov.sir.core.negocio.modelo.Documento) transferObject.getDocumento();
} catch (Throwable th){}
if(documento != null)
{
    gov.sir.core.negocio.modelo.jdogenie.DocumentoEnhanced objEn =
(gov.sir.core.negocio.modelo.jdogenie.DocumentoEnhanced)cache.get(documento);
    if(objEn == null)
    {
        objEn =
(gov.sir.core.negocio.modelo.jdogenie.DocumentoEnhanced)DocumentoTransferCopier.deepCopy(docu
mento, cache);
    }
    enhancedObject.setDocumento(objEn);
}
```

**CLASE:****gov.sir.hermod.HermodService.java****Código Insertado Línea 7798:**

```
/**
 * @author : Julio Alcázar Rivas
 * @change : Obtiene y avanza la secuencia de los recibos de certificados masivos, de acuerdo
 * a los parametros recibidos.
 * @Caso Mantis : 000941
 * @param circuloId El identificador del <code>Circulo</code> en el que se va a
 * expedir el recibo de certificados masivos.
 * @param year El año en el que se va a expedir el recibo de certificados masivos.
 * @return El secuencial requerido.
 * @throws HermodException
 */
public long getSecuencialMasivosExento (String circuloId, String year)
    throws HermodException
{

    try {
        return factory.getProcesosDAO().getSecuencialMasivosExento(circuloId,year);
    } catch (DAOException e) {
        HermodException he = new
HermodException(HermodException.SECUENCIAL_MASIVOS_NO_OBTENIDO, e);
        ExceptionPrinter ep = new ExceptionPrinter(he.fillInStackTrace());
        Log.getInstance().error(this.getClass(),he.getMessage(), e);
        throw he;
    }
}
```

**CLASE:****gov.sir.hermod.ejb.HermodServiceEJBBean.java****Código Insertado Línea 2856:**

```
/**
 * @author : Julio Alcázar Rivas
 * @change : Metodo para obtener el secuencial masivo
 * @Caso Mantis : 000941
 * @see gov.sir.hermod.interfaz.HermodServiceInterface#getSecuencialMasivosExento(String, String)
 * @ejb:interface-method view-type="both"
 */
public long getSecuencialMasivosExento(String circuloId, String year) throws HermodException {
    return hermod.getSecuencialMasivos(circuloId,year);
}
```

**CLASE:****gov.sir.hermod.interfaz.HermodServiceInterface.java****Código Insertado Línea 3786:**

```
/**
 * @author : Julio Alcázar Rivas
 * @change : Obtiene y avanza la secuencia de los recibos de certificados masivos, de acuerdo
 * a los parametros recibidos.
 * @Caso Mantis : 000941
 * @param circuloId El identificador del <code>Circulo</code> en el que se va a
 * expedir el recibo de certificados masivos.
 * @param year El año en el que se va a expedir el recibo de certificados masivos.
 * @return El secuencial requerido.
 * @throws HermodException
 */
public long getSecuencialMasivosExento (String circuloId, String year)
    throws Throwable;
```

**CLASE:**

**gov.sir.hermode.dao.ProcesosDAO.java**

**Código Insertado Linea 123:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Obtiene y avanza la secuencia de los recibos de certificados masivos, de acuerdo
 * a los parametros recibidos.
 * @Caso Mantis : 000941
 * @param circuloId El identificador del <code>Circulo</code> en el que se va a
 * expedir el recibo de certificados masivos.
 * @param year El año en el que se va a expedir el recibo de certificados masivos.
 * @return El secuencial requerido.
 * @throws HermodException
 */
public long getSecuencialMasivosExento (String circuloId, String year)
throws DAOException;
```

**CLASE:**

**gov.sir.hermode.dao.impl.jdogenie.JDOProcesosDAO.java**

**Código Insertado Linea 858:**

```
/*
 * @author : Julio Alcázar Rivas
 * @change : Obtiene y avanza la secuencia de los recibos de certificados masivos, de acuerdo
 * a los parametros recibidos.
 * @Caso Mantis : 000941
 * @param circuloId El identificador del <code>Circulo</code> en el que se va a
 * expedir el recibo de certificados masivos.
 * @param year El año en el que se va a expedir el recibo de certificados masivos.
 * @return El secuencial requerido.
 * @throws HermodException
 */
public long getSecuencialMasivosExento(String circuloId, String year) throws DAOException {
long rta = -1;
PersistenceManager pm = null;
VersantPersistenceManager pm2 = null;

if(circuloId == null) {
    throw new DAOException("El identificador del círculo es inválido");
}
try {
    pm = AdministradorPM.getPM();
    pm.currentTransaction().begin();
    //Se hace el cambio de tipo de bloqueo pesimista para
    //que se bloquee la tabla la cual nos
    //provee el secuencial
    pm2 = (VersantPersistenceManager) pm;
    pm2.setDatastoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
    pm = pm2;
    CirculoProcesoEnhancedPk sid = new CirculoProcesoEnhancedPk();
    sid.anio = year;
    sid.idCirculo = circuloId;
    Long idProcesoString = new Long (CProceso.PROCESO_CERTIFICADOS_MASIVOS);
    sid.idProceso = idProcesoString.longValue();

    CirculoProcesoEnhanced circuloProceso = this.get CirculoProcesoByID(sid, pm);
    if(circuloProceso == null)
    {
        pm2.setDatastoreTxLocking(VersantPersistenceManager.LOCKING_NONE);
    }
}
```

```

        throw new DAOException ("No se pudo obtener el secuencial para el recibo de certificados masivos");
    }

circuloProceso.setSecuencialConstMasivos(circuloProceso.getSecuencialConstMasivos() + 1);

//Volvemos a setear el tipo de bloqueo pesimista
//para que no nos bloquee los siguientes registros
//consultados

pm2.setDatystoreTxLocking(VersantPersistenceManager.LOCKING_NONE);

rta = circuloProceso.getSecuencialConstMasivos();
}

//No se encontró el círculo Proceso.
catch (JDOObjectNotFoundException e) {
    pm2.setDatystoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
if(pm.currentTransaction().isActive())
{
    pm.currentTransaction().rollback();
}
throw new DAOException("No se pudo obtener el secuencial para el recibo de certificados masivos");
}

//Excepción JDO
catch (JDOException e) {
    pm2.setDatystoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
if(pm.currentTransaction().isActive())
{
    pm.currentTransaction().rollback();
}
throw new DAOException("No se pudo obtener el secuencial para el recibo de certificados masivos",e);
}
catch (DAOException e) {
    pm2.setDatystoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
if(pm.currentTransaction().isActive())
{
    pm.currentTransaction().rollback();
}
throw new DAOException("No se pudo obtener el secuencial para el recibo de certificados masivos",e);
}
catch (Exception e) {
    pm2.setDatystoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
if(pm.currentTransaction().isActive())
{
    pm.currentTransaction().rollback();
}
throw new DAOException("No se pudo obtener el secuencial para el recibo de certificados masivos",e);
}
//Otra Excepción
catch (Throwable t) {
    pm2.setDatystoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
if(pm.currentTransaction().isActive())
{
    pm.currentTransaction().rollback();
}
}

```

```

        throw new DAOException("No se pudo obtener el secuencial para el recibo de certificados masivos");
    }

    //Cerrar la transacción
    finally {
        //pm2.setDatastoreTxLocking(VersantPersistenceManager.LOCKING_ALL);
        if(pm.currentTransaction().isActive()) {
            pm.currentTransaction().commit();
        }
        pm.close();
    }

    return rta;
}

```

**CLASE:**

**gov.sir.hermod.dao.impl.jdogenie.JDOSolicitudesDAO.java**

**Método:**

```

protected SolicitudEnhanced makeTransientSolicitud(SolicitudEnhanced sr, PersistenceManager
pm)
throws DAOException {

```

**Código Insertado Línea 2425:**

```

/**
 * @author : Julio Alcázar Rivas
 * @change : makeTransient al documento de la solicitud
 * Caso Mantis : 000941
 */
if(sc.getDocumento()!= null)
{
    pm.makeTransient (sc.getDocumento().getTipoDocumento());
    if(sc.getDocumento().getOficinaOrigen()!=null)
    {

        pm.makeTransient(sc.getDocumento().getOficinaOrigen().getVereda().getMunicipio().getDepartamento());

    pm.makeTransient(sc.getDocumento().getOficinaOrigen().getVereda().getMunicipio());
    pm.makeTransient(sc.getDocumento().getOficinaOrigen().getVereda());

        pm.makeTransient(sc.getDocumento().getOficinaOrigen().getTipoOficina());
    }

    pm.makeTransient(sc.getDocumento().getOficinaOrigen());
}

pm.makeTransient(sc.getDocumento());
}

```

**Método:**

```

protected SolicitudCertificadoMasivoEnhanced crearSolicitudCertificadoMasivo(
SolicitudCertificadoMasivoEnhanced s, PersistenceManager pm)
throws DAOException {

```

**Código Insertado Línea 5505:**

```

/**
 * @author : Julio Alcázar Rivas
 * @change : Enhanced al documento de la solicitud
 * Caso Mantis : 000941
 */
if(s.getDocumento() != null)
{

```

```

long a = this.getSecuencial(CDocumento.TABLE_NAME,null);
DocumentoEnhanced docEnhanced = s.getDocumento();
docEnhanced.setIdDocumento(String.valueOf(a));

//Tipo de Documento
TipoDocumentoEnhanced tipoDocEnh = docEnhanced.getTipoDocumento();
if(tipoDocEnh != null)
{
    TipoDocumentoEnhancedPk idTipoDocEnh = new TipoDocumentoEnhancedPk();
    idTipoDocEnh.idTipoDocumento = tipoDocEnh.getIdTipoDocumento();

    TipoDocumentoEnhanced tipoDocPers = (TipoDocumentoEnhanced)
pm.getObjectById(idTipoDocEnh,true);
    if(tipoDocPers == null)
    {
        throw new DAOException ("No se encontró el tipo de documento con el identificador:
"+idTipoDocEnh.idTipoDocumento);
    }
    docEnhanced.setTipoDocumento(tipoDocPers);
}

//Oficina Origen
OficinaOrigenEnhanced ofOrigEnh = docEnhanced.getOficinaOrigen();
if(ofOrigEnh != null)
{
    OficinaOrigenEnhancedPk idOfOrigenEnh = new OficinaOrigenEnhancedPk();
    idOfOrigenEnh.idOficinaOrigen = ofOrigEnh.getIdOficinaOrigen();

    OficinaOrigenEnhanced ofOrigPers = (OficinaOrigenEnhanced)
pm.getObjectById(idOfOrigenEnh,true);
    if(ofOrigPers == null)
    {
        throw new DAOException ("No se encontró la oficina origen con el identificador:
"+idOfOrigenEnh.idOficinaOrigen);
    }
    docEnhanced.setOficinaOrigen(ofOrigPers);
}
if(docEnhanced!=null&&s.getCirculo()!=null){
    docEnhanced.setCirculo(s.getCirculo().getIdCirculo());
}
s.setDocumento(docEnhanced);
}

```

**CLASE:**

**gov.sir.hermod.dao.impl.jdogenie.JDOTurnosDAO.java**

**Método:**

**protected void makeTransientTurno(TurnoEnhanced tr, PersistenceManager pm) throws DAOException {**

Código Insertado Linea 3513:

```

/**
 * @author : Julio Alcázar Rivas
 * @change : makeTransient al documento de la solicitud
 * Caso Mantis : 000941
 */
if(sc.getDocumento()!= null)
{
    pm.makeTransient (sc.getDocumento().getTipoDocumento());

    if(sc.getDocumento().getOficinaOrigen()!=null)
    {

```

```

pm.makeTransient(sc.getDocumento().getOficinaOrigen().getVereda().getMunicipio().getDepartamento());
m.makeTransient(sc.getDocumento().getOficinaOrigen().getVereda().getMunicipio());
pm.makeTransient(sc.getDocumento().getOficinaOrigen().getVereda());

pm.makeTransient(sc.getDocumento().getOficinaOrigen().getTipoOficina());
pm.makeTransient(sc.getDocumento().getOficinaOrigen());
}
pm.makeTransient(sc.getDocumento());
}

CLASE:
gov.sir.core.negocio.modelo.imprimibles.ImprimibleRecibo.java
Método:
public ImprimibleRecibo(Pago pago, Circulo circulo, String fechaImpresion, String tipoImprimible) {

```

**Código Insertado Linea 255:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : asigna al documento de la solicitud al imprimible
 * Caso Mantis : 000941
 */
List liquidaciones = pago.getLiquidacion().getSolicitud().getLiquidaciones();
Liquidacion liquidacion = (Liquidacion) liquidaciones.get(liquidaciones.size() - 1);
if(pago.getLiquidacion().getSolicitud() instanceof SolicitudCertificadoMasivo
    &&
((LiquidacionTurnoCertificadoMasivo)liquidacion).getTipoTarifa().equals(CTipoTarifa.EXENTO)){
    this.documento = ((SolicitudCertificadoMasivo)
pago.getLiquidacion().getSolicitud()).getDocumento();
}


```

**Método:**

```
private void imprimirCuerpoReciboCertificadoMasivo() {
```

**Código Insertado Linea 752:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Imprime el documento de la solicitud
 * Caso Mantis : 000941
 */
this.imprimirDocumento();


```

**Método:**

```
private void imprimirDatosCertificadosMasivos() {
```

**Código Insertado Linea 2287:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Eliminados dos espacios de lineas par formato del recibo
 * Caso Mantis : 000941
 */
this.imprimirLinea(ImprimibleConstantes.PLANO, "");


```

**NUEVO Método:**

```
protected void imprimirDocumento() {
```

**Código Insertado Linea 3561:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : Imprime el Documento de la solicitud.
 * Caso Mantis : 000941
 * @param
 * @return
 */
protected void imprimirDocumento() {
if(this.documento != null){


```

```

this.imprimirLinea(ImprimibleConstantes.PLANO, "");
this.imprimirLinea(ImprimibleConstantes.PLANO, "DATOS DOCUMENTO: ");

this.imprimirLinea(ImprimibleConstantes.PLANO,
    "Tipo Documento: "+ this.documento.getTipoDocumento().getNombre());

this.imprimirLinea(ImprimibleConstantes.PLANO,
    "Número Documento: "+ this.documento.getNumero());

Calendar hoy = Calendar.getInstance();
hoy.setTime(this.documento.getFecha());
String
fechaDoc=hoy.get(Calendar.DAY_OF_MONTH)+"/"+(hoy.get(Calendar.MONTH)+1)+"/"+hoy.get(Calendar.YEAR);

this.imprimirLinea(ImprimibleConstantes.PLANO, "Fecha documento: "+ fechaDoc);

this.imprimirLinea(ImprimibleConstantes.PLANO,"OFICINA DE PROCEDENCIA: ");

if (this.documento.getOficinaInternacional() == null){

    this.imprimirLinea(ImprimibleConstantes.PLANO, "Departamento: "+
this.documento.getOficinaOrigen().getVereda().getMunicipio().getDepartamento().getNombre());

    this.imprimirLinea(ImprimibleConstantes.PLANO, "Municipio: "+
this.documento.getOficinaOrigen().getVereda().getMunicipio().getNombre());

    this.imprimirLinea(ImprimibleConstantes.PLANO, "Tipo de Oficina de Origen: "+
this.documento.getOficinaOrigen().getTipoOficina().getNombre());

    this.imprimirLinea(ImprimibleConstantes.PLANO, "Nombre: "+
this.documento.getOficinaOrigen().getNombre());
    }else {

        this.imprimirLinea(ImprimibleConstantes.PLANO, "Oficina ubicación Internacional: "+
this.documento.getOficinaInternacional());
    }
}
}

CLASE:  

gov.sir.core.negocio.acciones.comun.ANPago.java  

Método:  

private EventoRespuesta procesarPagoMasivos(EvnPago evento) throws EventoException {  

Código Insertado Linea 1492:  

/*
 * @author : Julio Alcázar Rivas
 * @change : cambio de acceso long numRecibo y String year al if externo
 * Caso Mantis : 000941
 */
long numRecibo;
String year=Integer.toString(Calendar.getInstance().get(Calendar.YEAR));  

Código Insertado Linea 1501:  

/*
 * @author : Julio Alcázar Rivas
 * @change : Se agrega un secuencial a los recibos Masivos Exentos
 * Caso Mantis : 000941
 */
Log.getInstance().debug(ANPago.class, "@@LOGS_MASIVOS.ANPago.procesarPagoMasivos:
El Turno es Tipo EXENTO se va a imprimir masivos");

```

```

    numRecibo=hermod.getSecuencialMasivosExento(circulo.getIdCirculo(),year);
}

Código Insertado Linea 1521:
/*
 * @author : Julio Alcázar Rivas
 * @change : Se permite la impresion del recibo Masivos Exentos
 * Caso Mantis : 000941
 */
//if(imprimirRecibo){
```

**CLASE:**

**gov.sir.core.negocio.acciones.administracion.ANImprimirFolio.java**

**Método:**

**private EventoRespuesta reImprimirRecibos(EvnImprimirFolio evento) throws EventoException{**

**Código Insertado Linea 408:**

```

/*
 * @author : Julio Alcázar Rivas
 * @change : se cambia la reimpresion de la solicitud de certificados masivos exento
 * a tamaño carta
 * Caso Mantis : 000941
 */
if( solicitud instanceof gov.sir.core.negocio.modelo.SolicitudDevolucion ||
    (solicitud instanceof SolicitudCertificadoMasivo &&
    ((LiquidacionTurnoCertificadoMasivo)liquidacion).getTipoTarifa().equals(CTipoTarifa.EXENTO))) {
    impRec.setTamanoCarta( true );
}
```

## 018 Subtipos de atención Reparto Registro

### **REQUERIMIENTO**

Subtipos de atención Reparto Registro

SIR-230.

El aplicativo no toma en cuenta la clasificación hecha en la pantalla administrativa “administración de subtipos de atención para calificadores” en la asignación de documentos por tipo (reglamentos, Gran tamaño, copia).

### SEGUIMIENTO ADA:

Clarificar el requerimiento.

**OBSERVACIONES SNR:** El requerimiento es completamente claro: en la pantalla administrativa se realiza la asignación de subtipo de atención a cada funcionario calificador, al ejecutarse el reparto de documentos no se hace la asignación de acuerdo a la configuración realizada en la pantalla administrativa.

### **DESARROLLO**

Se envió correo solicitando la siguiente información el 02 de Octubre del 2009

### **DATOS ADICIONALES, NECESARIOS PARA LA EJECUCION DEL REQUERIMIENTO**

Se hace necesario que para el proceso de REPARTO REGISTRO se defina claramente la configuración que debe realizarse de los Subtipos de Atención (NORMAL, GRAN TAMAÑO, COPIA, REGLAMENTO y TESTAMENTO) en la pantalla administrativa PROCESO REGISTRO DE DOCUMENTOS -> Administración de Subtipos de Solicitud en la cual se debe determinar el conjunto de Tipos de Actos y Tipo de Solicitud vinculados a Cada Subtipo de Atención.

Información a tener en cuenta al momento de realizar la configuración:

Para cada Subtipo de Atención la configuración de Tipos de Actos y Tipo de Solicitud debe ser diferente por ejemplo si al configurar el Subtipo de Atención COPIA con el Tipo de Solicitud MAS MASTRICULAS y el Tipo de Acto COPIA esta configuración será exclusiva de este Subtipo de Atención.

La única excepción al punto 1 se da cuando dos Subtipo de Atención tiene la misma configuración de Tipo de Solicitud y el Tipo de Acto pero difieren en el número mínimo de Anotaciones como por ejemplo el Subtipo de Atención GRAN TAMAÑO que posee como mínimo 150 Anotaciones.

### **OBSERVACIONES DE SNR**

Respondieron en el correo del 24 de Agosto del 2010, por parte del funcionario Hector Elejalde Troyano a la información requerida anteriormente, lo siguiente;

Para la ejecución del presente Requerimiento se deben tener en cuenta las siguientes observaciones:

Referente a los campos SBTN\_ORDEN y UNIDADES\_TRABAJO no tenemos observaciones al respecto.

GRAN TAMAÑO	
Min_anotaciones	150
Min_Folios_Asociados	1
Min Actos del listado para este subtipo de atención	1

Cuando un turno contenga 10 o más folio asociados y uno o varios de ellos contengan más de 150 anotaciones el sistema debe seleccionarlo como GRAN TAMAÑO.

REGLAMENTO	
Min_anotaciones	
Min_Folios_Asociados	1
Min Actos del listado para este subtipo de atención	1

Para el subtipo de atención REGLAMENTOS, como condición debe tener mínimo un folio asociado y contener mínimo uno de los actos del listado, no se tiene en cuenta el número de anotaciones.

COPYS	
Min_anotaciones	
Min_Folios_Asociados	10
Min Actos del listado para este subtipo de atención	1

Para el Subtipo de Atención Normal se debe tener en cuenta todos los actos de liquidación

y que no cumplan las condiciones de Reglamentos, Copys, Gran Tamaño y Testamentos.

Para el sub-tipo de atención de testamentos se debe validar que el sub-tipo de solicitud sea Testamento.

Las validaciones para asignar un turno de documento a un subtipo de atención, se deben hacer con base a los actos liquidados según el listado adjunto, y las condiciones con números de anotaciones y números de folios asociados para cada sub-tipo.

Se advierte que se deben conservar las validaciones y reglas del negocio establecidas para el reparto de documentos, en las cuales no se hacen observaciones en este requerimiento.

Cuando se seleccione en subtipo de solicitud mas matriculas, el sistema debe buscar el subtipo-de atención que le corresponda (copy, reglamento, gran tamaño).

Cuando se seleccione en subtipo de solicitud normal, el sistema debe seleccionar el subtipo-de atención normal.

Cuando se seleccione en subtipo de solicitud sin matrícula, el sistema debe seleccionar el subtipo-de atención normal.

Cuando se seleccione en subtipo de solicitud sin matrícula y se agregan datos de antiguo sistema el sistema automáticamente lo debe redireccionar al rol de antiguo sistema revisión inicial.

Cuando se seleccione en subtipo de solicitud testamento, el sistema sabe que es un turno para enviar al rol testamentos.

Estas validaciones de (mas matriculas y testamentos) ya las tiene el sistema y está funcionando en forma correcta.

GRAN TAMAÑO	
ID	NOMBRE ACTO
1	VENTA
2	HIPOTECA
3	REMATE
4	SUCESION
5	COPIAS
6	PERMUTA
8	CANCELACION HIPOTECA
9	PATRIMONIO DE FAMILIA
10	EMBARGO
11	DEMANDA
12	REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000)
13	CANCELACION PATRIMONIO DE FAMILIA
15	HIPOTECA SIN LIMITE
16	ACLARACION

	17	CANCELACION EMBARGO	
	18	AFECTACION VIVIENDA FAMILIAR	
	19	MATRICULAS	
	20	TRANS. PERMISO	
	21	ENGLOBE	
	22	CONDICION RESOLUTORIA	
	23	USUFRUCTO (SIN CUANTIA PARA IMPTO)	
	24	ADJUDICACION BALDIO (SIN CUANTIA PARA IMPTO)	
	25	VALORIZACION	
	26	CANCELACION VALORIZACION	
	28	FIDUCIA	
	29	COMODATO	
	30	CANCELACION USUFRUCTO	
	31	SERVIDUMBRE (CON CUANTIA PARA IMPTO)	
	32	CANCELACION DEMANDA	
	33	LIQUIDACION COMUNIDAD	
	34	CANCELACION SERVIDUMBRE	
	35	DECLARACION CONSTRUCCION (CON CUANTIA PARA IMPTO)	
	36	DACION EN PAGO	
	37	ACLARACION DE LA CUANTIA	
	38	VIVIENDA DE INTERES SOCIAL	
	40	USUFRUCTO (CON CUANTIA PARA IMPTO)	
	41	ADJUDICACION BALDIO (CON CUANTIA PARA IMPTO)	
	42	SERVIDUMBRE (SIN CUANTIA PARA IMPTO)	
	43	DECLARACION CONSTRUCCION (SIN CUANTIA PARA IMPTO)	
	44	RESTITUCION FIDUCIA (SIN CUANTIA PARA IMPTO)	
	45	HIPOTECA SUSTITUCION GARANTIA (CON CUANTIA PARA IMPTO)	
	46	CESION DE BIENES FISCALES (CON CUANTIA PARA IMPUESTO)	
	47	RESTITUCION FIDEICOMISO (SIN CUANTIA PARA IMPTO)	
	48	ESCISION (CON CUANTIA PARA IMPTO)	
	49	LIQUIDACION SOCIEDAD CONYUGAL (SIN CUANTIA PARA IMPTO)	
	50	FUSION (SIN CUANTIA PARA IMPTO)	
	51	CESION (SIN CUANTIA PARA IMPTO)	
	52	CANCELACION OFERTA DE COMPRA	
	53	AMPLIACION DE HIPOTECA	
	54	CANCELACION PACTO DE RETROVENTA	
	60	RESTITUCION FIDUCIA	
	61	DONACION INTERNACIONAL	
	62	SUCESION ESPECIAL	
	63	LIQUIDACION SOCIEDAD CONYUGAL ESPECIAL	
	64	TESTAMENTOS	
	65	TRANSFERENCIA NUDA PROPIEDAD	
	66	LIQUIDACION SOCIEDAD COMERCIAL	
	68	LEASING	
	69	HIPOTECA SUSTITUCION GARANTIA (SIN CUANTIA PARA IMPTO)	
	70	CANCELACION HIPOTECA ESPECIAL	

	71	CANCELACION COMODATO	
	75	ADMINISTACION ANTICRETICA	
	76	CANCELACION CONDICION RESOLUTORIA	
	77	CANCELACION ADMINISTRACION ANTICRETICA	
	78	CANCELACION AFECTACION FAMILIAR	
	79	TRANSFERENCIA DE DOMINIO ANT.31-12-90	
	80	CESION BIENES FISCALES	
	81	VENTA INDEN.PLANOS CATASTRALES	
	82	CONSTITUCION SOCIEDAD COMERCIAL	
	83	CANCELACION LEASING	
	84	ARRENDAMIENTO	
	86	FIDEICOMISO ESPECIAL	
	87	REVOCATORIA DE TESTAMENTOS	
	88	CANCELACION FIDUCIA	
	89	RESTITUCION FIDEICOMISO (CON CUANTIA PARA IMPTO)	
	90	FIDEICOMISO CIVIL CON TRANSFERENCIA	
	91	RESTITUCION EN FIDUCIA MERCANTIL	
	93	RESCISION DE CONTRATO	
	94	RESOLUCION DE CONTRATO	
	95	RESCILIACION	
	96	CONCORDATO	
	97	LIQUIDACION OBLIGATORIA	
	98	LAUDO ARBITRAL	
	99	INSCRIPCION FOLIO ADICIONAL	
	100	ESCISION (SIN CUANTIA PARA IMPTO)	
	103	DONACION	
	120	LIQUIDACION SOCIEDAD CONYUGAL (CON CUANTIA PARA IMPTO)	
	200	ACTUALIZACION DE NOMENCLATURA	
	201	INSCRIPCION DE OFERTA	
	202	DACION EN PAGO ESPECIAL	
	203	PACTO RETROVENTA	
	204	CANCELACION ARRENDAMIENTO	
	210	HIPOTECA ESPECIAL	
	250	FUSION (CON CUANTIA PARA IMPTO)	
	251	CESION (CON CUANTIA PARA IMPTO)	
	300	ESTAMPILLA	
	474	REGISTRO UNICO DE PREDIOS Y TERRITORIOS ABANDONADOS -RUPTA	
	555	DEPURACION	
	777	VARIOS	
	888	OTROS	
	<b>REGLAMENTO</b>		
	ID	NOMBRE ACTO	
	7	DIVISION MATERIAL	
	14	PROPIEDAD HORIZONTAL	
	27	DESENGLOBE	

	72	REFORMA PROP.HORIZONTAL O COPROPIEDAD	
	73	REGIMEN DE COPROPIEDAD	
	74	CANCEL.PROPIEDAD HORIZONTAL O COPROP.	
	121	LOTEO	
<b>COPYS</b>			
	ID	NOMBRE ACTO	
	1	VENTA	
	2	HIPOTECA	
	3	REMATE	
	4	SUCESION	
	5	COPIAS	
	6	PERMUTA	
	8	CANCELACION HIPOTECA	
	9	PATRIMONIO DE FAMILIA	
	10	EMBARGO	
	11	DEMANDA	
	12	REPRODUCCION CONSTANCIA DE INSCRIPCION (ART. 9 DECRETO 1428/2000)	
	13	CANCELACION PATRIMONIO DE FAMILIA	
	15	HIPOTECA SIN LIMITE	
	16	ACLARACION	
	17	CANCELACION EMBARGO	
	18	AFECTACION VIVIENDA FAMILIAR	
	19	MATRICULAS	
	20	TRANS. PERMISO	
	21	ENGLOBE	
	22	CONDICION RESOLUTORIA	
	23	USUFRUCTO (SIN CUANTIA PARA IMPTO)	
	24	ADJUDICACION BALDIO (SIN CUANTIA PARA IMPTO)	
	25	VALORIZACION	
	26	CANCELACION VALORIZACION	
	28	FIDUCIA	
	29	COMODATO	
	30	CANCELACION USUFRUCTO	
	31	SERVIDUMBRE (CON CUANTIA PARA IMPTO)	
	32	CANCELACION DEMANDA	
	33	LIQUIDACION COMUNIDAD	
	34	CANCELACION SERVIDUMBRE	
	35	DECLARACION CONSTRUCCION (CON CUANTIA PARA IMPTO)	
	36	DACION EN PAGO	
	37	ACLARACION DE LA CUANTIA	
	38	VIVIENDA DE INTERES SOCIAL	
	40	USUFRUCTO (CON CUANTIA PARA IMPTO)	
	41	ADJUDICACION BALDIO (CON CUANTIA PARA IMPTO)	
	42	SERVIDUMBRE (SIN CUANTIA PARA IMPTO)	

	43	DECLARACION CONSTRUCCION (SIN CUANTIA PARA IMPTO)	
	44	RESTITUCION FIDUCIA (SIN CUANTIA PARA IMPTO)	
	45	HIPOTECA SUSTITUCION GARANTIA (CON CUANTIA PARA IMPTO)	
	46	CESION DE BIENES FISCALES (CON CUANTIA PARA IMPUESTO)	
	47	RESTITUCION FIDEICOMISO (SIN CUANTIA PARA IMPTO)	
	48	ESCISION (CON CUANTIA PARA IMPTO)	
	49	LIQUIDACION SOCIEDAD CONYUGAL (SIN CUANTIA PARA IMPTO)	
	50	FUSION (SIN CUANTIA PARA IMPTO)	
	51	CESION (SIN CUANTIA PARA IMPTO)	
	52	CANCELACION OFERTA DE COMPRA	
	53	AMPLIACION DE HIPOTECA	
	54	CANCELACION PACTO DE RETROVENTA	
	60	RESTITUCION FIDUCIA	
	61	DONACION INTERNACIONAL	
	62	SUCESION ESPECIAL	
	63	LIQUIDACION SOCIEDAD CONYUGAL ESPECIAL	
	64	TESTAMENTOS	
	65	TRANSFERENCIA NUDA PROPIEDAD	
	66	LIQUIDACION SOCIEDAD COMERCIAL	
	68	LEASING	
	69	HIPOTECA SUSTITUCION GARANTIA (SIN CUANTIA PARA IMPTO)	
	70	CANCELACION HIPOTECA ESPECIAL	
	71	CANCELACION COMODATO	
	75	ADMINISTACION ANTICRETICA	
	76	CANCELACION CONDICION RESOLUTORIA	
	77	CANCELACION ADMINISTRACION ANTICRETICA	
	78	CANCELACION AFECTACION FAMILIAR	
	79	TRANSFERENCIA DE DOMINIO ANT.31-12-90	
	80	CESION BIENES FISCALES	
	81	VENTA INDEN.PLANOS CATASTRALES	
	82	CONSTITUCION SOCIEDAD COMERCIAL	
	83	CANCELACION LEASING	
	84	ARRENDAMIENTO	
	86	FIDEICOMISO ESPECIAL	
	87	REVOCATORIA DE TESTAMENTOS	
	88	CANCELACION FIDUCIA	
	89	RESTITUCION FIDEICOMISO (CON CUANTIA PARA IMPTO)	
	90	FIDEICOMISO CIVIL CON TRANSFERENCIA	
	91	RESTITUCION EN FIDUCIA MERCANTIL	
	93	RESCISION DE CONTRATO	
	94	RESOLUCION DE CONTRATO	
	95	RESCILIACION	
	96	CONCORDATO	
	97	LIQUIDACION OBLIGATORIA	
	98	LAUDO ARBITRAL	

		99	INSCRIPCION FOLIO ADICIONAL	
		100	ESCISION (SIN CUANTIA PARA IMPTO)	
		103	DONACION	
		120	LIQUIDACION SOCIEDAD CONYUGAL (CON CUANTIA PARA IMPTO)	
		200	ACTUALIZACION DE NOMENCLATURA	
		201	INSCRIPCION DE OFERTA	
		202	DACION EN PAGO ESPECIAL	
		203	PACTO RETROVENTA	
		204	CANCELACION ARRENDAMIENTO	
		210	HIPOTECA ESPECIAL	
		250	FUSION (CON CUANTIA PARA IMPTO)	
		251	CESION (CON CUANTIA PARA IMPTO)	
		300	ESTAMPILLA	
		474	REGISTRO UNICO DE PREDIOS Y TERRITORIOS ABANDONADOS -RUPTA	
		555	DEPURACION	
		777	VARIOS	
		888	OTROS	
		<b>TESTAMENTOS</b>		
	ID	NOMBRE ACTO		
	64	TESTAMENTOS		
		<b>Observaciones:</b>		
		Script Bases de Datos:		
		Ejecutar los Script en el siguiente orden delete_sir_op_sub_atenc_tipo_acto.sql delete_Subtipos.sql SIR_OP_SUB_ATENC_TIPO_ACTO.sql		
		Creacion De Clases:		
		NINGUNA		
		Se modificaron los siguientes archivos:		
		<b>CLASE:</b> <b>gov.sir.core.negocio.acciones.registro.ANReparto.java</b>		
		<b>Método:</b> <b>private Map calcularPromedioPorSubtipoAtencion(List allTurnos, Map mapSubtipoUsuarios)</b>		
		<b>Código Línea 892:</b>		
		if(numeroTurnosPorSubtipo.containsKey(sr.getSubtipoAtencion())){ Long numero = (Long)numeroTurnosPorSubtipo.get(sr.getSubtipoAtencion()); Long nuevoNumero = new Long(numero.longValue() + 1); numeroTurnosPorSubtipo.put(sr.getSubtipoAtencion(), nuevoNumero); }		
		<b>Por:</b>		
		/* * @author : Julio Alcazar Rivas * @change : Se modifica el if para realizar el balanceo del reparto de acuerdo al numero de calificadores por subtipo * Caso Mantis : 0002593 */ if(mapSubtipoUsuarios.containsKey(sr.getSubtipoAtencion())){ Long nuevoNum = new Long(0);		

	<pre> if (numeroTurnosPorSubtipo.containsKey(sr.getSubtipoAtencion())) {     nuevoNum = (Long) numeroTurnosPorSubtipo.get(sr.getSubtipoAtencion()); } Long nuevoNumero = new Long(nuevoNum.longValue() + 1); numeroTurnosPorSubtipo.put(sr.getSubtipoAtencion(), nuevoNumero); } </pre>	
	<b>CLASE:</b>	
	<b>gov.sir.hermes.dao.impl.jdogenie.JDOSolicitudesDAO.java</b>	
	<b>Método:</b>	
	<b>private boolean cumpleReglasNegocio(SubtipoAtencionEnhanced subtipo, SolicitudRegistroEnhanced s, PersistenceManager pm)</b>	
	<b>Código Línea 892:</b>	
	<pre> if(s.getSolicitudFolios().size()&lt;=subtipo.getMinFolios()){  <b>Por:</b> /*  * @author : Julio Alcazar Rivas  * @change : Se modifica el if para que se return false solo cuando el numero de folios de la solicitud sea menor que  *           el numero minimo de folios del subtipo evaluado.  * Caso Mantis : 0002593 */ if(s.getSolicitudFolios().size()&lt;subtipo.getMinFolios()){     return false; } </pre>	

## 099 Reporte 15 y 18

### REQUERIMIENTO

**Reporte 15:** Informe cuentas por pagar devoluciones (día xx al día yy),

Parámetros de Consulta fecha inicial de aprobación pago y final de aprobación pago: este rango de fechas corresponden a las fechas cuando se avanza el turno de la fase de Recurso Devolución a Pago Devolución, es decir que la máxima fase en la que debe encontrarse el turno es Pago Devolución.

Los campos que debe mostrar son: Radicación, fecha de radicación del turno, turno asociado, numero consignación o cheque del turno asociado, valor derechos, valor impuestos, valor certificados, pago exceso, valor a pagar, numero de resolución, fecha de resolución (dd-mm-yyyy), solicitantes. Al final se deben Totalizar todos los items.

**Reporte 18:** Informe de devoluciones pagadas en cheque (día xx al día yy), este reporte debe llamarse Informe de devoluciones pagadas (forma de pago) del (día xx al día yy).

Parámetros de Consulta fecha inicial de pago y final de pago: este rango de fechas corresponden a las fechas cuando se avanza el turno de la fase de Pago Devolución ha Entregado, es decir que la máxima fase en la que debe encontrarse el turno es Entregado.

. Los campos que debe mostrar son: Radicación, fecha de pago, Turno Asociado, Valor, Solicitantes, Derechos, Impuestos, Certificados, Numero Resolución, fecha resolución (dd-mm-yyyy), forma de pago que debe ser Tipo Pago (Cheque=Cheque; Nota Debito= Nota D. Efectivo=Efect. Al final se deben Totalizar todos los items. Adicionar un resumen de cada una de las formas de pago (Efectivo, Cheque y Nota Debito)

**Reporte 79:** Informe de devoluciones pagadas del (día xx al día yy)

En la columna de fecha de pago está mostrando una fecha que no corresponde a la fecha de pago, en este campo debe mostrar la fecha del Pago del turno de Devolución (Fase Pago Devolución).

### DESARROLLO

#### Análisis de Requerimiento:

REPORTE 15 - INFORME CUENTAS POR PAGAR DEVOLUCIONES

Se realizan modificaciones en el procedimiento SP\_REPORTE\_15

Se agrego a la consulta la tabla SIR\_OP\_TURNO\_HISTORIA

```
/*AHERRENO 12/08/2010/
, sir_op_turno_historia turno_historia
```

**ANTES:**

```
(SELECT MAX(turno_historia.trhs_fecha_registro)
FROM sir_op_turno_historia turno_historia
WHERE turno_devolucion.anio = turno_historia.anio
AND turno_devolucion.id_circulo = turno_historia.id_circulo
AND turno_devolucion.id_proceso = turno_historia.id_proceso
AND turno_devolucion.id_turno = turno_historia.id_turno
AND (turno_historia.trhs_id_fase = 'DEV_PAGO_DEVOLUCION' AND turno_historia.trhs_id_fase_anterior='DEV_RECURSOS'))
```

**POR:**

```
/*AHERRENO 12/08/2010/
,turno_historia.trhs_fecha_registro      fecha_registro
```

**ANTES:**

```
AND oficio.FCIO_FECHA_CREACION
BETWEEN TO_DATE( '2010-08-12', 'YYYY-MM-DD')
AND TO_DATE( '2010-08-30', 'YYYY-MM-DD') + 1*/
```

**POR:**

```
/*AHERRENO 12/08/2010/
AND turno_historia.trhs_fecha_registro
BETWEEN TO_DATE( '2010-08-12', 'YYYY-MM-DD')
AND TO_DATE( '2010-08-30', 'YYYY-MM-DD') + 1
```

**ANTES**

```
--Que hay pasado por la fase de resolucion
AND EXISTS (
SELECT *
FROM sir_op_turno_historia turno_historia
WHERE turno_devolucion.anio = turno_historia.anio
AND turno_devolucion.id_circulo = turno_historia.id_circulo
AND turno_devolucion.id_proceso = turno_historia.id_proceso
AND turno_devolucion.id_turno = turno_historia.id_turno
AND turno_historia.trhs_id_fase = 'DEV_RESOLUCION')
```

**POR.**

```
--Que haya pasado por la fase de resolucion
/*AHERRENO 12/08/2010/
AND turno_devolucion.anio = turno_historia.anio
AND turno_devolucion.id_circulo = turno_historia.id_circulo
AND turno_devolucion.id_proceso = turno_historia.id_proceso
AND turno_devolucion.id_turno = turno_historia.id_turno
AND turno_historia.trhs_id_fase = 'DEV_RESOLUCION'
```

**ANTES:**

```
--la fase del turno es aprobado. Se aprueba despues de recursos,su fase es resolucion y respuesta confirmar
and ( (turno_devolucion.trno_id_fase='DEV_PAGO_DEVOLUCION' and
turno_devolucion.trno_ultima_respuesta='CONFIRMAR')
OR(turno_devolucion.trno_id_fase='DEV_RECURSOS' and turno_devolucion.trno_ultima_respuesta='CONFIRMAR' ))
```

**POR:**

```
--la fase del turno es aprobado. Se aprueba despues de recursos, fase PAGO DEVOLUCION y respuesta confirmar
/*AHERRENO 12/08/2010/
AND (turno_devolucion.trno_id_fase = 'DEV_PAGO_DEVOLUCION' AND turno_devolucion.trno_ultima_respuesta =
'CONFIRMAR')
```

**SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java**

**LINEA528**

```
//AHERRENO 16/11/2010
//Reporte 15 - Se agrega parametro P_USUARIO_LOG
public static final String REPORTE_15_PARAM_CMDKEY = "reporte-15.jasper";
public static final String REPORTE_15_NOMBRE = "";
public static final String REPORTE_15_PARAM_PLIQUIDFECHAINI = "P_LIQUIDACION_FECHA_INI";
public static final String REPORTE_15_PARAM_PLIQUIDFECHAFIN = "P_LIQUIDACION_FECHA_FIN";
public static final String REPORTE_15_PARAM_PCIRCULONOMBRE = "P_CIRCULO_ID";
```

```

public static final String REPORTE_15_PARAM_USUARIOLOG = "P_USUARIO_LOG";

LINEA 549
//AHERRENO 16/11/2010
//Reporte 18 - Se agrega parametro P_USUARIO_LOG
public static final String REPORTE_18_PARAM_CMDKEY = "reporte-18.jasper";
public static final String REPORTE_18_NOMBRE = "";
public static final String REPORTE_18_PARAM_PFECHAINI = "P_FECHA_INI";
public static final String REPORTE_18_PARAM_PFECHAFIN = "P_FECHA_FIN";
public static final String REPORTE_18_PARAM_PCIRCULONOMBRE = "P_CIRCULO_ID";
public static final String REPORTE_18_PARAM_USUARIOLOG = "P_USUARIO_LOG";

LINEA 739
//AHERRENO 16/11/2010
//Reporte 79 - Se agrega parametro P_USUARIO_LOG
public static final String REPORTE_79_PARAM_CMDKEY = "reporte-79.jasper";
public static final String REPORTE_79_NOMBRE = "";
public static final String REPORTE_79_PARAM_PFECHAINI = "P_FECHA_INI";
public static final String REPORTE_79_PARAM_PFECHAFIN = "P_FECHA_FIN";
public static final String REPORTE_79_PARAM_PCIRCULONOMBRE = "P_CIRCULO_ID";
public static final String REPORTE_79_PARAM_USUARIOLOG = "P_USUARIO_LOG";

LINEA 3750
//AHERRENO 16/11/2010
//Reporte 15 - Se agrega parametro P_USUARIO_LOG
private void do_Reporte15_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_15_PARAM_PLIQUIDFECHAINI);

    String param_02 = request.getParameter(REPORTE_15_PARAM_PLIQUIDFECHAFIN);

    String param_03 = request.getParameter(REPORTE_15_PARAM_PCIRCULONOMBRE);

    String param_04 = request.getParameter(REPORTE_15_PARAM_USUARIOLOG);

    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {
        exception.addError("parametro: Fecha Inicio: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio : La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Fecha Fin: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_02);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_02 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Fin : La fecha es inválida" + e.getMessage());
        }
    }
}

```

```

// apply validators
if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {
    exception.addError("parametro: Circulo: Debe digitar un valor");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    }
}

// apply validators
if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
    exception.addError("parametro: Usuario Logueado: Debe digitar un valor");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
        exception.addError("parametro: Usuario Logueado: Debe digitar un valor");
    }
}

do_Reporte15_SaveState(request);

// raise -----
if (exception.getErrores().size() > 0) {
    throw exception;
}
// -----
String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY, REPORTE_15_PARAM_PLIQUIDFECHAINI,
    REPORTE_15_PARAM_PLIQUIDFECHAFIN, REPORTE_15_PARAM_PCIRCULONOMBRE,
    REPORTE_15_PARAM_USUARIOLOG
};

String[] values = new String[]{
    REPORTE_15_PARAM_CMDKEY, param_01, param_02, param_03, param_04
};

String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_Reporte15_RemoveState(request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);

}

private void do_Reporte15_Back_Action(HttpServletRequest request) {
    do_Reporte15_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
} // :do_Reporte15_Back_Action

private void do_Reporte15_SaveState(HttpServletRequest request) {
    HttpSession session;

    String[] itemIds = new String[]{
        REPORTE_15_PARAM_PLIQUIDFECHAINI, REPORTE_15_PARAM_PLIQUIDFECHAFIN,
        REPORTE_15_PARAM_PCIRCULONOMBRE, REPORTE_15_PARAM_USUARIOLOG
    };
    session = request.getSession();
}

```

```

        save_PageltemsState(itemIds, request, session);
    } // :do_Reporte15_SaveState

    private void do_Reporte15_RemoveState(HttpServletRequest request) {
        HttpSession session;
        String[] itemIds = new String[]{
            REPORTE_15_PARAM_PLIQUIDFECHAINI, REPORTE_15_PARAM_PLIQUIDFECHAFIN,
            REPORTE_15_PARAM_PCIRCULONOMBRE, REPORTE_15_PARAM_USUARIOLOG
        };
        session = request.getSession();
        delete_PageltemsState(itemIds, request, session);
    } // :do_Reporte15_RemoveState

```

**LINEA 4150**

```

//AHERRENO 16/11/2010
//Reporte 18 - Se agrega parametro P_USUARIO_LOG
private void do_Reporte18_Send_Action(HttpServletRequest request)
    throws ValidacionParametrosException {
    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();
    String param_01 = request.getParameter(REPORTE_18_PARAM_PFECHAINI);
    String param_02 = request.getParameter(REPORTE_18_PARAM_PFECHAFIN);
    String param_03 = request.getParameter(REPORTE_18_PARAM_PCIRCULONOMBRE);
    String param_04 = request.getParameter(REPORTE_18_PARAM_USUARIOLOG);
    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {
        exception.addError("parametro: Fecha Inicio: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio : La fecha es inválida" + e.getMessage());
        }
    }
    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Fecha Fin: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_02);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_02 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Fin : La fecha es inválida" + e.getMessage());
        }
    }
    // apply validators
    if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
            exception.addError("parametro: Circulo: Debe digitar un valor");
        }
    }
    // apply validators
    if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
        exception.addError("parametro: Usuario Logueado: Debe digitar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
            exception.addError("parametro: Usuario Logueado: Debe digitar un valor");
        }
    }
}

```

```

do_Reporte18_SaveState(request);
// raise -----
if (exception.getErrores().size() > 0) {
    throw exception;
}
// ----

String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY, REPORTE_18_PARAM_PFECHAINI,
    REPORTE_18_PARAM_PFECHAFIN, REPORTE_18_PARAM_PCIRCULONOMBRE,
    REPORTE_18_PARAM_USUARIOLOG
};
String[] values = new String[]{
    REPORTE_18_PARAM_CMDKEY, param_01, param_02, param_03, param_04
};
String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);
do_Reporte18_RemoveState(request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

private void do_Reporte18_Back_Action(HttpServletRequest request) {
    do_Reporte18_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
} // :do_Reporte18_Back_Action
private void do_Reporte18_SaveState(HttpServletRequest request) {
    HttpSession session;

    String[] itemIds = new String[]{
        REPORTE_18_PARAM_PFECHAINI, REPORTE_18_PARAM_PFECHAFIN,
        REPORTE_18_PARAM_PCIRCULONOMBRE, REPORTE_18_PARAM_USUARIOLOG
    };
    session = request.getSession();
    save_PageItemsState(itemIds, request, session);
} // :do_Reporte18_SaveState
private void do_Reporte18_RemoveState(HttpServletRequest request) {
    HttpSession session;
    String[] itemIds = new String[]{
        REPORTE_18_PARAM_PFECHAINI, REPORTE_18_PARAM_PFECHAFIN,
        REPORTE_18_PARAM_PCIRCULONOMBRE, REPORTE_18_PARAM_USUARIOLOG
    };
    session = request.getSession();
    delete_PageItemsState(itemIds, request, session);
} // :do_Reporte18_RemoveState

```

#### LINEA 10587

```

//AHERRENO 16/11/2010
//Reporte 79 - Se agrega parametro P_USUARIO_LOG
private void do_Reporte79_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {
    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();
    String param_01 = request.getParameter(REPORTE_79_PARAM_PFECHAINI);
    String param_02 = request.getParameter(REPORTE_79_PARAM_PFECHAFIN);
    String param_03 = request.getParameter(REPORTE_79_PARAM_PCIRCULONOMBRE);
    String param_04 = request.getParameter(REPORTE_79_PARAM_USUARIOLOG);
    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {
        exception.addError("parametro: Fecha Inicio: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio : La fecha es inválida" + e.getMessage());
        }
    }
    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Fecha Fin: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;

```

```

try {
    fecha = DateFormatUtil.parse(param_02);
    send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
    param_02 = send1;
} catch (Exception e) {
    exception.addError("parametro: Fecha Fin : La fecha es inválida" + e.getMessage());
}
// apply validators
if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {
    exception.addError("parametro: Circulo: Debe digitar un valor");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    }
}
// apply validators
if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
    exception.addError("parametro: Usuario Logueado: Debe digitar un valor");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
        exception.addError("parametro: Usuario Logueado: Debe digitar un valor");
    }
}
String[] itemIds = new String[]{
    REPORTE_79_PARAM_PFECHAINI, REPORTE_79_PARAM_PFECHAFIN,
    REPORTE_79_PARAM_PCIRCULONOMBRE, REPORTE_79_PARAM_USUARIOLOG
};
do_ReporteXX_SaveState(itemIds, request);
// raise -----
if (exception.getErrores().size() > 0) {
    throw exception;
}
// -----
String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY, REPORTE_79_PARAM_PFECHAINI,
    REPORTE_79_PARAM_PFECHAFIN, REPORTE_79_PARAM_PCIRCULONOMBRE,
    REPORTE_79_PARAM_USUARIOLOG
};
String[] values = new String[]{
    REPORTE_79_PARAM_CMDKEY, param_01, param_02, param_03, param_04
};
String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);
do_ReporteXX_RemoveState(itemIds, request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

```

#### SIR-APP\web\WEB-INF\jspSeguro\administracion\ADMINREPORTES\_REPORT18\_PARAMFORM

##### LINEA 417

```

<% //AHERRENDO 16/11/2010
//Reporte 18 - Se agrega parametro P_USUARIO_LOG
Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
%>
<input type="hidden" name="<%=AWReportes.REPORTE_18_PARAM_USUARIOLOG%>" 
id="<%=AWReportes.REPORTE_18_PARAM_USUARIOLOG%>" 
value="<%=usuario.getUsername()%>">

```

#### SIR-APP\web\WEB-INF\jspSeguro\administracion\ADMINREPORTES\_REPORT79\_PARAMFORM

##### LINEA 396

```

<% //AHERRENDO 16/11/2010
//Reporte 79 - Se agrega parametro P_USUARIO_LOG
Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
%>
<input type="hidden" name="<%=AWReportes.REPORTE_79_PARAM_USUARIOLOG%>" 
id="<%=AWReportes.REPORTE_79_PARAM_USUARIOLOG%>" 
value="<%=usuario.getUsername()%>">

```

#### SIR-APP\web\WEB-INF\jspSeguro\administracion\ADMINREPORTES\_REPORT15\_PARAMFORM

##### Línea 338

```
<td align="left" width="150"><div align="right" >Fecha Inicial:</div></td>
```

##### Por

```
<td align="left" width="150"><div align="right" >Fecha Inicial de aprobaci&ocuten pago:</div></td>
```

## Línea 369

```
<td align="left" width="150"><div align="right" >Fecha Final:</div></td>
```

## Por

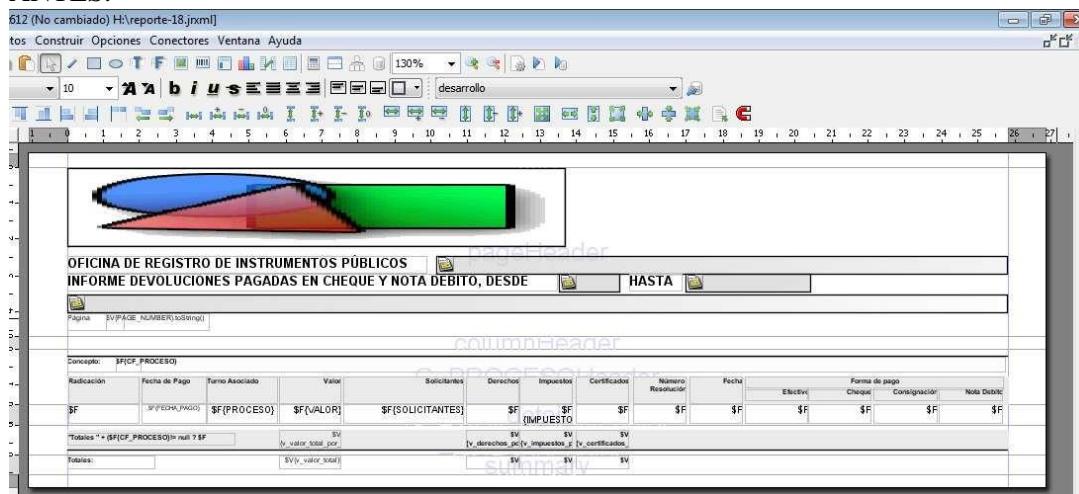
```
<td align="left" width="150"><div align="right" >Fecha Final de aprobaci&ocuten pago:</div></td>
```

## LINEA 421

```
<% //AHERRENO 16/11/2010  
//Reporte 15 - Se agrega parametro P_USUARIO_LOG  
Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);  
%>  
<input type="hidden" name="<%=>AWReportes.REPORTE_15_PARAM_USUARIOLOG%"  
id="<%=>AWReportes.REPORTE_15_PARAM_USUARIOLOG%"  
value="<%=>usuario.getUsername()%>">
```

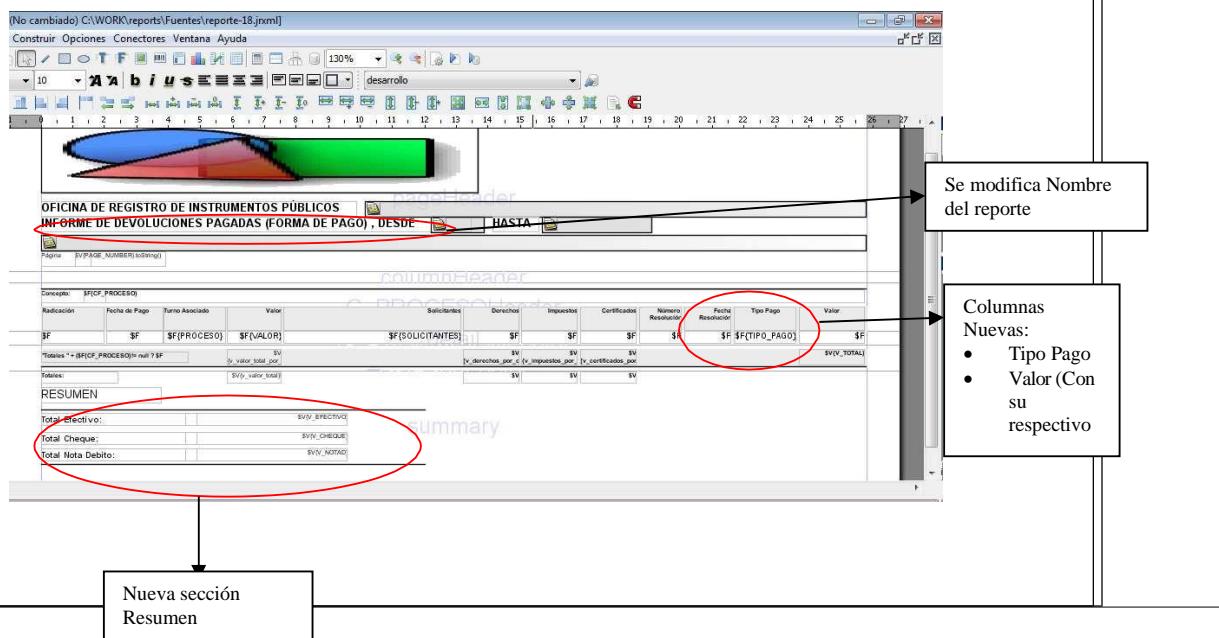
Se modifica archivo reporte-18.jrxml

## ANTES:



## AHORA:

Se modificó el tamaño de todos los campos para una mejor visualización



## REPORTE 79 – INFORME DEVOLUCIONES PAGADAS

Se modifica query reporte-79.jrxml

ANTES:

sir\_op\_pago.pago\_fecha\_creacion                            pago\_fecha,

POR:

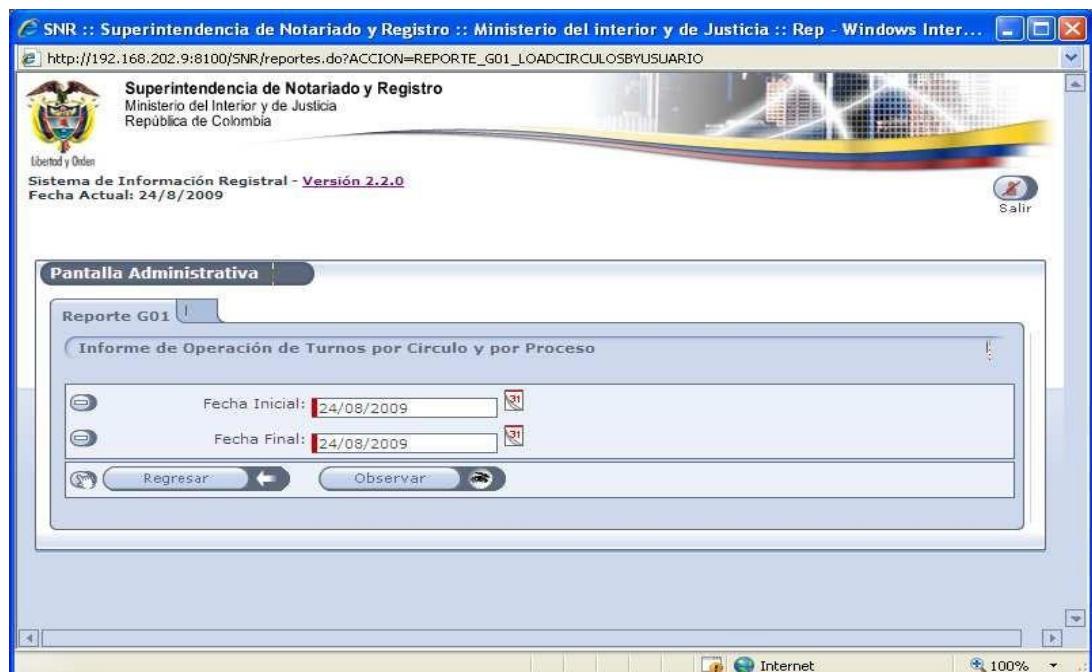
/\*AHERRENO 13/08/2010\*/

sir\_op\_pago.pago\_fecha                                    pago\_fecha,

## 105 Reporte G01

### REQUERIMIENTO

Reporte G01 – Informe de Operación de turnos por Círculo y por Proceso: Este reporte debe permitir seleccionar desde un rango de fecha los turnos radicados, en trámite y entregados de todos los procesos de cada Circulo Registral.



El rango de fechas debe ser: Fecha inicial de Radicación y Fecha final de Radicación., la información que debe reflejarse es la siguiente:

TITULO: OFICINA DE REGISTRO QUE GENERA EL REPORTE Y CODIGO

TITULO DEL REPORTE: INFORME DE OPERACIÓN DE TURNOS POR CÍRCULO Y POR PROCESO, FECHA Y HORA DE LA CUAL SE GENERA EL REPORTE, USUARIO QUE GENERA EL REPORTE, RANGO DE FECHAS, PAGINAS No.

LAS COLUMNAS QUE DEBE MOSTRAR SON: PROCESOS, RADICADO, EN TRAMITE Y ENTREGA, INDICANDO TOTALES POR PROCESO

### DESARROLLO

El rango de fechas debe ser: Fecha inicial de Radicación y Fecha final de Radicación., la

información que debe reflejarse es la siguiente:

**TITULO: OFICINA DE REGISTRO QUE GENERA EL REPORTE Y CODIGO**

**TITULO DEL REPORTE: INFORME DE OPERACIÓN DE TURNOS POR CIRCULO Y POR PROCESO, FECHA Y HORA DE LA GUAL SE GENERA EL REPORTE, USUARIO QUE GENERA EL REPORTE, RANGO DE FECHAS, PAGINAS No.**

LAS COLUMNAS QUE DEBE MOSTRAR SON: PROCESOS, RADICADO, EN TRÁMITE Y ENTREGA, INDICANDO TOTALES POR PROCESO. Para aprovechar el espacio de la hoja se desarollo el reporte agrupando por proceso, en el cual se muestran dos columnas turno y estado (En Tramite y Finalizado)

**Radicado:** Es la sumatoria de los turnos que se encuentran en estado “Tramite y Finalizado”

**Tramite:** Turnos que no se han finalizado.

**Finalizado:** Turnos entregados.

Script Bases de Datos: SIR.SP\_REPORTE\_G01

Creación De Clases:

NINGUNA (el mismo formato de modificación clases o jsp)

Se modificó el archivo **[reporte-g-01.jrxml]**

**REPORT : SIR.SP\_REPORTE\_G01**

```
Código Nuevo SIR.SP_REPORTE_G01

/*********************************/
/* GCABRERA 06/10/2010
   Reporte G01 – Informe de Operación de turnos por Círculo y por Proceso*/
/*********************************/
create or replace
PROCEDURE SP_REPORTE_G01(P_CIRCULO VARCHAR2, P_FECHA_INICIO VARCHAR2, P_FECHA_FIN
VARCHAR2, P_SEC_PROCESO NUMBER)
IS
BEGIN
    INSERT INTO SIR.SIR_OP_Reporte_G01(SEC_PROCESO, TURNO, PROCESO, ESTADO,
FECHA_RADICA, ID_PROCESO)
        SELECT P_SEC_PROCESO, TU.TRNO_ID_WORKFLOW TURNO, P.PRCs_NOMBRE PROCESO,
        CASE WHEN TU.TRNO_FECHA_FIN IS NULL THEN 'EN TRÁMITE'
        ELSE 'FINALIZADO'
        END ESTADO,
        TU.trno_fecha_creacion FECHA_RADICA,
        tu.id_proceso ID_PROCESO
        FROM SIR.SIR_OP_TURNO TU INNER JOIN SIR.SIR_OP_Proceso P ON TU.ID_PROCESO =
P.ID_PROCESO
        WHERE tu.id_circulo = P_CIRCULO
                AND TU.trno_fecha_creacion BETWEEN TO_DATE(P_FECHA_INICIO,
'YYYY-MM-DD')
                AND TO_DATE(P_FECHA_FIN, 'YYYY-MM-DD')+1
                AND TU.TRNO_ANULADO = 'N'
        ORDER BY tu.id_proceso, TU.trno_fecha_creacion ASC;
        COMMIT;
END;
```

Se creó índice IDX\_TRNO\_FECRE para la tabla SIR\_OP\_TURNO

	<pre>CREATE INDEX "SIR"."IDX_TRNO_FCRE" ON "SIR"."SIR_OP_TURNO" ("TRNO_FECHA_CREACION") TABLESPACE "TBS_SIR_D_1" PARALLEL 16 ;</pre>	
<b>Configuración Reporte G01 – Informe de operación de turnos por círculo y por proceso</b>		
<b>SIR-APP\WEB\WEB-INF\jsp\Seguro\administracion\ADMINREPORTES_REPORTG01_PARAMFORM</b>		
<b>LINEA 25</b>		
	<pre>//AHERRENO 25/10/2010 Si el usuario logueado tiene el rol de administrador nacional //muestre el listado de círculos, si no, que el parametro //del círculo sea el círculo del usuario List rolesUsuario = (List) request.getSession().getAttribute(WebKeys.LISTA_ROLES); if(rolesUsuario!=null){     Iterator itRoles = rolesUsuario.iterator();     while(itRoles.hasNext()){         Rol rol = (Rol)itRoles.next();         if(rol.getRolId().equals(CRol.SIR_ROL_ADMINISTRADOR_NACIONAL)){             usuarioRegistradorNacional = true;             break;         }     } }</pre>	
<b>LINEA 398</b>		
	<pre>&lt;%--AHERRENO 25/10/2010 --%&gt; &lt;%     if(usuarioRegistradorNacional)(&gt;         &lt;tr&gt;             &lt;td width="20"&gt;&lt;img src="&lt;%=request.getContextPath()%&gt;/jsp/images/ind_campotxt.gif" width="20" height="15"&gt;&lt;/td&gt;             &lt;td align="left" width="150"&gt;&lt;div align="right" &gt;C&amp;iacute;rculo Registrado:&lt;/div&gt;&lt;/td&gt;             &lt;td width="200"&gt;                 &lt;%                     try {                         circuloHelper.setId(AWReportes.REPORTE_G01_PARAM_IDCIRCULO);                         circuloHelper.setNombre(AWReportes.REPORTE_G01_PARAM_IDCIRCULO);                         circuloHelper.setCssClase("camposformtext");                         circuloHelper.render(request,out);                     }                     catch(HelperException re){                         out.println("ERROR " + re.getMessage());                     }                 %&gt;             &lt;/td&gt;         &lt;/tr&gt;     &lt;%}else{         Circulo circulo = (Circulo)request.getSession().getAttribute(WebKeys.CIRCULO);     %&gt;         &lt;input type="hidden" name="&lt;%=AWReportes.REPORTE_G01_PARAM_IDCIRCULO%&gt;" id="&lt;%=AWReportes.REPORTE_G01_PARAM_IDCIRCULO%&gt;" value="&lt;%=circulo.getIdCirculo()%&gt;"&gt;     &lt;%}&gt;     &lt;%         Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);     %&gt;         &lt;input type="hidden" name="&lt;%=AWReportes.REPORTE_G01_PARAM_IDUSUARIO%&gt;" id="&lt;%=AWReportes.REPORTE_G01_PARAM_IDUSUARIO %&gt;" value="&lt;%="usuario.getUsername()%&gt;"&gt; </pre>	
<b>SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java</b>		
<b>LINEA 894</b>		
	<pre>//AHERRENO 25/10/2010 //Se agrega parámetro REPORTE_G01_PARAM_IDUSUARIO public static final String REPORTE_G01_PARAM_CMDKEY = "reporte-g-01.jasper"; public static final String REPORTE_G01_NOMBRE = ""; public static final String REPORTE_G01_PARAM_FECHAINI = "P_FECHA_INI"; public static final String REPORTE_G01_PARAM_FECHAFIN = "P_FECHA_FIN"; public static final String REPORTE_G01_PARAM_IDCIRCULO = "P_ID_CIRCULO"; public static final String REPORTE_G01_PARAM_IDUSUARIO = "P_ID_USUARIO";</pre>	
<b>LINEA 8466</b>		
	<pre>//AHERRENO 25/10/2010 //Se agrega validación de parámetros REPORTE_G01_PARAM_IDCIRCULO y REPORTE_G01_PARAM_IDUSUARIO private void do_ReporteG01_Send_Action(HttpServletRequest request) throws ValidacionParametrosException {</pre>	

```

ValidacionParametrosException exception =
    new AdminReportes_InvalidParametersExceptionCollector();

String param_01 = request.getParameter(REPORTE_G01_PARAM_FECHAINI);
String param_02 = request.getParameter(REPORTE_G01_PARAM_FECHAFIN);
String param_03 = request.getParameter(REPORTE_G01_PARAM_IDCIRCULO);
String param_04 = request.getParameter(REPORTE_G01_PARAM_IDUSUARIO);

Log.getInstance().debug(AWReportes.class, "parametro 1: " + param_01);
Log.getInstance().debug(AWReportes.class, "parametro 2: " + param_02);
Log.getInstance().debug(AWReportes.class, "parametro 3: " + param_03);
Log.getInstance().debug(AWReportes.class, "parametro 4: " + param_04);

// apply validators
if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {
    exception.addError("parametro: Fecha Inicio: Debe digitar un valor");
} else {
    String send1 = null;

    Date fecha = null;
    try {
        fecha = DateFormatUtil.parse(param_01);
        send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
        param_01 = send1;
    } catch (Exception e) {
        exception.addError("parametro: Fecha Inicio : La fecha es inválida" + e.getMessage());
    }
}

// apply validators
if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
    exception.addError("parametro: Fecha Fin: Debe digitar un valor");
} else {
    String send1 = null;

    Date fecha = null;
    try {
        fecha = DateFormatUtil.parse(param_02);
        send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
        param_02 = send1;
    } catch (Exception e) {
        exception.addError("parametro: Fecha Fin : La fecha es inválida" + e.getMessage());
    }
}

// apply validators
if( ( null == param_03 )
|| ( "".equalsIgnoreCase( param_03 ) ) ) {
    exception.addError( "parametro: Circulo: Debe digitar un valor" );
}
else {

    if( WebKeys.SIN_SELECCIONAR.equals( param_03 ) ) {
        exception.addError( "parametro: Circulo: Debe digitar un valor" );
    }
}

if( ( null == param_04)
|| ( "".equalsIgnoreCase(param_04) ) ) {
    exception.addError( "parametro: Usuario: Debe digitar un valor" );
}
else {

```

```

if( WebKeys.SIN_SELECCIONAR.equals( param_04 ) ) {
    exception.addError( "parametro: Usuario: Debe digitar un valor" );
}
}
do_ReporteG01_SaveState(request);

// raise -----
// -----
if (exception.getErrores().size() > 0) {
    throw exception;
}

String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY, REPORTE_G01_PARAM_FECHAINI,
    REPORTE_G01_PARAM_FECHAFIN, REPORTE_G01_PARAM_IDCIRCULO,
    REPORTE_G01_PARAM_IDUSUARIO
};

String[] values = new String[]{
    REPORTE_G01_PARAM_CMDKEY, param_01, param_02, param_03, param_04
};

String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_ReporteG01_RemoveState(request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

LINEA 8584
//AHERRENO 25/10/2010
//Se agrega parámetro REPORTE_G01_PARAM_IDUSUARIO
private void do_ReporteG01_SaveState(HttpServletRequest request) {

    HttpSession session;

    String[] itemIds = new String[]{
        REPORTE_G01_PARAM_FECHAINI, REPORTE_G01_PARAM_FECHAFIN,
        REPORTE_G01_PARAM_IDCIRCULO, REPORTE_G01_PARAM_IDUSUARIO
    };

    session = request.getSession();

    save_PageltemsState(itemIds, request, session);

} // :do_ReporteG01_SaveState
LINEA 8600
//AHERRENO 25/10/2010
//Se agrega parámetro REPORTE_G01_PARAM_IDUSUARIO
private void do_ReporteG01_RemoveState(HttpServletRequest request) {

    HttpSession session;

    String[] itemIds = new String[]{
        REPORTE_G01_PARAM_FECHAINI, REPORTE_G01_PARAM_FECHAFIN,
        REPORTE_G01_PARAM_IDCIRCULO, REPORTE_G01_PARAM_IDUSUARIO
    };

    session = request.getSession();

    delete_PageltemsState(itemIds, request, session);

} // :do_ReporteG01_RemoveState
JASPER - reporte-G01.jasper
Se realizo cambio del jasper.
SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDORreportesJasperDAO.java
LINEA 277
//AHERRENO 25/10/2010
//Se agrega parámetro REPORTE_G01_PARAM_IDCIRCULO
//Reporte G01
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_G01_PARAM_IDCIRCULO);

```

```

listaParametros.add(AWReportes.REPORTE_G01_PARAM_FECHAINI);
listaParametros.add(AWReportes.REPORTE_G01.PARAM_FECHAFIN);
mapaReportes.put(AWReportes.REPORTE_G01.PARAM_CMDKEY, new Object[]{
    "SELECT sec_reporte_G01.NEXTVAL FROM dual",
    "sp_reporte_G01(?, ?, ?, ?)", listaParametros,
});

```

## 106 Reporte G03 - Informe de Turnos en Registro por Fases

### **REQUERIMIENTO**

Reporte G-03 Informe de Turnos de Registro Por Fase. Se requiere que para este reporte se listen los turnos que se encuentran en cada una de las fases del proceso de Registro agrupado por usuario.

### **PARAMETROS DE CONSULTA:**

El parámetro fecha inicial se reemplazara por el usuario el cual debe permitir seleccionar uno o todos los del círculo.

El reporte debe mostrar el nombre y código de la Oficina, fecha y hora que se genera el reporte, debe ir paginado, debe mostrar las columnas con títulos así:

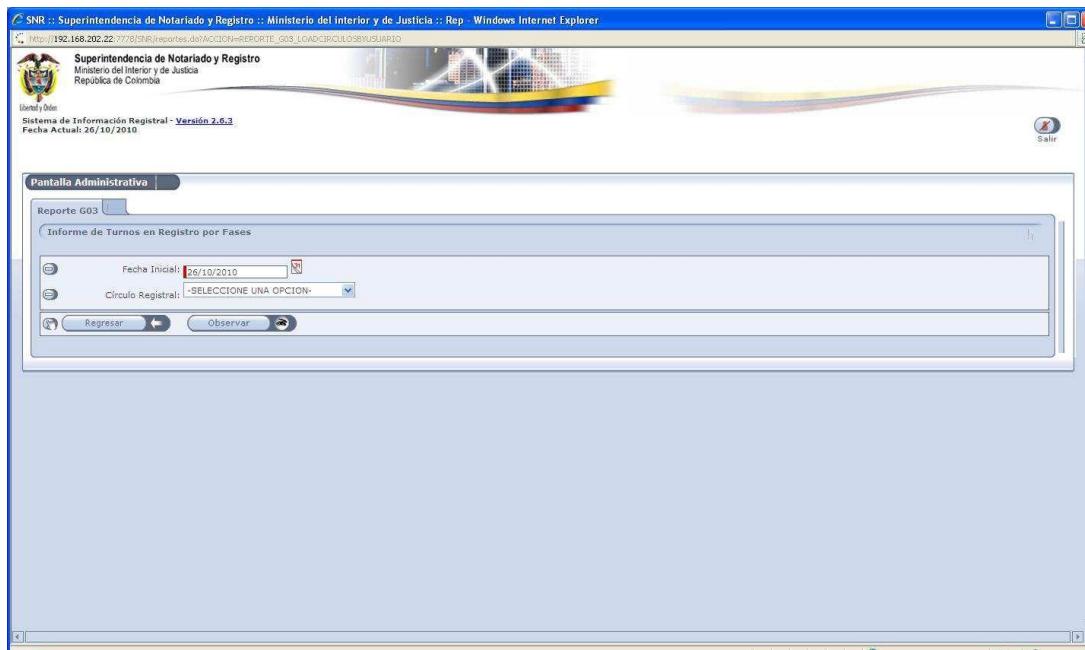
Confrontación, reparto, calificación, testamentos, mesa de control, firma de registrador, certificados asociados en las cuales se debe mostrar el número del turno relacionado a esa fase.

La información se debe mostrar agrupada por Usuario.

Se debe totalizar por fases y al final del reporte debe haber un resumen general por fases de todos los usuarios. Ya sea anulados o habilitados, actualmente presenta error incrementando o disminuyendo el valor de los turnos mencionados.

### **DESARROLLO**

#### **Vista Anterior de Pantallas administrativas:**



#### **Análisis de Requerimiento:**

Para las estaciones compartidas se van a desplegar los mismos turnos a los usuarios que

	<p>tienen esta estación ejm:</p> <p>X-CONRONTADOR : JOSE.GARCIA y HUGO.MANTILLA.</p> <p>Script Bases de Datos: SIR.SP_REPORTE_G03 Se modiflico el archivo:[<b>reporte-g-03.jrxml</b>]</p>	
<b>REPORTE : SIR.SP_REPORTE_G03</b>		
<p>Código Nuevo SIR.SP_REPORTE_G03</p> <pre> create or replace PROCEDURE /*********************************/ /* GCABRERA 15/10/2010 Reporte G03: Informe de Turnos en Registro por Fases*/ /*********************************/ /*********************************/ SP_REPORTE_G03(P_CIRCULO VARCHAR2, P_ID_USUARIO VARCHAR2, SEC_PROCESO VARCHAR2) IS --CURSOR PARA OBTENER TODOS LOS USUARIOS CON ROLES DEL PROCESO DE REGISTRO. CURSOR C_ALLUSER IS SELECT distinct u.id_usuario                       FROM SIR.SIR_OP_USUARIO U INNER JOIN SIR.SIR_REL_USUARIO_CIRCULO UC ON U.ID_USUARIO=UC.ID_USUARIO                       INNER JOIN auriga.cmn_rel_usu_rol_est r on u.srio_nombre_de_usuario= r.usuario_id                       WHERE UC.ID_CIRCULO = P_CIRCULO                       AND U.SRIO_ACTIVO = 1                       and r.rol_id in (select ROL from sir_op_procesos_v                                       where nombre_proceso = 'REGISTRO'                                       AND ID_FADE IN (select distinct id_fase_inicial from sir.wf_registro   where id_proceso = 6)); D_ALLUSER C_ALLUSER%ROWTYPE; BEGIN IF P_ID_USUARIO &lt;&gt; 'TODOS' THEN --CONSULTA UTILIZANDO LA TABLA SIR.SIR_NE_TURNO_EJECUCION BEGIN     INSERT INTO SIR.SIR_OP_REPORTE_G03(SEC_PROCESO, TURNO,FASE,USUARIO)     SELECT SEC_PROCESO, tu.trno_id_workflow TURNO, TE.trej_fase,     (SELECT srio_nombre_de_usuario FROM SIR_OP_USUARIO     WHERE id_usuario = P_ID_USUARIO) USUARIO     FROM SIR.SIR_OP_TURNO TU INNER JOIN SIR.SIR_OP_TURNO_EJECUCION TE ON     TU.trno_id_workflow = te.trej_item_key     WHERE tu.id_circulo = P_CIRCULO     AND tu.id_proceso = '6'     AND TU.trno_fecha_fin IS NULL     AND TU.TRNO_DESCRIPCION IS NULL     AND te.trej_estacion IN ( select DISTINCT estacion_id                                 from sir.op_usuario u inner join auriga.cmn_rel_usu_rol_est r on u.srio_nombre_de_usuario= r.usuario_id                                 where u.id_usuario = P_ID_USUARIO                                 and r.rol_id in (select ROL from sir_op_procesos_v   where nombre_proceso = 'REGISTRO'   AND ID_FADE IN (select distinct id_fase_inicial from sir.wf_registro   where id_proceso = 6)));     COMMIT; EXCEPTION     WHEN OTHERS THEN         ROLLBACK;         RAISE; END; ELSE --ABRIMOS EL CURSOR DE TODOS LOS USUARIOS DEL CIRCULO OPEN C_ALLUSER; LOOP FETCH C_ALLUSER INTO D_ALLUSER;     EXIT WHEN C_ALLUSER%NOTFOUND;     BEGIN         INSERT INTO SIR.SIR_OP_REPORTE_G03(SEC_PROCESO, TURNO,FASE,USUARIO)         SELECT SEC_PROCESO, tu.trno_id_workflow TURNO, TE.trej_fase,         (SELECT srio_nombre_de_usuario FROM SIR_OP_USUARIO         WHERE id_usuario = D_ALLUSER.ID_USUARIO) USUARIO         FROM SIR.SIR_OP_TURNO TU INNER JOIN SIR.SIR_OP_TURNO_EJECUCION TE ON     </pre>		

```

TU.trno_id_workflow = te.trej_item_key
WHERE tu.id_circulo = P_CIRCULO
AND tu.id_proceso = '6'
AND TU.trno_fecha_fin IS NULL
AND TU.TRNO_DESCRIPCION IS NULL
AND te.trej_estacion IN ( select DISTINCT estacion_id
                           from sir.sir_op_usuario u inner join auriga.cmn_rel_usu_rol_est r on
                           u.srio_nombre_de_usuario= r.usuario_id
                           where u.id_usuario = D_ALLUSER.ID_USUARIO
                           and r.rol_id in (select ROL from sir.op_procesos_v
                                             where nombre_proceso = 'REGISTRO'
                                             AND ID_FASE IN (select distinct id_fase_inicial from sir.wf_registro
                                                               where id_proceso = 6)));
                           COMMIT;
EXCEPTION
WHEN OTHERS THEN
    ROLLBACK;
    RAISE;
END;
END LOOP;
CLOSE C_ALLUSER;
END IF;
END;

```

## Configuración Reporte G03 – Informe de Turnos de Registro por Fases

SIR-APPweb\WEB-INF\jspSeguro\administracion\ADMINREPORTES\_REPORTG03\_PARAMFORM

```

LINEA 314
<tr>
<td width="20">
</td>
<td align="left" width="150"><div align="right" >C&iacute;rculo Registral:</div></td>
<td width="200" align="left">
<% try {
    circuloHelper.setNombre(AWReportes.REPORTE_G03_PARAM_IDCIRCULO);
    circuloHelper.setCssClase("camposformtext");
    circuloHelper.setId(AWReportes.REPORTE_G03_PARAM_IDCIRCULO);

    circuloHelper.setFuncion("onChange='cambiarAccionAndSend(\""+AWReportes.REPORTE_G03_LOADUSUARIOSBYCIRCULO+"")'");
    circuloHelper.render(request,out);
}

LINEA 366
<tr>
<td width="20"></td>
<td align="left" width="150"><div align="right" >Usuario:</div></td>
<td width="200" align="left">

<%if(request.getSession().getAttribute(AWReportes.USUARIOS_CONSULTAR_POR_CIRCULO)!=null){%
    %
    <% try {
        ListaElementoHelper listaHelper = new ListaElementoHelper();

        listaHelper.setTipos((List)request.getSession().getAttribute(AWReportes.USUARIOS_CONSULTAR_POR_CIRCULO));
        listaHelper.setNombre(AWReportes.REPORTE_G03.PARAM_IDUSUARIO);
        listaHelper.setId(AWReportes.REPORTE_G03.PARAM_IDUSUARIO);
        listaHelper.setCssClase("camposformtext");
        listaHelper.render(request,out);
    }catch(HelperException re){
        out.println("ERROR " + re.getMessage());
    }
    %
    <%>
    <%}else{ %>
        No hay Datos
    <%} %>
</td>
</tr>
<%
    Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
%>
<input type="hidden" name="<%={AWReportes.REPORTE_G01_PARAM_USUARIOLOG%}">
```

	<pre> id=&lt;%=AWReportes.REPORTE_G01_PARAM_USUARIOLOG%&gt; value=&lt;%=usuario.getUsername()%&gt;&gt; </pre>	
<b>SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java</b>		
<b>LINEA 419</b>	<pre> // AHERRENO 28/10/2010 // Reporte G03 public static final String REPORTE_G03_LOADCIRCULOSBYUSUARIO = "REPORTE_G03_LOADCIRCULOSBYUSUARIO"; public static final String REPORTE_G03_LOADUSUARIOSBYCIRCULO = "REPORTE_G03_LOADUSUARIOSBYCIRCULO"; </pre>	
<b>LINEA 912</b>	<pre> // AHERRENO 28/10/2010 // Reporte G03 public static final String REPORTE_G03_PARAM_CMDKEY = "reporte-g-03.jasper", public static final String REPORTE_G03_NOMBRE = ""; public static final String REPORTE_G03_PARAM_IDCIRCULO = "P_ID_CIRCULO"; public static final String REPORTE_G03_PARAM_IDUSUARIO = "P_ID_USUARIO"; public static final String REPORTE_G03_PARAM_USUARIOLOG = "P_USUARIO_LOG"; </pre>	
<b>LINEA 1439</b>	<pre> // AHERRENO 28/10/2010 // Reporte G03 if (REPORTE_G03_LOADCIRCULOSBYUSUARIO.equals(accion)) {     return do_ReporteXX_LoadCirculosByUsuario_Action(request); } if (REPORTE_G03_LOADUSUARIOSBYCIRCULO.equals(accion)) {     session.removeAttribute(AWReportes.USUARIOS_CONSULTAR_POR_CIRCULO);     return consultarUsuariosPorCirculo(request, AWReportes.REPORTE_G03_PARAM_IDCIRCULO, EvnReportes.USUARIOS_CONSULTAR_POR_CIRCULO_ROLES); } </pre>	
<b>LINEA 8761</b>	<pre> // AHERRENO 28/10/2010 // Reporte G03 private void do_ReporteG03_Send_Action(HttpServletRequest request) throws ValidacionParametrosException {      ValidacionParametrosException exception =         new AdminReportes_InvalidParametersExceptionCollector();      String param_01 = request.getParameter(REPORTE_G03_PARAM_IDCIRCULO);     String param_02 = request.getParameter(REPORTE_G03_PARAM_IDUSUARIO);     String param_03 = request.getParameter(REPORTE_G03_PARAM_USUARIOLOG);      Log.getInstance().debug(AWReportes.class, "parametro 1: " + param_01);     Log.getInstance().debug(AWReportes.class, "parametro 2: " + param_02);     Log.getInstance().debug(AWReportes.class, "parametro 3: " + param_03);      // apply validators     if ((null == param_01)    ("".equalsIgnoreCase(param_01))) {         exception.addError("parametro: Circulo: Debe digitar un valor");     } else {         if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {             exception.addError("parametro: Circulo: Debe digitar un valor");         }     }      // apply validators     if ((null == param_02)    ("".equalsIgnoreCase(param_02))) {         exception.addError("parametro: Usuario: Debe digitar un valor");     } else {         if (WebKeys.SIN_SELECCIONAR.equals(param_02)) {             exception.addError("parametro: Usuario: Debe digitar un valor");         }     }      if ((null == param_03)    ("".equalsIgnoreCase(param_03))) {         exception.addError("parametro: Usuario Logueado: Debe digitar un valor");     } else {         if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {             exception.addError("parametro: Usuario Logueado: Debe digitar un valor");         }     } } </pre>	

```

        }

        do_ReporteG03_SaveState(request);

        // raise -----
        if (exception.getErrores().size() > 0) {
            throw exception;
        }

        // -----
        String[] params = new String[]{
            REPORTE_XX_PARAM_CMDKEY, REPORTE_G03_PARAM_IDCIRCULO,
            REPORTE_G03_PARAM_IDUSUARIO, REPORTE_G03_PARAM_USUARIOLOG
        };

        String[] values = new String[]{
            REPORTE_G03_PARAM_CMDKEY, param_01, param_02, param_03
        };

        String url = makeUrl(params, values);
        reportsServices_ConfigureParameterUri(request, url);

        do_ReporteG03_RemoveState(request);
        do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
    }

    private void do_ReporteG03_Back_Action(HttpServletRequest request) {

        do_ReporteG03_RemoveState(request);
        do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);

    } // :do_ReporteG03_Back_Action

    private void do_ReporteG03_SaveState(HttpServletRequest request) {

        HttpSession session;

        String[] itemIds = new String[]{
            REPORTE_G03_PARAM_IDCIRCULO, REPORTE_G03_PARAM_IDUSUARIO,
            REPORTE_G03_PARAM_USUARIOLOG
        };

        session = request.getSession();
        save_PageltemsState(itemIds, request, session);

    } // :do_ReporteG03_SaveState

    private void do_ReporteG03_RemoveState(HttpServletRequest request) {

        HttpSession session;

        String[] itemIds = new String[]{
            REPORTE_G03_PARAM_IDCIRCULO, REPORTE_G03_PARAM_IDUSUARIO,
            REPORTE_G03_PARAM_USUARIOLOG
        };

        session = request.getSession();
        session.removeAttribute(AWReportes.USUARIOS_CONSULTAR_POR_CIRCULO);
        delete_PageltemsState(itemIds, request, session);

    } // :do_ReporteG03_RemoveState

LINEA 8867
// AHERRENO 28/10/2010
// Reporte G03
private EvnReportes consultarUsuariosPorCirculo(HttpServletRequest request, String param_pcirculonombre,
    String eventoConsulta) throws AccionWebException {
    HttpSession session = request.getSession();
    ValidacionParametrosException exception = new ValidacionParametrosException();
    String idCirculo = request.getParameter(param_pcirculonombre);
}

```

```

if ((idCirculo == null) || (idCirculo.trim().length() == 0) || idCirculo.equals(WebKeys.SIN_SELECCIONAR)) {
    exception.addError("Debe Seleccionar un Circulo.");
} else {
    session.setAttribute(param_pcirculonombre, idCirculo);
}
if (!exception.getErrores().isEmpty()) {
    throw exception;
}
session.setAttribute(CCirculo.ID_CIRCULO, idCirculo);

Usuario usuario = (Usuario) session.getAttribute(SMARTKeys.USUARIO_EN_SESION);
Circulo circulo = new Circulo();
circulo.setIdCirculo(idCirculo);

EvnReportes evento = new EvnReportes(usuario, eventoConsulta);
evento.setCirculo(circulo);

List roles = new ArrayList();
roles.add(CRol.SIR_ROL_CONFRONTADOR);
roles.add(CRoles.SIR_ROL_JEFE_OPERATIVO);
roles.add(CRol.ROL_ABOGADO);
roles.add(CRol.SIR_ROL_TESTAMENTOS);
roles.add(CRol.SIR_ROL_MESA_CONTROL);
roles.add(CRol.SIR_ROL_REGISTRADOR);

evento.setRoles(roles);
evento.setRol(null);

return evento;
}

```

**LINEA 13468**

```

//AHERRENO 28/10/2010
//Se modifica metodo para validación de roles
for (; local_RolesListIterator.hasNext();) {
    local_RolesListElement = (org.auriga.core.modelo.transferObjects.Rol) local_RolesListIterator.next();
    String rol = local_RolesListElement.getRolId();

    if
(accion.equals(REPORTE_G03_LOADCIRCULOSBYUSUARIO)||accion.equals(REPORTE_G01_LOADCIRCULOS
BYUSUARIO)||accion.equals(REPORTE_161
_LOADCIRCULOSBYUSUARIO)||accion.equals(REPORTE_E03_LOADCIRCULOSBYUSUARIO)) {
        if (rol.equals(CRoles.ADMINISTRADOR_NACIONAL) ) {
            isAdministradorNacional = true;
            break;
        }
    }else if (rol.equals(CRoles.ADMINISTRADOR_NACIONAL) || rol.equals(CRoles.SIR_ROL_GESTION) ||
rol.equals(CRoles.CONSULTA_NACIONAL) || rol.equals(CRoles.REPORTES_NACIONAL)) {
        isAdministradorNacional = true;
        break;
    } // if
} // for

```

**JASPER - reporte-G03.jasper**

Se realizo cambio del jasper.

**JASPER - SUBREPORTE\_G03\_FASE.jasper**

Archivo nuevo.

**SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDOResortesJasperDAO.java**

**LINEA 289**

```

//AHERRENO 26/10/2010
//Se agrega parámetro REPORTE_G01_PARAM_IDCIRCULO
//Reporte G03
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_G03.PARAM_IDCIRCULO);
listaParametros.add(AWReportes.REPORTE_G03.PARAM_IDUSUARIO);
mapaReportes.put(AWReportes.REPORTE_G03.PARAM_CMDKEY, new Object[]{
    "SELECT sec_reporte_G03.NEXTVAL FROM dual",
    "sp_reporte_G03(?, ?, ?)", listaParametros,
    "DELETE FROM sir_op_reporte_G01 WHERE sec_proceso = ?"});

```

**SIR-APP\src\gov\sir\core\ negocio\modelo\ constantes\CRol.java**

**LINEA 65**

```

//AHERRENO 28/10/2010
//Se crean variables para llamado de roles
public static final String SIR_ROL_CONFRONTADOR = "SIR_ROL_CONFRONTADOR";

```

	<pre> public static final String SIR_ROL_TESTAMENTOS = "SIR_ROL_TESTAMENTOS"; public static final String SIR_ROL_CERTIFICADOS_ASOCIADOS_REGISTRO = "SIR_ROL_CERTIFICADOS_ASOCIADOS_REGISTRO"; <b>SIR-APP\src\gov\sir\core\web\navegacion\administracion\ControlNavegacionReportes.java</b> <b>LINEA 253</b> public static final String REPORTE_G03_LOADUSUARIOBYCIRCULO_EVENTRESP_OK = "REPORTE_G03_LOADUSUARIOBYCIRCULO_EVENTRESP_OK"; <b>Mapas.jar</b> org.auriga.smart.web <b>LINEA 806</b> &lt;!-- AHERRENTO 27/10/2010 report-G03 --&gt; &lt;navegacion&gt; &lt;resultado&gt;REPORTE_G03_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK&lt;/resultado&gt; &lt;url-redireccion&gt;admin.reportes.report-g-03.param-form.view&lt;/url-redireccion&gt; &lt;/navegacion&gt;  &lt;navegacion&gt; &lt;resultado&gt;REPORTE_G03_LOADUSUARIOBYCIRCULO_EVENTRESP_OK&lt;/resultado&gt; &lt;url-redireccion&gt;admin.reportes.report-g-03.param-form.view&lt;/url-redireccion&gt; &lt;/navegacion&gt; </pre>	
<b>119 Reporte Estadistica Correciones</b>		
<b>REQUERIMIENTO</b>		
Se debe crear un reporte para el proceso de correcciones con las siguientes especificaciones:		
<p>Criterio de selección:</p> <p>Rango de fechas: fecha inicial y fecha final</p> <p>Círculo Registral: por defecto debe traer el círculo desde donde se está generando el reporte.</p>		
Al generar el reporte debe tener el encabezado general para todos los reportes sir, nombre de la oficina de registro y el circulo (numero), titulo del reporte: REPORTE ESTADÍSTICAS DE CORRECCIONES y rango de fechas (el seleccionado), Fecha de la impresión y hora.		
Debe tener las siguientes ITEMS y contabilizar los turnos de correcciones así:		
<p>CORRECCIONES RADICADAS(Cantidad de turnos en el rango de fecha seleccionado)</p> <p>CORRECCIONES APROBADAS (Cantidad de turnos en el rango de fecha seleccionado y que estén en estado desanotado).</p> <p>CORRECCIONES NEGADAS (Cantidad de turnos en el rango de fecha seleccionado y que estén en estado desanotado )</p> <p>CORRECCIONES EN TRAMITE (Cantidad de turnos en el rango de fecha seleccionado y para este ITEM se deben indicar la cantidad de turnos de correcciones en las diferentes fases del proceso (revisión y análisis, Ejecutar corrección, revisar y aprobar, Mesa de control correcciones).</p>		
<b>DESARROLLO</b>		
El total de correcciones no va a coincidir ya que las correcciones que se encuentran en otras fases no se van a reflejar.		
Este nuevo reporte se debe asignar a los roles Responsable de Correcciones y administrador Regional.		

## Configuración de la pantalla "Reporte Folios Masivos"

```
INSERT INTO SIR_OP_PANTALLA_ADMINISTRATIVA (id_pantalla_administrativa,  
id_tipo_pantalla, ptdm_nombre, ptdm_vista, ptdm_pagina)  
VALUES (1115, 13, 'Reporte E03: Estadística Correcciones', 'admin.reportes.report-  
E03.param-form.view', 'PG_REPORTES');
```

Asignación de la pantalla "Reporte E03: Estadística Correcciones" al Rol SIR\_ROL\_ADMINISTRADOR REGIONAL

```
INSERT INTO SIR_OP_ROL_PANTALLA (id_rol,id_pantalla_administrativa)
VALUES ('SIR_ROL_ADMINISTRADOR_REGIONAL',1115);
```

## Asignación de la pantalla "Reporte E03: Estadística Correcciones" al Rol SIR\_ROL\_RESPONSABLE\_CORRECCIONES

```
INSERT INTO SIR_OP_ROL_PANTALLA (id_rol,id_pantalla_administrativa)
VALUES ('SIR_ROL_RESPONSABLE_CORRECCIONES',1115);
```

Se creó la secuencia SEQ\_Reporte\_E03

Configuración Reporte E03 – Estadística Correcciones

## SIR-APP\WEB\WEB-INF\jspSeguro\ADMINREPORTES\_Reporte03\_ParamForm

Archivo Nuevo.

SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java

LINEA 282

// AHERRENO 02/08/2010  
// Reporte E03

```
public static final String REPORTE_E03_SEND_ACTION = "REPORTE_E03_SEND_ACTION";
public static final String REPORTE_E03_BACK_ACTION = "REPORTE_E03_BACK_ACTION";
```

LINEA 452

// AHERRENO 02/08/2010

```
// Reporte E03  
public static final String REPORTE_E03_LOADCIRCULOSBYUSUARIO =  
REPORTE_E03_LOADCIRCULOSBYUSUARIO";
```

LINEA 984

// AHERRENO 02/08/2010

```
// Declaración de las Variables para el reporte E03
public static final String REPORTE_E03_PARAM_CMDKEY = "reporte-E03.jasper";
public static final String REPORTE_E03_NOMBRE = "";
public static final String REPORTE_E03_PARAM_PFECHAINI = "V_FECHAINI";
public static final String REPORTE_E03_PARAM_PFECHAFIN = "V_FECHAFIN";
public static final String REPORTE_E03_PARAM_PCIRCULONOMBRE = "V_CIRCU";
public static final String REPORTE_E03_PARAM_USUARIOLOG = "P_USUARIO_LG";
```

LINEA 2252

```
//AHERRONO 02/08/2010  
//Reporte E03  
if (REPORTE_E03_LOAD  
    return do_ReporteXX_L  
}
```

LINFA 984

// AHERRENO 02/08/2010  
// Reporte E03  
if (REPORTE\_E03.SEND\_ACTION equals(accion)) {

```

        do_ReporteE03_Send_Action(request);
    }
    if (REPORTE_E03_BACK_ACTION.equals(accion)) {
        do_ReporteE03_Back_Action(request);
    }

LINEA 984
//AHERRENO 02/08/2010
//Reporte E03
private void do_ReporteE03_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_E03_PARAM_PFECHAINI);
    String param_02 = request.getParameter(REPORTE_E03_PARAM_PFECHAFIN);
    String param_03 = request.getParameter(REPORTE_E03_PARAM_PCIRCULONOMBRE);
    String param_04 = request.getParameter(REPORTE_E03_PARAM_USUARIOLOG);

    // apply validators
    if ((null == param_01) || ("").equalsIgnoreCase(param_01)) {
        exception.addError("parametro: Fecha Inicio Radicación: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio Radicación: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_02) || ("").equalsIgnoreCase(param_02)) {
        exception.addError("parametro: Fecha Fin Radicación: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_02);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_02 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Fin Radicación: La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_03) || ("").equalsIgnoreCase(param_03)) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
            exception.addError("parametro: Circulo: Debe digitar un valor");
        }
    }

    // apply validators
    if ((null == param_04) || ("").equalsIgnoreCase(param_04)) {
        exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
            exception.addError("parametro: Usuario Logueado: Debe ingresar un valor");
        }
    }

    do_ReporteE03_SaveState(request);

    // raise -----
    if (exception.getErrores().size() > 0) {
        throw exception;
    }
}

```

```

// -----
String[] params = new String[]{
    REPORTE_E03_PARAM_CMDKEY, REPORTE_E03_PARAM_PFECHAINI,
    REPORTE_E03_PARAM_PFECHAFIN, REPORTE_E03_PARAM_PCIRCULONOMBRE,
    REPORTE_E03_PARAM_USUARIOLOG    };
String[] values = new String[]{
    REPORTE_E03_PARAM_CMDKEY, param_01, param_02, param_03 , param_04
};

String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_ReporteE03_RemoveState(request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

private void do_ReporteE03_Back_Action(HttpServletRequest request) {

do_ReporteE03_RemoveState(request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);

} // :do_ReporteE03_Back_Action

private void do_ReporteE03_SaveState(HttpServletRequest request) {
HttpSession session;
String[] itemIds = new String[]{
    REPORTE_E03_PARAM_PFECHAINI, REPORTE_E03_PARAM_PFECHAFIN,
    REPORTE_E03_PARAM_PCIRCULONOMBRE , REPORTE_E03_PARAM_USUARIOLOG
};

session = request.getSession();
save_PageItemsState(itemIds, request, session);
} // :do_ReporteE03_SaveState

private void do_ReporteE03_RemoveState(HttpServletRequest request) {

HttpSession session;

String[] itemIds = new String[]{
    REPORTE_E03_PARAM_PFECHAINI, REPORTE_E03_PARAM_PFECHAFIN,
    REPORTE_E03_PARAM_PCIRCULONOMBRE , REPORTE_E03_PARAM_USUARIOLOG
};

session = request.getSession();
delete_PageItemsState(itemIds, request, session);

} // :do_ReporteE03_RemoveState

```

#### JASPER - reporte-E03.jasper

Archivo Nuevo.

#### SIR-APP\src\gov\lsir\core\web\navegacion\administracion\ControlNavegacionReportes.java

```

//AHERRENO 02/08/2010
//Reporte E03
public static final String REPORTE_E03_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK =
"REPORTE_E03_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK";

//Reporte E03
listaReportesJasper.add(AWReportes.REPORTE_E03_SEND_ACTION);

//AHERRENO 02/08/2010
// -- navegacion sobre pagina
if( AWReportes.REPORTE_E03_LOADCIRCULOSBYUSUARIO.equals( accion ) ){
    return REPORTE_E03_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK;
}

//AHERRENO 02/08/2010
//Reporte E03
if( AWReportes.REPORTE_E03.BACK_ACTION.equals( accion ) ) {
    return REPORTE_XX_BACK_EVENTRESP_OK;
}

```

#### SIR-APP\src\gov\lsir\hermod\dao\impl\jdogenie\JDOResportesJasperDAO.java

```

//AHERRENO 02/08/2010
//Reporte E03
//Se obtienen los parámetros para enviar al procedimiento
listaParametros = new ArrayList();

```

	<pre> listaParametros.add(AWReportes.REPORTE_E03_PARAM_PFECHAINI); listaParametros.add(AWReportes.REPORTE_E03_PARAM_PFECHAFIN); listaParametros.add(AWReportes.REPORTE_E03_PARAM_PCIRCULONOMBRE); mapaReportes.put(AWReportes.REPORTE_E03_PARAM_CMDKEY, new Object[]{     "SELECT seq_reporte_E03.NEXTVAL FROM dual",     "sp_reporte_E03(?, ?, ?, ?)", listaParametros,     "DELETE FROM sir_op_reporte_E03 WHERE sec_proceso = "}); </pre>
	<b>SIR-APP\src\org.auriga.smart.vistas.xml</b>
	<pre> &lt;!-- AHERRENO 02/08/2010 report-E03 --&gt;  &lt;vista&gt; &lt;autenticacion&gt;true&lt;/autenticacion&gt; &lt;excepcion&gt;gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector&lt;/excepcion&gt; &lt;excepcion&gt;gov.sir.core.web.acciones.excepciones.ValidacionParametrosException&lt;/excepcion&gt; &lt;nombre&gt;admin.reportes.report-E03.param-form.view&lt;/nombre&gt; &lt;parametro&gt;&lt;llave&gt;error&lt;/llave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/ERROR_MULTIPLE.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;parametro&gt;&lt;llave&gt;head&lt;/llave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/HEADER_LOGOUT.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;parametro&gt;&lt;llave&gt;body&lt;/llave&gt;&lt;valor&gt;/WEB- INF/jspSeguro/administracion/ADMINREPORTES_REPORTE03_PARAMFORM.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;parametro&gt;&lt;llave&gt;bottom&lt;/llave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/FOOTER_DEFAULT.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;/vista&gt; </pre>
	<b>SIR-APP\src\org.auriga.smart.web.xml</b>
	<pre> &lt;!-- AHERRENO 02/08/2010 report-E03 --&gt; &lt;navegacion&gt; &lt;resultado&gt;REPORTE_E03_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK&lt;/resultado&gt; &lt;url-redireccion&gt;admin.reportes.report-E03.param-form.view&lt;/url-redireccion&gt; &lt;/navegacion&gt; </pre>
	<b>mapas.jar</b>
	<p>Se actualizan los archivos:</p> <p>org.auriga.smart.vistas.xml org.auriga.smart.web.xml</p>
	<b>128 Problema reportes</b>
	<p><b>REQUERIMIENTO</b></p> <p>Problema Reportes. Se comunica Margarita Mejía Escobar informando que cuando se están generando los reportes:</p> <p>Reporte 11: Estadísticas por actos de registro (día xx al día yy) y los reportes de todo el mes de septiembre de Reporte 09: Diario radicador de documentos del día existe una diferencia de \$2.000 pesos, al verificar los reportes Reporte 09: Diario radicador de documentos del día el problema radica en el día 01/09/09 generando en este un valor total de \$256.500, pero cuando se genera el reporte 11 de ese día únicamente arroja un valor total de \$254.000.</p> <p>La funcionaria informa que el problema, comparado con los recibos físicos se encuentra en el reporte 11 del día 01/09/09.</p> <p>En el acto de venta se reporta un valor de \$97.000 comparado con el reporte 9 que es de \$105.000.</p> <p>informando que el valor correcto que debe aparecer en el reporte 11 de ese día es \$105.000, para el acto aclaración el valor que aparece en el reporte 11 de ese día es \$76.000, comparado con los recibos físicos el valor correcto que debe aparecer es de \$70.000</p> <p>Margarita Mejía informa que debe presentar reportes a la SNR en el día de hoy. Andes. Id Glpi 12757.</p> <p><b>DESARROLLO</b></p>

**Correo ETB**

Se comunica Margarita Mejía Escobar informando que cuando se están generando los reportes:

Reporte 11: Estadísticas por actos de registro (día xx al día yy) y los reportes de todo el mes de septiembre de Reporte 09: Diario Radicador de documentos del día existe una diferencia de \$2.000 pesos, al verificar los reportes Reporte 09: Diario Radicador de documentos del día el problema radica en el día 01/09/09 generando en este un valor total de \$256.500, pero cuando se genera el reporte 11 de ese día únicamente arroja un valor total de \$254.000.

La funcionaria informa que el problema, comparado con los recibos físicos se encuentra en el reporte 11 del día 01/09/09.

En el acto de venta se reporta un valor de \$97.000 comparado con el reporte 9 que es de \$105.000.

Informando que el valor correcto que debe aparecer en el reporte 11 de ese día es \$105.000, para el acto aclaración el valor que aparece en el reporte 11 de ese día es \$76.000, comparado con los recibos físicos el valor correcto que debe aparecer es de \$70.000

Margarita Mejía informa que debe presentar reportes a la SNR en el día de hoy. Andes.

En el análisis del caso se detectó que el reporte 09: Diario Radicador de Documentos del día; presenta fallas en el procedimiento “SP\_REPORTE\_09” ya que está tomando el primer acto que encuentra de una solicitud y no tiene en cuenta si se han registrado más actos, lo cual se presta para confusión en la interpretación del reporte pues al tomar el primer acto le relaciona el valor de la liquidación por derechos del turno mas no de los actos. Se sugiere que para este reporte se muestre el detalle de cada acto relacionado al turno o la otra sugerencia es que no se tenga en cuenta la columna acto en este reporte y se le indique a los usuarios que genere el reporte 11: Estadísticas por actos de Registro el cual tiene como objetivo desplegar un total por acto en un rango de fechas. Con una pronta revisión de este caso la solución se podría mostrar en el próximo despliegue.

**Correo Recibido el 19 de Agosto, enviado por los funcionarios Yamile Moreno y José Miguel Galvis.**

**OBSERVACIONES SNR:** Se requiere que el reporte 09 DIARIO RADICADOR DE DOCUMENTOS DEL DIA refleje todos los actos agregados en el proceso de radicación de cada turno de registro de documentos y en la columna de derechos se refleje el valor total de los derechos liquidados del turno, como se observa actualmente en el imprimible; para mejor comprensión de la información del reporte el título correspondiente a la columna “Derechos” se debe ajustar así: “Total Derechos”.

**Análisis y Desarrollo del requerimiento:**

Cuando se genera el reporte 11 de ese día únicamente arroja un valor total de \$254.000. La funcionaria informa que el problema, comparado con los recibos físicos se encuentra en el **reporte 11** del día 01/09/2009. En el acto de venta se reporta un valor de **\$97.000** comparado con el reporte 9 que es de \$105.000 informando que el valor correcto que debe aparecer en el reporte 11 de ese día es **\$105.000**

Cuando se genera el reporte 11 de ese día únicamente arroja un valor total de \$254.000. La funcionaria informa que el problema, comparado con los recibos físicos se encuentra en el **reporte 11** del día 01/09/2009. En el acto de venta se reporta un valor de **\$97.000** comparado con el reporte 9 que es de \$105.000 informando que el valor correcto que debe aparecer en el reporte 11 de ese día es **\$105.000**

El reporte 11 tiene en cuenta la liquidación de turnos anteriores, el valor de \$97.000 sale teniendo en cuenta lo siguiente:

Turno	Acto	Valor
2009-004-6-1891	Venta	16.500
2007-004-6-363	Venta	24.500 (Turno Manual)

DIFERENCIA -8.000

Sumado con el valor de liquidación del turno 2010-004-6-1892 el cual tiene una liquidación por venta de 105.000

- - - - Da un total de 105.000 – 8.000 = 97.000

El error radica en la liquidación del turno manual 2007-004-6-363

Teniendo en cuenta la observación enviada por SNR, se realizó el siguiente cambio:

Reporte 09: Diario radicador de Documentos del día

Modificación archivos

Se modifica nombre de Columna Derechos a Total Derechos

### Reporte-09.jasper

LINEA 504

```
// AHERRENTO 02/11/2010
// Reporte 09 - Se agrega parametro P_USUARIO_LOG
public static final String REPORTE_09_PARAM_CMDKEY = "reporte-09.jasper";
public static final String REPORTE_09_NOMBRE = "";
public static final String REPORTE_09_PARAM_PCIRCULONOMBRE = "P_CIRCULO_NOMBRE";
```

```

public static final String REPORTE_09_PARAM_PNUMPAGINA = "P_NUM_PAGINA";
public static final String REPORTE_09_PARAM_PTURNOFECHA = "P_TURNO_FECHA";
public static final String REPORTE_09_PARAM_USUARIOLOG = "P_USUARIO_LOG";

LINEA 519
// AHERRENO 02/11/2010
// Reporte 11 - Se agrega parametro P_USUARIO_LOG
public static final String REPORTE_11_PARAM_CMDKEY = "reporte-11.jasper";
public static final String REPORTE_11_NOMBRE = "";
public static final String REPORTE_11_PARAM_PFECHAINI = "V_FECHAINI";
public static final String REPORTE_11_PARAM_PFECHAFIN = "V_FECHAFIN";
public static final String REPORTE_11_PARAM_PCIRCULONOMBRE = "V_CIRCULO";
public static final String REPORTE_11_PARAM_USUARIOLOG = "P_USUARIO_LOG";

LINEA 3330
// AHERRENO 02/11/2010
// Reporte 09 - Se obtiene parametro REPORTE_09_PARAM_USUARIOLOG
private void do_Reporte09_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_09_PARAM_PTURNOFECHA);

    String param_02 = request.getParameter(REPORTE_09_PARAM_PCIRCULONOMBRE);

    String param_03 = request.getParameter(REPORTE_09_PARAM_PNUMPAGINA);

    String param_04 = request.getParameter(REPORTE_09_PARAM_USUARIOLOG);

    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {
        exception.addError("parametro: Fecha Turno: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Desanotacion : La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_02)) {
            exception.addError("parametro: Circulo: Debe digitar un valor");
        }
    }

    // apply validators
    if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {
        exception.addError("parametro: No. de Página: Debe digitar un valor");
    } else {
        for (int i = 0; i < param_03.length(); i++) {
            if (!Character.isDigit(param_03.charAt(i))) {
                exception.addError("parametro: No. de Página: Debe ser un valor numerico");
                i = param_03.length();
            }
        }
    }
}

```

```

// apply validators
if ((null == param_04) || ("").equalsIgnoreCase(param_04)) {
    exception.addError("parametro: Usuario Logueado: Debe digitar un valor");

} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
        exception.addError("parametro: Usuario Logueado: Debe digitar un valor");
    }
}

do_Reporte09_SaveState(request);

// raise -----
// -----
if (exception.getErrors().size() > 0) {
    throw exception;
}

// -----
String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY, REPORTE_09_PARAM_PCIRCULONOMBRE,
    REPORTE_09_PARAM_PNUMPAGINA, REPORTE_09_PARAM_PTURNOFECHA,
    REPORTE_09_PARAM_USUARIOLOG
};

String[] values = new String[]{
    REPORTE_09_PARAM_CMDKEY, param_02, param_03, param_01, param_04
};

String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_Reporte09_RemoveState(request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

private void do_Reporte09_Back_Action(HttpServletRequest request) {

    do_Reporte09_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);

} // :do_Reporte09_Back_Action

private void do_Reporte09_SaveState(HttpServletRequest request) {

    HttpSession session;

    String[] itemIds = new String[]{
        REPORTE_09_PARAM_PCIRCULONOMBRE, REPORTE_09_PARAM_PNUMPAGINA,
        REPORTE_09_PARAM_PTURNOFECHA, REPORTE_09_PARAM_USUARIOLOG
    };

    session = request.getSession();
    save_PageItemsState(itemIds, request, session);

} // :do_Reporte09_SaveState

private void do_Reporte09_RemoveState(HttpServletRequest request) {

    HttpSession session;

    String[] itemIds = new String[]{
        REPORTE_09_PARAM_PCIRCULONOMBRE, REPORTE_09_PARAM_PNUMPAGINA,
        REPORTE_09_PARAM_PTURNOFECHA, REPORTE_09_PARAM_USUARIOLOG
    };

    session = request.getSession();
    delete_PageItemsState(itemIds, request, session);
}

```

```

} // :do_Reporte09_RemoveState

LINEA 330
// AHERRENO 02/11/2010
// Reporte 11 - Se obtiene parametro REPORTE_11_PARAM_USUARIOLOG
private void do_Reporte11_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_11_PARAM_PFECHAINI);

    String param_02 = request.getParameter(REPORTE_11_PARAM_PFECHAFIN);

    String param_03 = request.getParameter(REPORTE_11_PARAM_PCIRCULONOMBRE);

    String param_04 = request.getParameter(REPORTE_11_PARAM_USUARIOLOG);

    // apply validators
    if ((null == param_01) || ("").equalsIgnoreCase(param_01)) {
        exception.addError("parametro: Fecha Inicio: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio : La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_02) || ("").equalsIgnoreCase(param_02)) {
        exception.addError("parametro: Fecha Fin: Debe digitar un valor");
    } else {
        String send1 = null;

        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_02);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);

            param_02 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Fin : La fecha es inválida" + e.getMessage());
        }
    }

    // apply validators
    if ((null == param_03) || ("").equalsIgnoreCase(param_03)) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
            exception.addError("parametro: Circulo: Debe digitar un valor");
        }
    }

    // apply validators
    if ((null == param_04) || ("").equalsIgnoreCase(param_04)) {
        exception.addError("parametro: Usuario Logueado: Debe digitar un valor");
    }
}

```

```

} else {

    if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
        exception.addError("parametro: Usuario Logueado: Debe digitar un valor");
    }

}

do_Reporte11_SaveState(request);

// raise -----
if (exception.getErros().size() > 0) {
    throw exception;
}

// -----
String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY, REPORTE_11_PARAM_PFECHAINI,
    REPORTE_11_PARAM_PFECHAFIN, REPORTE_11_PARAM_PCIRCULONOMBRE,
    REPORTE_11_PARAM_USUARIOLOG
};

String[] values = new String[]{
    REPORTE_11_PARAM_CMDKEY, param_01, param_02, param_03, param_04
};

String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_Reporte11_RemoveState(request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

private void do_Reporte11_Back_Action(HttpServletRequest request) {

    do_Reporte11_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);

} // :do_Reporte11_Back_Action

private void do_Reporte11_SaveState(HttpServletRequest request) {

    HttpSession session;

    String[] itemIds = new String[]{
        REPORTE_11_PARAM_PFECHAINI, REPORTE_11_PARAM_PFECHAFIN,
        REPORTE_11_PARAM_PCIRCULONOMBRE, REPORTE_11_PARAM_USUARIOLOG
    };

    session = request.getSession();
    save_PageItemsState(itemIds, request, session);

} // :do_Reporte11_SaveState

private void do_Reporte11_RemoveState(HttpServletRequest request) {

    HttpSession session;

    String[] itemIds = new String[]{
        REPORTE_11_PARAM_PFECHAINI, REPORTE_11_PARAM_PFECHAFIN,
        REPORTE_11_PARAM_PCIRCULONOMBRE, REPORTE_11_PARAM_USUARIOLOG
    };

    session = request.getSession();
    delete_PageItemsState(itemIds, request, session);

} // :do_Reporte11_RemoveState

```

SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES\_REPORT09\_PARAMFORM.jsp

**LINEA 412**

```
<%-- AHERRRENO 02/11/2010 --%>
<%
    Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
%>
<input type="hidden" name="<%="AWReportes.REPORTE_09_PARAM_USUARIOLOG%">
id="<%="AWReportes.REPORTE_09_PARAM_USUARIOLOG%">
value="<%="usuario.getUsername()%>">
```

**SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES\_REPORT11\_PARAMFORM.jsp****LINEA 417**

```
<%-- AHERRRENO 02/11/2010 --%>
<%
    Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO);
%>
<input type="hidden" name="<%="AWReportes.REPORTE_11_PARAM_USUARIOLOG%">
id="<%="AWReportes.REPORTE_11_PARAM_USUARIOLOG%">
value="<%="usuario.getUsername()%>">
```

**Script Bases de Datos:****SP\_Reporte\_09.sql**

--AHERRENO 04/11/2010

--Se crea cursor para validar actos relacionados a una solicitud  
CURSOR c\_actos (p\_sol VARCHAR2) IS SELECT tpct\_nombre FROM sir\_op\_acto a, sir\_op\_tipo\_acto ta  
WHERE a.id\_tipo\_acto = ta.id\_tipo\_acto  
AND a.id\_solicitud = p\_sol;

```
--AHERRENO 04/11/2010
--Se abre cursor para concatenar actos asociados a una solicitud
v_nombre := null;
OPEN c_actos(t_liq(liq).id_solicitud);
LOOP
  FETCH c_actos BULK COLLECT INTO t_actos LIMIT 1000;
  EXIT WHEN c_actos%NOTFOUND AND t_actos.COUNT = 0;
  FOR act IN t_actos.FIRST..t_actos.LAST
  LOOP
    IF act = 1 THEN
      v_nombre := t_actos(act).tpct_nombre ;
    ELSE
      v_nombre := v_nombre || ',' || t_actos(act).tpct_nombre ;
    END IF;
  END LOOP;
END LOOP;
CLOSE c_actos;
```

**SP\_Reporte\_09\_Marcha\_Atras.sql (Archivo Adjunto)****154 Relaciones generadas en reparto Registro y avance de turnos****REQUERIMIENTO****RELACIONES GENERADAS EN REPARTO REGISTRO Y AVANCE DE TURNOS**

Se generó relación de Reparto Registro No. 6691177 de fecha 13-11-2009 para cuatro usuarios calificadores, en ambiente de pruebas de la Orip de Pasto y se encontró lo siguiente:

En el reporte generado inicialmente tiene logo y escudo de la Superintendencia y en el genérico no, esta presentación debe ser estándar para todos los reportes e imprimibles llevando el logo y escudo.

En el reporte genérico omite el número del reparto, en este reporte se debe reflejar la misma información contenida en el reporte inicial.

En la columna matrículas del generado inicialmente no las está ordenando, se requiere que sean ordenadas en forma ascendente,

En el reporte genérico en la columna turno está repitiendo los registros se requiere que no se repita el número de turno.

En ambiente de pruebas se avanzaron turnos de registro por relación así:  
Se generó relación No. 2009-300-4932 de mesa de control a Firma avanzando  
Los siguientes turnos (33).

2009-300-6-36808, 2009-300-6-37034, 2009-300-6-37182, 2009-300-6-37184  
2009-300-6-37187, 2009-300-6-37294, 2009-300-6-37295, 2009-300-6-37313  
2009-300-6-37345, 2009-300-6-37380, 2009-300-6-37395, 2009-300-6-37413  
2009-300-6-37417, 2009-300-6-37422, 2009-300-6-37427, 2009-300-6-37432  
2009-300-6-37440, 2009-300-6-37454, 2009-300-6-37459, 2009-300-6-37477  
2009-300-6-37489, 2009-300-6-37498, 2009-300-6-37508, 2009-300-6-37510  
2009-300-6-37517, 2009-300-6-37518, 2009-300-6-37520, 2009-300-6-37524  
2009-300-6-37525, 2009-300-6-37535, 2009-300-6-37536, 2009-300-6-37552  
2009-300-6-37554.

Luego al avanzarlos de firma a la fase Certificados Asociados el sistema no permite avanzar 8 turnos del listado inicial, se observa que cuando tienen asociada la misma matrícula más de un turno, el sistema muestra mensaje de error como el siguiente pantallazo.

The screenshot displays a web-based application interface. At the top, the title bar reads "SNR :: Superintendencia de Notariado y Registro :: Ministerio del interior y de Justicia :: Rep - Windows Internet Explorer" and the URL "http://192.168.202.9:8100/SNR/relation.do". Below the title bar, there is a header "Impresión de Relaciones" with a back button and a forward button. The main content area is divided into two sections:

- Turnos Avanzados en la Relación:** A table showing the following data:

2009-300-6-36808	2009-300-6-37034	2009-300-6-37187	2009-300-6-37294	2009-300-6-37295
2009-300-6-37313	2009-300-6-37345	2009-300-6-37380	2009-300-6-37395	2009-300-6-37413
2009-300-6-37417	2009-300-6-37422	2009-300-6-37427	2009-300-6-37432	2009-300-6-37440
2009-300-6-37454	2009-300-6-37459	2009-300-6-37477	2009-300-6-37498	2009-300-6-37510
2009-300-6-37518	2009-300-6-37535	2009-300-6-37536	2009-300-6-37552	2009-300-6-37554
- Turnos NO Avanzados en la Relación:** A list of errors corresponding to specific turn numbers:

2009-300-6-37182	Error Avanzando Turno, Fallo en el servicio de delegar bloqueo de folio
2009-300-6-37525	Existen turnos activos anteriores que comparten folios y no han sido firmados.
2009-300-6-37524	Error Avanzando Turno, Fallo en el servicio de delegar bloqueo de folio
2009-300-6-37489	Error Avanzando Turno, Fallo en el servicio de delegar bloqueo de folio
2009-300-6-37508	Error Avanzando Turno, Fallo en el servicio de delegar bloqueo de folio
2009-300-6-37520	Existen turnos activos anteriores que comparten folios y no han sido firmados.
2009-300-6-37517	Existen turnos activos anteriores que comparten folios y no han sido firmados.
2009-300-6-37184	Existen turnos activos anteriores que comparten folios y no han sido firmados.

En la fase de certificados asociados el sistema no permite imprimir los certificados asociados a dos turnos de registro mostrando mensaje de error porque las matrícula asociada al certificado no tuvo cambios, se consultaron las matrículas y se observa que en estos mismos turnos se inscribieron anotaciones en la fase de calificación, según este pantallazo.

SNR :: Superintendencia de Notariado y Registro :: Ministerio del interior y de Justicia :: Rep - Windows Internet Explorer  
 http://192.168.202.9:8100/SNR/jsp/plantillas/report-displayer/mesa.do

**Error en la Validación de Datos del Formulario**

- Los Siguientes Errores han sido encontrados:
- Para el turno: 2009-300-1-179376
- La matrícula 300-222292, asociada al certificado no tuvo cambios. Por lo tanto, no se permite su impresión.
- Para el turno: 2009-300-1-179375
- La matrícula 300-222291, asociada al certificado no tuvo cambios. Por lo tanto, no se permite su impresión.

[« Volver](#)

**Certificados Asociados**

Turnos de certificados asociados para los documentos de la relación 2009-300-4932. Turnos de documentos INSCRITOS.

Turno registro	Seleccionar	Turno asociado	Matrícula	Número copias	Cambiar Matrícula
2009-300-6-36808	<input type="checkbox"/>	2009-300-1-175782	300-49131	1	<a href="#">Cambiar Matrícula</a>
2009-300-6-37034	<input type="checkbox"/>	2009-300-1-177507	300-18546	1	<a href="#">Cambiar Matrícula</a>
2009-300-6-37187	<input type="checkbox"/>	2009-300-1-177985	300-148093	1	<a href="#">Cambiar Matrícula</a>
2009-300-6-37294	<input type="checkbox"/>	2009-300-1-177986	300-148001	1	<a href="#">Cambiar Matrícula</a>
2009-300-6-37295	<input type="checkbox"/>	2009-300-1-178146	300-325145	1	<a href="#">Cambiar Matrícula</a>
2009-300-6-37313	<input type="checkbox"/>	2009-300-1-178221	300-256957	1	<a href="#">Cambiar Matrícula</a>
2009-300-6-37345	<input type="checkbox"/>				
2009-300-6-37380	<input type="checkbox"/>				
2009-300-6-37395	<input type="checkbox"/>				
2009-300-6-37413	<input type="checkbox"/>				
2009-300-6-37417	<input type="checkbox"/>	2009-300-1-178734	300-240937	1	<a href="#">Cambiar Matrícula</a>
2009-300-6-37422	<input type="checkbox"/>	2009-300-1-178770	300-158	1	<a href="#">Cambiar Matrícula</a>
2009-300-6-37427	<input type="checkbox"/>				
2009-300-6-37432	<input type="checkbox"/>				
2009-300-6-37440	<input type="checkbox"/>				
2009-300-6-37454	<input type="checkbox"/>	2009-300-1-178894	300-152093	1	<a href="#">Cambiar Matrícula</a>
2009-300-6-37459	<input type="checkbox"/>				
2009-300-6-37477	<input type="checkbox"/>				
2009-300-6-37498	<input type="checkbox"/>				
2009-300-6-37510	<input type="checkbox"/>				
2009-300-6-37518	<input type="checkbox"/>				
2009-300-6-37535	<input type="checkbox"/>	2009-300-1-179375	300-222291	1	<a href="#">Cambiar Matrícula</a>
2009-300-6-37536	<input type="checkbox"/>	2009-300-1-179376	300-222292	1	<a href="#">Cambiar Matrícula</a>
2009-300-6-37552	<input type="checkbox"/>				
2009-300-6-37554	<input type="checkbox"/>				

En la relación de documentos de Certificados Asociados a desanotados para Entrega no lista los turnos en orden, esta inconsistencia se observan en el listado inicial y en el generado en Pantallas Administrativas-Reportes.

La ETB realizo ajuste a la funcionalidad respecto al listar los turnos de registro en orden ascendente, se generaron nuevamente las relaciones en ambiente de pruebas, avanzando turnos de la fase confrontar a Reparto Registro y de la fase de mesa de control hasta entrega, y se observan que los turnos de registro se listan en orden y no se duplican los turnos de registro que tienen más de un turno de certificado asociado, lo cual es correcto en las relaciones generadas inicialmente (proceso) y en Pantallas Administrativas (en reportes).

En las relaciones iniciales y en las generadas en Pantallas Administrativas (reportes) se observa lo siguiente :

En la relación de la fase reparto Registro en la columna matriculas el sistema no las ordena cuando se han asociado más de una matrícula

En la relación de las fases de mesa de control, firma de registro y certificados asociados en la columna de certificados asociados no ordena los turnos de certificados cuando

existen más de un turno de certificado asociado a un turno de registro .

Se observa que continua la inconsistencia cuando una matriculas se encuentra asociada a más de un turno de registro el sistema no permite avanzarlo por relación,

### **DESARROLLO**

A continuación se detallan entre estos símbolos << >> las anotaciones realizadas por Yamile Moreno Melo y Héctor Ejalde Troyano, sucesivamente se anota la aclaración a las mismas.

<<

Luego al avanzarlos de firma a la fase Certificados Asociados el sistema no permite avanzar 8 turnos del listado inicial, se observa que cuando tienen asociada la misma matricula más de un turno, el sistema muestra mensaje de error como el siguiente pantallazo.

>>

El mensaje de verificación es acertado dado que la lógica del negocio está establecida de tal manera que un turno activo no puede tener un folio que esté asignado a un turno anterior dado que como el sistema informa el turno anterior bloqueará el folio para cualquier turno creado posteriormente que desea usar dicho folio.

<<

En la fase de certificados asociados el sistema no permite imprimir los certificados asociados a dos turnos de registro mostrando mensaje de error porque las matricula asociada al certificado no tuvo cambios, se consultaron las matriculas y se observa que en estos mismos turnos se inscribieron anotaciones en la fase de calificación, según este pantallazo.

>>

El mensaje de verificación es apropiado dado que no tiene ninguna razón imprimir un certificado de una matrícula o folio que no ha sufrido una modificación.

<<

Se observa que continua la inconsistencia cuando una matriculas se encuentra asociada a más de un turno de registro el sistema no permite avanzarlo por relación.

>>

Como se anotó previamente un turno activo no puede ser avanzado si tiene asignada un folio que se está siendo usado por un turno previo, dado que el turno previo bloquea el dicho folio.

Reporte REL01: Reporte genérico de relaciones

Script Bases de Datos:

NINGUNO

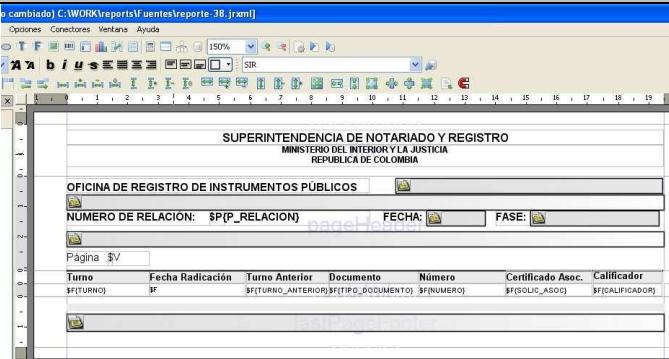
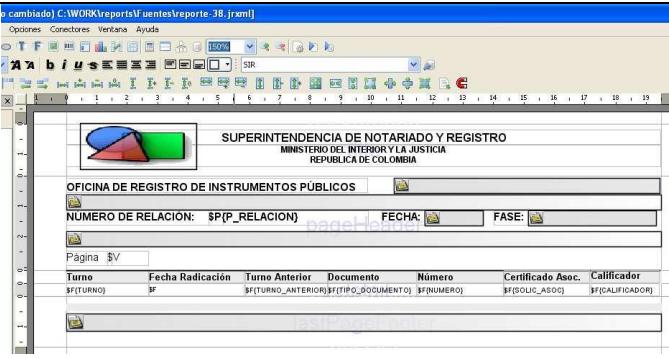
Creación De Clases:

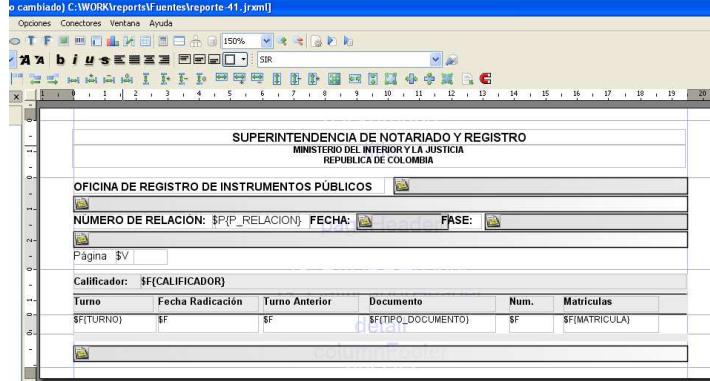
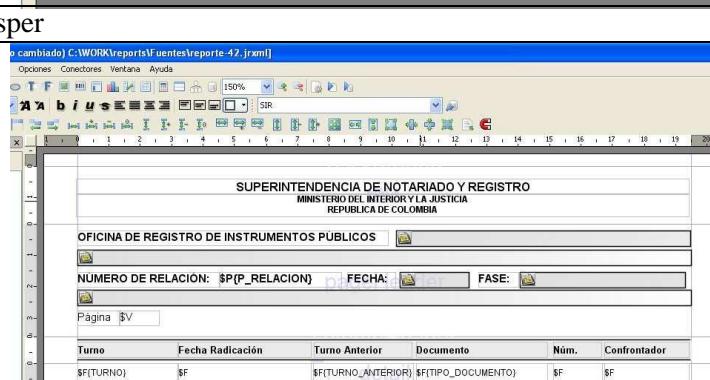
NINGUNO

Modificación de archivos:

reporte-35.jasper

```
/*AHERRENO 23/07/2010
SE AGREGA ORDENAMIENTO POR CERTIFICADOS ASOCIADOS*/
,turno_sol_asoc.trno_id_workflow
```

	<b>reporte-38.jasper</b>
/*AHERRENO 23/07/2010 SE AGREGA ORDENAMIENTO POR CERTIFICADOS ASOCIADOS*/ .turno_sol_asoc.trno_id_workflow	
	<b>reporte-41.jasper</b>
/***** AHERRENO 26/07/2010 ****/ /* SE AGREGA ORDENAMIENTO POR MATRICULA REQ 154*/ ,matricula /*****	
	<b>reporte-42.jasper</b>
***** AHERRENO 26/07/2010 ***** /*SE MODIFICA ORDENAMIENTO REQ 154*/ TO_DATE(turno.trno_fecha_creacion, 'DD-MM-YYYY HH:MI:SS AM'), turno.trno_id_workflow /*****	
	<b>reporte-97.jasper</b>
/*AHERRENO 23/07/2010 SE AGREGA ORDENAMIENTO POR CERTIFICADOS ASOCIADOS*/ ,Solic_Asoc	
	<b>reporte-155.jasper</b>
/***** AHERRENO 26/07/2010 ****/ /* SE AGREGA ORDENAMIENTO POR MATRICULA REQ 154*/ ,solicitud_folio.id_matricula /*****	
	<b>reporte-38.jasper</b>
ANTES	
AHORA	
	<b>reporte-41.jasper</b>

	ANTES	
	AHORA	
<b>reporte-42.jasper</b>		
	ANTES	
	AHORA	
<b>Reporte-155.jasper</b>		

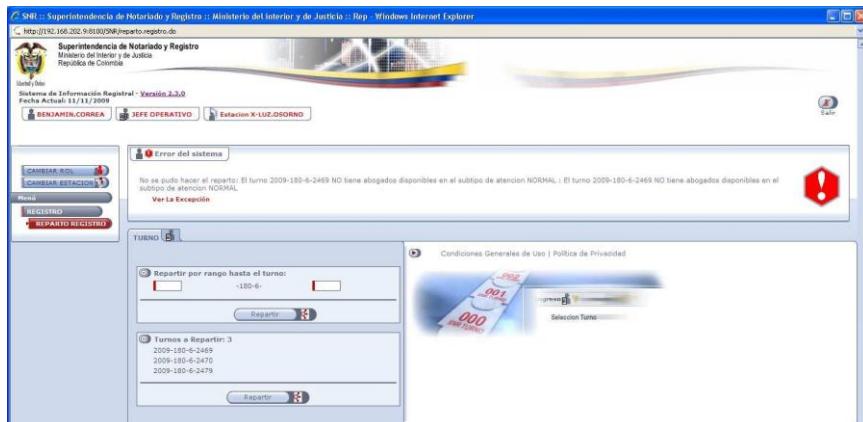
<p><b>ANTES</b></p>	
<p><b>AHORA</b></p>	
<b>155 Reparto registro -Subtipos de atencion</b>	
<p><b>REQUERIMIENTO</b> REPARTO REGISTRO</p> <p>Se efectúo prueba seleccionando subtipos de atención para calificadores de copias, gran tamaño, testamentos, reglamentos para usuarios activos, se realizo el reparto de turnos de registro y el sistema muestra el siguiente mensaje:</p>	
<p>El mensaje debe ajustarse: no debe mostrar los turnos, sino debe decir algo general, “No existen subtipos de atención para calificadores del tipo (copias, gran tamaño, reglamento, testamento, normal según sea el caso) “.</p>	

En ambiente de pruebas se efectuó reparto de la Orip de Quibdó con usuarios inactivos y seleccionando todos los subtipos de atención para calificadores a excepción de la opción normal o seleccionando únicamente normal, el sistema muestra el mensaje coherente sin embargo lo repite y genera confusión y permite volver a la pantalla donde se realiza el reparto mostrando el siguiente mensaje “No se pudo hacer el reparto: El subtipo de atención NORMAL NO tiene calificadores con estaciones disponibles, se debe habilitar por lo menos un calificador con su respectiva estación : El subtipo de atención NORMAL NO tiene calificadores con estaciones disponibles, se debe habilitar por lo menos un calificador con su respectiva estación “

### **En este texto se debe corregir la palabra calicadores por calificadores.**

“No se pudo hacer el reparto: No existen subtipos de atención para calicadores del tipo: GRAN TAMAÑO. : No existen subtipos de atención para calicadores del tipo: GRAN TAMAÑO”

Se realizo prueba de reparto de documentos para usuario activos y sin seleccionar subtipos de atención para calificadores el sistema no permite realizar el reparto validez que es correcta y muestra mensaje “No se pudo hacer el reparto: El turno 2009-180-6- 2469 NO tiene abogados disponibles en el subtipo de atención NORMAL : El turno 2009- 180-6-2469 NO tiene abogados disponibles en el subtipo de atención NORMAL” en este mensaje se observa que el sistema lo duplica lo correcto es que no debe repetir el texto del mensaje.



Se requiere que el mensaje sea general como se solicito anteriormente, no debe mencionar ningún turno.

Se expuso en reunión de seguimiento del Aplicativo SIR respecto a las inconsistencias encontradas en la revisión del ajuste realizado a la funcionalidad de Reparto Registro cuando no se seleccionan subtipos de atención y se intenta hacer reparto con usuarios activos e inactivos y que el requerimiento como tal se realizo, sin embargo se acordó la prioridad de desarrollar los ajustes a las inconsistencias halladas en la revisión del despliegue de esta fecha

## **DESARROLLO**

### **CLASE o ARCHIVO JSP:**

**C:\WORK\SIR-APP\src\gov\sir\core\negocio\acciones\registro\ANReparto.java**

### **Método:**

**Código Línea 669:**

```

        throw new EventoException("No se pudo hacer el reparto: " + t.getMessage(),t);

Por:
/*
 * @author   : Henry Gómez Rocha
 * @change   : Se modifica el número de parametros enviados para evitar que se imprima dos
 *              veces el mismo mensaje.
 * Caso Mantis : 0005012
 */
        throw new EventoException("No se pudo hacer el reparto: " + t.getMessage());

```

### Código Línea 771:

```

if(abogados==null){
    throw new EventoException("El turno "+t.getIdWorkflow()+" NO tiene abogados disponibles en el
    subtipo de atencion "+sr.getSubtipoAtencion().getNombre());
}

```

```

if(abogados.size()==0){
    throw new EventoException("El turno "+t.getIdWorkflow()+" NO tiene abogados disponibles en el
    subtipo de atencion "+sr.getSubtipoAtencion().getNombre());
}

```

### Por:

```

/*
 * @author   : Henry Gómez Rocha
 * @change   : Se unen dos if y así mismo, se modifica el mensaje evitando que se muestre en la
 *              excepción el turno y a su vez se hace más claro el mensaje final para el usuario.
 * Caso Mantis : 0005012
 */
if(abogados==null || abogados.size()==0){
    throw new EventoException("No existen subtipos de atención para calificadores del tipo
    "+sr.getSubtipoAtencion().getNombre());
}

```

### Código Línea 863:

```

while(llaves.hasNext()){
    SubtipoAtencion subtipo = (SubtipoAtencion)llaves.next();
    Long turnosPorSubtipo = (Long)numeroTurnosPorSubtipo.get(subtipo);
    if(mapSubtipoUsuarios.containsKey(subtipo)){
        List abogados = (List)mapSubtipoUsuarios.get(subtipo);
        numeroAbogados = abogados.size();
        if(numeroAbogados==0){
            throw new EventoException("El subtipo de atención "+subtipo.getNombre()+" NO tiene
            calificadores con estaciones disponibles, se debe habilitar por lo menos un calificador con su
            respectiva estación");
        }
        promedio = turnosPorSubtipo.longValue() / numeroAbogados;
        if(turnosPorSubtipo.longValue()%numeroAbogados!=0){
            promedio++;
        }
    }
    else{
        //El -1 indica que no existen usuarios del subtipo de atención indicado
        //que puedan atender turnos de ese subtipo de atención
        promedio = -1;
    }

    //Se ingresa el promedio en la tabla
    rta.put(subtipo, new Long(promedio));
}

```

### Por:

```

/*
 * @author   : Henry Gómez Rocha
 * @change   : Se crea la variable evitaDosImpresiones con el fin de evitar
 *              que el mismo mensaje sea presentado al usuario final dos veces.
 * Caso Mantis : 0005012
 */
boolean evitaDosImpresiones = true;
while(llaves.hasNext()){
    SubtipoAtencion subtipo = (SubtipoAtencion)llaves.next();
    Long turnosPorSubtipo = (Long)numeroTurnosPorSubtipo.get(subtipo);
    if(mapSubtipoUsuarios.containsKey(subtipo)){
        List abogados = (List)mapSubtipoUsuarios.get(subtipo);
        numeroAbogados = abogados.size();
        if(numeroAbogados==0 && evitaDosImpresiones){
            evitaDosImpresiones = false;
            throw new EventoException("El subtipo de atención "+subtipo.getNombre()+" NO tiene
            calificadores con estaciones disponibles, se debe habilitar por lo menos un calificador con su
            respectiva estación");
        }
        promedio = turnosPorSubtipo.longValue() / numeroAbogados;
    }
}

```

```

        if(turnosPorSubtipo.longValue()%numeroAbogados!=0){
            promedio++;
        }
    }else{
        //El -1 indica que no existen usuarios del subtipo de atención indicado
        //que puedan atender turnos de ese subtipo de atención
        promedio = -1;
    }

    //Se ingresa el promedio en la tabla
    rta.put(subtipo, new Long(promedio));
}

```

**Código Línea 948:**

```

if(abogados == null){
throw new EventoException("No existen subtipos de atención para calicadores del tipo: " +
sr.getSubtipoAtencion().getNombre());
}

```

**Por:**

```

/*
 * @author : Henry Gómez Rocha
 * @change : Se corrige la palabra calicadores por calificadores.
 * Caso Mantis : 0005012
 */
if(abogados == null){
throw new EventoException("No existen subtipos de atención para calificadores del tipo: " +
sr.getSubtipoAtencion().getNombre());
}

```

**Observaciones:**

Se hicieron los respectivos ajustes a los mensajes y se verificó de igual forma que no se muestren dos veces al usuario final dichos mensajes. Se hicieron las pruebas con la colaboración de Javier Rocha quien hace parte del personal de mesa de ayuda.

En las pruebas realizadas para el despliegue 002\_6\_2 se observó lo siguiente:

1. El mensaje ya no se duplica, lo cual es correcto
2. El mensaje es coherente siempre y cuando solo exista un turno en el reparto.
3. El mensaje no es coherente si existen varios turnos con diferentes subtipos de atención, solo muestra el subtipo de atención del primer turno y no muestra los otros subtipos que no se encuentran configurados. Estos subtipos de atención deben estar separados por coma.

**CLASE:**

**gov.sir.core.negocio.acciones.registro.ANReparto.java**

**Código Insertado Línea 86:**

```

/*
 * @author : Julio Alcazar Rivas
 * @change : Se define el Set ListaSubtipos como HashSet que guarda los subtipos que no tiene
calificadores asignados
 * Caso Mantis : 0005012
*/
private Set ListaSubtipos;

```

**Método:**

**private EvnRespReparto crearReparto(EvnReparto evento) throws  
EventoException {**

**Código Insertado Línea 155:**

```

/*
 * @author : Julio Alcazar Rivas
 * @change : Se define el Set ListaSubtipos como HashSet que guarda los subtipos que no tiene
calificadores asignados
 * Caso Mantis : 0005012
*/
ListaSubtipos = new HashSet();

```

**Código Insertado Línea 579:**

```

/*
 * @author    : Julio Alcazar Rivas
 * @change    : Se lanza una excepcion de los tipos de atencion que no tienen calificador agrupados
en el HashSet ListaSubtipos
 * Caso Mantis : 0005012
*/
if (!ListaSubtipos.isEmpty()){
    Iterator ite = ListaSubtipos.iterator();
    String subtipos = " ";
    while (ite.hasNext()){
        if (" ".equals(subtipos)){
            subtipos += ", ";
        }
        subtipos += (String) ite.next();
    }
    throw new EventoException("Los subtipos de atencion"+subtipos+" NO tiene calificadores con
estaciones disponibles, se debe habilitar por lo menos un calificador con su respectiva estación");
}

```

**Método:**

```

private void repartirPorListaRotada(Turno t, Map mapSubtipoUsuarios,
Map mapSubtipoPromedio, Map mapUsuarioTurnos) throws
EventoException {

```

**Código Línea 786:**

```

throw new EventoException("No existen subtipos de atención para calificadores del tipo
"+sr.getSubtipoAtencion().getNombre());

```

**Por:**

```

/*
 * @author    : Julio Alcazar Rivas
 * @change    : a)Se agregan los subtipos de atención que no tienen calificador a um    HashSet
para luego enviar una excepción global de los turnos en reparto.
        b)Se crea un else para que continue el camino en el caso que el subtipo de
atención tenga calificador configurado.
 * Caso Mantis : 0005012

*/
ListaSubtipos.add(sr.getSubtipoAtencion().getNombre());
}
else {

```

**Método:**

```

private Map calcularPromedioPorSubtipoAtencion(List allTurnos, Map
mapSubtipoUsuarios) throws EventoException {

```

**Código Línea 870:**

```

long promedio;

```

**Por:**

```

/*
 * @author    : Julio Alcazar Rivas
 * @change    : Se inicializa la variable promedio = 0
 * Caso Mantis : 0005012
*/
long promedio = 0;

```

**Código Línea 885:**

```

evitaDosImpresiones = false;
throw new EventoException("El subtipo de atención "+subtipo.getNombre()+" NO tiene calificadores
con estaciones disponibles, se debe habilitar por lo menos un calificador con su respectiva estación");
}

```

```

promedio = turnosPorSubtipo.longValue() / numeroAbogados;
        if(turnosPorSubtipo.longValue()%numeroAbogados!=0){
            promedio++;
        }

```

**Por:**

```

/*
 * @author    : Julio Alcazar Rivas
 * @change    : a)Se agregan los subtipos de atencion que no tienen calificador a un HashSet para
luego enviar una excepcion global de los turnos en reparto.
        b)Se crea un else para que continue el camino en el caso que el subtipo de atencion
tenga calificador configurado.
 * Caso Mantis : 0005012
*/
    ListaSubtipos.add(subtipo.getNombre());
}

else{
    promedio = turnosPorSubtipo.longValue() / numeroAbogados;
    if(turnosPorSubtipo.longValue()%numeroAbogados!=0){
        promedio++;
    }
}

```

**Método:**

```

private boolean
abogadoPuedeAtender(gov.sir.core.negocio.modelo.Usuario abogado,
Turno t, Map mapSubtipoUsuarios) throws EventoException{

```

**Código Línea 962:**

```

        throw new EventoException("No existen subtipos de atención para calificadores del tipo: " +
sr.getSubtipoAtencion().getNombre());
}
Iterator i = abogados.iterator();
while (i.hasNext() && rta==false){
    Usuario u = (Usuario)i.next();
    if (u.equals(abogado)){
        rta = true;
    }
}

```

**Por:**

```

/*
 * @author    : Julio Alcazar Rivas
 * @change    : a)Se agregan los subtipos de atencion que no tienen calificador a un HashSet para
luego enviar una excepcion global de los turnos en reparto.
        b)Se crea un else para que continue el camino en el caso que el subtipo de
atencion tenga calificador configurado.
 * Caso Mantis : 0005012
*/
    ListaSubtipos.add(sr.getSubtipoAtencion().getNombre());
}
else {
    Iterator i = abogados.iterator();
    while (i.hasNext() && rta==false){
        Usuario u = (Usuario)i.next();
        if (u.equals(abogado)){
            rta = true;
        }
    }
}

```

**162 Número Recibo Pago  
(Informativo)**

**REQUERIMIENTO**

Turno 2009-140-6-11166. Informa Javier Dickson que en el Reporte 99: Ingresos diarios por documentos del 16 de octubre del presente año; aparece el turno 2009-140-6-11166, CALIFICACION X-BETTY.ANAYA, aparece con el número de recibo 40822651, NO coincidiendo este con el recibo de caja original impreso: 40822656. Se generó el reporte 99 con los siguientes parámetros:

Fecha Inicial: 16/10/2009

Fecha Final: 16/10/2009

Círculo Registral: Montería

Aparece el turno 2009-140-6-11166 con el número de recibo 40822651, al consultarlo por ver detalles del turno coincide con el número de recibo, pero no con el número de recibo

impreso: 40822656, favor verificar que pudo haber sucedido y tomar las medidas correctivas, gracias. Montería. Id Glpi 13123.

NOTA: Se aclara que el recibo expuesto 40822656 es error de digitación

## **DESARROLLO**

No existe ninguna inconsistencia en la aplicación, ya que, los valores de pago\_num\_recibo y pago\_last\_num\_recibo de la tabla sir\_op\_pago si pueden ser diferentes en algún momento puesto para un turno pueden existir varios pagos, (Ver pago mayor valor) generando un numero de recibo diferente. Por esto se debe corregir el reporte para tenga en cuenta el campo pago\_last\_num\_recibo.

```
/*AHERRENO 21/07/2010
   SE MODIFICA EL CAMPO DEL NUMERO DEL RECIBO
   ANTES t_pag(pag).pago_num_recibo
   AHORA t_pag(pag).pago_last_num_recibo*/
(v_fecha, TO_CHAR(t_tur(tur).trno_fecha_inicio, 'DD-MM-YYYY'), t_tur(tur).trno_id_workflow
,t_pag(pag).pago_last_num_recibo, v_nombre, v_acto, p_circulo, v_turno_ant.trno_id_workflow, t_act(act).id_tipo_derecho_reg,
t_act(act).acto_valor, t_act(act).acto_cuantia, v_acto_anterior.acto_valor, v_acto_anterior.acto_cuantia, t_act(act).acto_valor -
v_acto_anterior.acto_valor, t_act(act).acto_cuantia - v_acto_anterior.acto_cuantia, p_sec_proceso);
```

Se modifican archivos.

SIR-APPweb\WEB-INF\jspSeguro\ADMINREPORTES_REPORT99_PARAMFORM.jsp
<%-- AHERRENO 02/11/2010 --%> <% Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO); %> <input type="hidden" name="<%="AWReportes.REPORTE_99_PARAM_USUARIOLOG%"> id="<%="AWReportes.REPORTE_99_PARAM_USUARIOLOG%"> value="<%="usuario.getUsername()%>">
Reporte-99.jasper
Se agrega parámetro P_USUARIO_LOG

La diferencia entre los dos campos es; entre pago\_num\_recibo: Numero de recibo generado en el pago pago\_last\_num\_recibo: Ultimo número de recibo impreso para el pago

Esto quiere decir que cuando se modifique el numero de recibo del pago se registrará es el campo pago\_last\_num\_recibo.

Se adjunta imagen de la descripción de la tabla.

**Oracle SQL Developer : TABLE SIR.SIR\_OP\_PAGO@sr**

Archivo Editar Ver Navegar Ejecutar Origen Control de Versiones Migración Herramientas Ayuda

Conexiones Inf. SIR SIR\_OP\_PAGO

Columnas | Datos | Restricciones | Permisos | Estadísticas | Disparadores | Dependencias | Detalles | Índices | SQL

Acciones...

Column Name	Data Type	Nutable	Data Default	COLUMN ID	Primary Key	Comments
ID_LIQUIDACION	VARCHAR2(20 BYTE)	No	(null)	1		1 Identificador del pago correspondiente a la liquidacion
ID_SOLICITUD	VARCHAR2(20 BYTE)	No	(null)	2		2 Identificador del pago correspondiente a la solicitud
PAGO_FECHA	DATE	Yes	(null)	3		(null) Fecha en que se realiza el pago
PAGO_NUM_RECIBO	VARCHAR2(255 BYTE)	Yes	(null)	4		(null) Numero de recibo generado en el pago
PAGO_FECHA_CREAION	DATE	Yes	SYSDATE	5		(null) Fecha creacion de registro en BD
PAGO_CONCEPTO	VARCHAR2(255 BYTE)	Yes	(null)	6		(null) Concepto del pago
ID_USUARIO	NUMBER	Yes	(null)	7		(null) Usuario que realiza el pago
PAGO_USUARIO_ANT	VARCHAR2(15 BYTE)	Yes	(null)	8		(null) Código Usuario Sistema Anterior
PAGO_FECHA_IMPRESION	DATE	Yes	(null)	9		(null) Fecha en que se realiza la ?ltima impresi?n del recibo del pago
PAGO_LAST_NUM_RECIBO	VARCHAR2(255 BYTE)	Yes	(null)	10		(null) Ultimo?n?vera de recibo impreso para el pago
ID_CIRCULO	VARCHAR2(20 BYTE)	No	(null)	11		(null) Identificador del Círculo
ID_NUMERACION	NUMBER(20,0)	Yes	(null)	12		(null) (null)

Mensajes - Log

Historial SQL

sr | SIR | SIR\_OP\_PAGO Editing

Inicio Bandera de e... NOVUR-DOC... Sin t?ulo - Blo... 0004960: Act... src 1206. Bind... Sin t?ulo - Blo... SIR - NetBea... sas.pdf - A... run-debug.bat Sin t?ulo - Blo... Formato Des... Sin t?ulo 1... Oracle SQL D... Microsoft Pro... Skype™ - Fie... http://192.1... ES Lunes 02/08/2010

## 196 Reporte 49\_Reporte de certificados reimpressos

### REQUERIMIENTO

*Requerimiento Glpi No 17327 - Reporte 49:Reporte de certificados reimpressos*

Motivo Inconsistencia: Informa la funcionaria Amparo Pastor que al generar el reporte 49: Reporte de certificados reimpressos del mes de enero hasta la fecha aparecen turnos reimpressos por el funcionario Napoleón Álvarez y también aparecen certificados reimpressos por las usuarias Eneida Assia y Amparo Pastor, pero la funcionaria amparo Pastor informa que estas usuarias no realizaron reimpresiones de certificados.

Se genera Reporte 49: Reporte de certificados reimpressos con los siguientes parámetros:

Fecha Inicial: 1/01/2010  
 Fecha Final: 26/04/2010  
 Círculo Registral: Sincelejo

En el reporte se logra ver los id de los usuarios que generaron las reimpresiones, favor informar estos id a que usuarios corresponden. Grupo Sincelejo

### DESARROLLO

Script Bases de Datos:  
**SP\_REPORTE\_49.sql**

```

*****
/** AHERREN 25/02/2010 */
/** Se agrega el campo id_usuario de la tabla SIR_OP_NOTA
/** campo en el cual se almacena la informaci?n de el usuario que
/** hace la reimpre?n
*****
SELECT s.id_solicitud, t.trno_id_workflow, f.id_matricula, c.cddn_apellido1, c.cddn_apellido2, c.cddn_nombre, l.lqdc_valor,
s.sltc_num_impresiones_cr, l.lqdc_valor_derechos_rg, t.trno_fecha_inicio, s.sltc_comentario, tc.tpcr_nombre, tn.tpna_nombre,
n.id_usuario
*****
/** AHERREN 25/02/2010 */

```

```

    /** Se elimina el campo t_his(his).id_usuario1 y se
    /** agrega el campo id_usuario de la tabla SIR_OP_NOTA
    /** campo en el cual se almacena la información de el usuario que
    /** hace la reimpresión "cursor c_tur"
    *****/
    t_tur(tur).id_usuario,

```

### SP\_REPORTE\_49\_Marcha\_Atras.sql (Archivo Adjunto)

Modificación Clase:

NINGUNO

Modificación de archivos:

#### SIR-APP\web\WEB-INF\jsp\Seguro\administracion\ADMINREPORTES\_REPORT49\_PARAMFORM.jsp

```

<%-- AHERRRENO 02/11/2010 --%>
<% Usuario usuario = (Usuario)request.getSession().getAttribute(WebKeys.USUARIO); %>
<input type="hidden" name="<%=_AWReportes.REPORTE_49_PARAM_USUARIOLOG%>" 
id="<%=_AWReportes.REPORTE_49_PARAM_USUARIOLOG%>" 
value="<%=_usuario.getUsername()%>">

```

#### Reporte-49.jasper

Se agrega parámetro P\_USUARIO\_LOG

#### 198 Error al cargar la pagina cuando se esta trabajando el turno

#### **REQUERIMIENTO Glpi No 17413**

Turno: 2009-350-3-1870

Matricula(s):350-76351 hasta el 350-76351.

Motivo Inconsistencia: Al tratar de trabajar el turno 2009-350-3-1870 no carga y se cae la página.

En la oficina están tratando de trabajar el turno 2009-350-3-1870 en la fase ejecutar corrección simple para realizar la respectiva corrección de los folios, pero al momento de tomarlo para trabajarla el sistema se queda cargando y se cae la página. Se adjunta la información que aparece al caerse la página y tratar de trabajar el turno. Grupo Ibagué

#### **DESARROLLO**

Se revisó en el proceso de corrección la fase Ejecutar corrección simple, y se encontró que efectivamente se demora excesivamente al momento de tomar los folios. Se encontró que el 90% del tiempo empleado para la toma de los folios los ocupa realizando una verificación de folios innecesaria que al final nunca es usada.

Se resuelve quitar estas líneas de código innecesarios. Se adjunta solución en el archivo "Formato Desarrollo Solución de Caso Mantis 4574.doc". Además se realizó una prueba de tiempo de carga para un turno con 453 folios asociados, sin la modificación la toma de los folios tardó 6:05 min, mientras que con la modificación se tardó 0:53 seg. Respetando todas las reglas de negocio actualmente configuradas en el sistema.

Turno 2010-999-3-387, ambiente -> producción.

Se modificó el siguiente archivo:

#### **CLASE o ARCHIVO JSP:**

#### **gov.sir.core.web.acciones.correccion.AWCorreccion.java**

#### **Línea 1157:**

```

    /**
     * @author: Fernando Padilla
     * @change: 4574: Acta - Requerimiento No 198 - No carga la pagina,
     *           Se elimina este código inoficioso.
     */
    try{
        ProcesadorEventosNegocioProxy proxy = new

```

```

ProcesadorEventosNegocioProxy();

        Evn_CorrSimpleMain_VerAlertasOptions eventoAlerta;
        eventoAlerta = new Evn_CorrSimpleMain_VerAlertasOptions(
Evn_CorrSimpleMain_VerAlertasOptions.CORRECCIONSIMPLEMAIN_VERALERTASOPTIONS_STEPO_BTNS
TART__EVENT );

        eventoAlerta.setUsuarioAuriga( usuarioAuriga );
        eventoAlerta.setUsuarioSir( usuarioSir );
        eventoAlerta.setTurno( turno );

        EvnResp_CorrSimpleMain_VerAlertasOptions respuesta =
            (EvnResp_CorrSimpleMain_VerAlertasOptions)proxy.manejarEvento(
                eventoAlerta);

        List lstFoliosSinCambios = respuesta.getFoliosInTurnoSinCambiosList();
        if(lstFoliosSinCambios!=null && lstFoliosSinCambios.size()>0){

            request.getSession().setAttribute("GENERAR_ALERTA_SIN_CAMBIOS", "TRUE");
        }
    }catch (Exception e) {
        // TODO: handle exception
    }/*

```

### 203 Desfase de Id anotacion temporales

#### **REQUERIMIENTO GLPI No 17411**

Turno: 2010-260-6-2766

Matricula(s): 260-216029, 260-265726, 260-265727.

Motivo Inconsistencia: El turno no avanza de la fase FIRMAR REGISTRO el sistema notifica error. Se adjunta imagen de error. Grupo Cúcuta

En el proceso de segregación se crean las anotaciones temporales para los folios segregados, con desfase en el ID\_ANOTACION con respecto al número de orden de la anotación.

#### **DESARROLLO**

Para este error reportado, no se pudo replicar en las pruebas realizadas.

Se realizaron pruebas en la aplicación desplegada en la maquina <http://192.168.202.9:8100/SNR/inicio.view>, [^] con el mismo turno y matricula reportada en el caso 5299, pero la aplicación se comportó correctamente y las segregaciones se realizaron sin ningún inconveniente.

Datos de pruebas

TURNOS: 2010-240-6-9991

MATRICULA: 240-25197

USUARIO: ANA.PATINO

CIRCULO: PASTO

Igualmente se realizó una validación en el paquete SIR.SP\_SP para validar los id de las anotaciones para los folios hijos o derivados.

#### **PROCEDIMIENTO ALMACENADO**

##### **SIR\_SP\_SG**

###### **Código Adicionado**

```

--@AUTOR:GUILLERMO CABRERA
--@FECHA:19-07-2010
--@CAMBIO:SE ADICIONA PARA VERIFICAR QUE EL ID DE LA ANOTACIÓN SEA EL CORRECTO
--ES DECIR QUE AL CREARSE UN FOLIO NUEVO, ESTE SE CREE CON EL ID DE ANOTACION EN 1

```

```
--Y NO COMIENCE EN OTRO ID COMO LOS CASOS REPORTADOS
--MANTIS 5306 ACTA 203.

SELECT COUNT(ID_ANOTACION_TMP)
INTO N_COMPARA_ID
FROM SIR.SIR_NE_ANOTACION_TMP
WHERE ID_MATRICULA = V_ID_MATRICULA_HIJA;

IF V_CONTADOR_ID_ANOTACION != N_COMPARA_ID+1 THEN
    V_CONTADOR_ID_ANOTACION:=N_COMPARA_ID;
END IF;
```

## 204 Bloqueo turno de digitación a calificación (Informativo)

### **REQUERIMIENTO**

Bloqueo de turnos.

Se viene presentando cuando los folios asociados al turno en la fase DIGITACIÓN CALIFICACIONES, se llevan a la fase CALIFICACION, quedando bloqueados por el usuario que lo trabajo en DIGITACION y viceversa.

Se anexa GLPI No 19455

Turno: 2010-382-6-1295

Matricula(s): 382-6268, 382-25936, 382-25937.

Motivo Inconsistencia: Error de bloqueo al trabajar los folios en la fase de DIGITACION CALIFICACIONES.

La funcionaria Imelda Muñoz envía E-mail informando error de bloqueo de usuario que notifica el sistema al momento de trabajar los folios 382-6268, 382-25936, 382-25937 en la fase de DIGITACION CALIFICACIONES. El folio 382-6268 se encuentra Bloqueado por: HECTOR DIEZ Turno: 2010-382-6-1295 Fecha del bloqueo: 13/7/2010. Por favor asignar el bloqueo de los folios a la usuaria X-MARIA.AGUILERA para seguir con el trámite del turno.

### **DESARROLLO**

La solución dada al Caso mantis 0004967 cuya Acta de Requerimiento es la No 148 y lleva por título "Digitador masivo – Direcciones - Cabida y Linderos" da solución al presente problema.

## 209 Configuracion de opcion de pago

### **REQUERIMIENTO**

La usuaria Clara Ochica nos informa que al tratar de realizar un registro manual de certificado el sistema no permite registrar el pago.

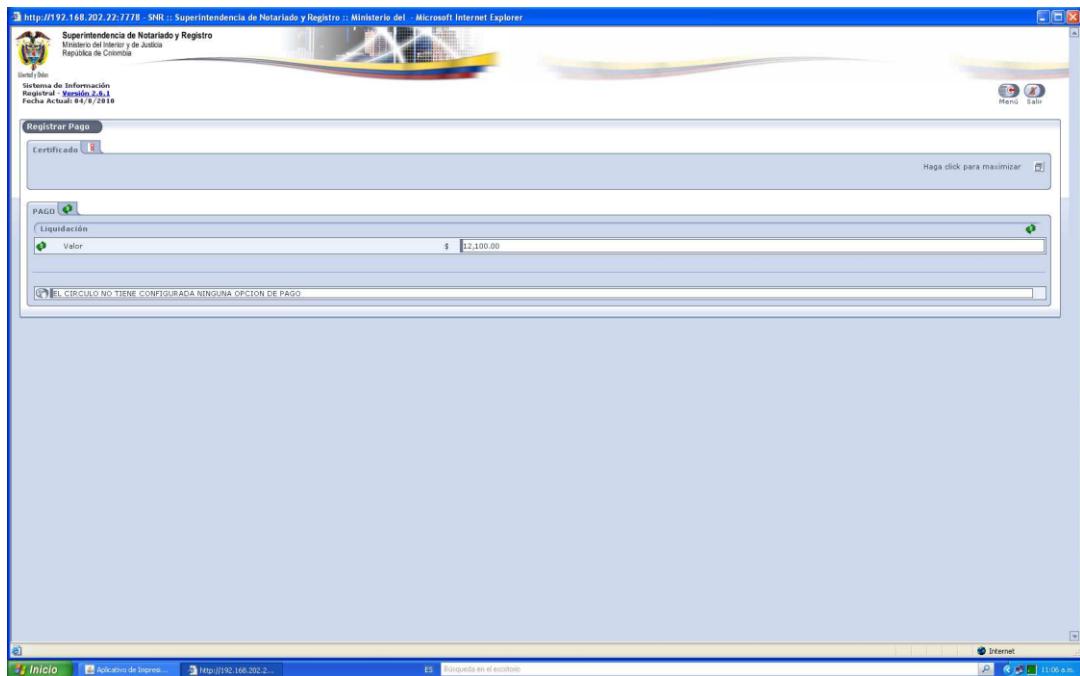
Al tratar de realizar un registro manual de certificado el sistema no permite registrar el pago, el pago es en consignación y cuando elimina el pago en efectivo el sistema informa que el circulo no tiene configurada ninguna opción de pago, se anexa imagen del error.

Grupo San José del Guaviare

### **DESARROLLO**

Para dar solución a este problema se modifco el archivo org.auriga.smart.web.xml que se encuentra en mapas.jar; Se modifco el tag url-origen = turnoManualCertificadoPago.do cambiando el control de navegación cuando el resultado sea RELISTAR\_APPLICACIONES\_CERTIFICADOS, la nueva url-redireccion para este

resultado será admin.turno.manual.certificado.registrar.pago.view.



## 210 Reporte 49\_Reporte de certificados reimpressos

### REQUERIMIENTO

La usuaria MARIA ISABEL BENAVIDES CAICEDO nos informa que al generar el Reporte 49: Reporte de certificados reimpressos la usuaria MARÍA ISABEL BENAVIDEZ con el código No. 53547 aparece con muchos certificados reimpressos, al verificar los roles asignados a dicha usuaria no tienen relación con la opción de reimpresión de certificados.

Al generar el Reporte 49: Reporte de certificados reimpressos la usuaria MARÍA ISABEL BENAVIDEZ con el código No. 53547 aparece con muchos certificados reimpressos, al verificar los roles asignados a dicha usuaria no tienen relación con la opción de reimpresión de certificados, se anexa reporte y se requiere verificar porque aparece dicha usuaria sin tener los roles para poder hacer reimpresión de certificados.

Grupo Pasto- Glpi 20294

### DESARROLLO

Se debe revisar el reporte, ya que revisando en la base de datos no se encuentra tal inconsistencia.

Se modifica procedimiento SP\_REPORTE\_49

```
*****
/** AHERRENO 13/08/2010 */
/** Se elimina el campo id_usuario1 y se
/** agrega el campo id_usuario
*****
```

SELECT id\_turno, anio, id\_circulo, id\_proceso, trhs\_id\_fase, id\_turno\_historia, trhs\_fecha\_creacion, trhs\_fecha\_registro, trhs\_matricula\_impresa, id\_usuario

<h3 style="text-align: center;">211 Opcion boton devolver calificación</h3>	
<p><b>REQUERIMIENTO</b></p> <p>Turno:2010-228-6-292 Matricula(s):228-5507</p> <p>Motivo Inconsistencia: Se comunica la funcionaria Sandra Merino informando que el turno 2010-228-6-292 fue ingresado como testamento, la funcionaria se dio cuenta cuando el turno estaba en la fase de Mesa de control, se devolvió el turno a testamento de allí se devolvió a confrontación se cambio el subtipo de solicitud por normal, paso a reparto de registro, calificación se calificó el Folio 228-6379 se imprimió el formulario de calificación pero faltó calificar el folio 228-5507 por eso se tiene que devolver el turno a calificación para calificar el Folio 228-5507</p> <p>Por favor verificar el proceso del turno 2010-228-6-292 ya que en firma al devolver a calificación el sistema no le da la opción solo le permite ingresar la nota informativa y devolver a Testamento.</p> <p>Grupo Sitio Nuevo</p>	
<p><b>DESARROLLO</b></p> <p>Análisis: Este problema se presenta porque dentro de las fases del turno existe la fase de testamento y el sistema no tiene en cuenta que exista una fase que calificación.</p> <p>Solución: Se modificó el código para que tuviera en cuenta la última fase del turno, ya sea TESTAMENTO o CAL_CALIFICACION y así mostrar el botón correspondiente. Se modificó el siguiente archivo.</p>	
<p><b>CLASE o ARCHIVO JSP:</b></p> <p><b>\SIR-APP\web\WEB-INF\jspSeguro\registro\TURNO_REGISTRO_FIRMA_REGISTRADOR.jsp</b></p> <p><b>Línea 69:</b></p> <pre>     /**      * @author: Fernando Padilla      * @change: Mantis 5643: Acta - Requerimiento No 211 - Opcion boton devolver calificación      *           Se verifica si existe en la historia la fase CAL_CALIFICACION para poner en false      *           la variable isTestamento.      */     else if(temp.getFase() != null &amp;&amp; temp.getFase().equals(gov.sir.core.web.acciones.registro.AWMesa.CAL_CALIFICACION)){     isTestamento = false; } </pre>	
<h3 style="text-align: center;">214 Configuración en subtipos de atención calificadores</h3>	
<p><b>REQUERIMIENTO</b></p> <p>Comedidamente se solicita que los usuarios listados en los subtipos de atención para calificadores se deben visualizar con el nombre y apellido, para identificarlos, cuando existen funcionarios con el mismo nombre, Así mismo se aclara que los subtipos de atención para calificadores son " GRAN TAMAÑO, NORMAL, TESTAMENTOS, COPY, REGLAMENTO; diferentes a los mencionados no se deben reflejar en esta pantalla administrativa.</p>	

SNR :: Superintendencia de Notariado y Registro :: Ministerio del Interior y de Justicia :: Rep - Windows Internet Explorer  
 http://192.168.202.22:7778/SNR/ssp/partidas/report-displayer/administracion/hermod.dos

Superintendencia de Notariado y Registro  
 Ministerio del Interior y de Justicia  
 República de Colombia

Llamada o Orden  
 Versión 2.6.1  
 Fecha Actual: 19/8/2010

Sair

Pantalla Administrativa | Administración De Subtipos de Atención para Calificadores

Subtipos de Atención para Calificadores

	COPIA	GRAN TAMAÑO	NORMAL	PRUEBA 1	REGLAMENTO	TESTAMENTO	USUARIO.PRUEBAS
LUZ	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
LUZ	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ESPERANZA	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
DORY	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
LUZ MARINA	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SUSANA	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
GLORIA O	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
CLAUDIA	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
HERCILIA	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
MARIA EDI	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
MARIA ERI	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
MARTHA ISABEL	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BELCY	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
JOSE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
GLADYS	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ANA	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
LADYS	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Agregar | Volver

## DESARROLLO

Se modifica el siguiente archivo-

### ARCHIVO JSP:

WEB-INF.jspSeguro.administracion.  
 ABOGADOS\_SUBTIPO\_ATENCION\_HERMOD.jsp

#### Código Línea 147:

```
<td class="camposformtext_noCase"><%= dato.getNombre() %></td>
```

#### Por:

```
<!--
 * @author : Julio Alcázar Rivas
 * @change : Se concatena al nombre del calificador los dos apellidos para
mayor claridad
 * Caso Mantis : 05985
!-->
<td class="camposformtext_noCase"><%= dato.getNombre() + " "+
((dato.getApellido1()!= null)? dato.getApellido1(): " ") + " " +
((dato.getApellido2()!= null)? dato.getApellido2(): " ")%></td>
```

### 223 Error al separar varias matriculas en proceso de corrección (Informativo)

### REQUERIMIENTO

Requerimiento GLPI No 21436

Turno:2010-300-3-1328

Matricula(s):300-320718, 300-320719, 300-320720, 300-340003, 300-340004

Motivo Inconsistencia: Informa el funcionario Jose Luis Garcia que al intentar agregar la matricula 300-340004 como hija de la matricula 300-320718, el sistema notifica error.

El turno se encuentra en EJECUTAR CORRECCION SIMPLE X-EJEC.CORR.300, se verifica el proceso en línea con el usuario: al ingresar por la opción Matriz/Segregaciones, el usuario le da clic en Agregar y digita Número matricula: 300-340004, en Número anotación: ingresa la anotación 1 y en Anotación en este folio: ingresa la anotación 2 pero aparece error.

Favor verificar los datos de los folios asociados al turno.

### **DESARROLLO**

Por negocio el sistema solo deja modificar una sola matricula en la segregación, se recomienda realizar dos turnos de corrección para realizar estos cambios.

Dado lo anterior se da por cerrado este caso y se corrobora con la Funcionaria Marleny Alaguna en correo enviado el 17/09/2010 donde se especifica que es una mala operación de usuarios

### **226 Procedimiento SP\_REPORTE\_51 (Informativo)**

#### **REQUERIMIENTO**

El día 14 de septiembre se realizo el despliegue de la versión 002\_6\_2 del SIR en la cual se detecto que la actualización del procedimiento SP\_REPORTE\_51 ocasiono lentitud en la generación del reporte y en algunos casos la no generación del mismo.

#### **DESARROLLO**

Análisis de Requerimiento:

ANÁLISIS: Reporte no se genera o tarda mucho tiempo en generarse.

SOLUCIÓN: Se realiza afinamiento a los inner join y a las condiciones de la consulta principal.

#### **RESULTADOS PRUEBAS**

ambiente de pruebas:

Base de Datos snrdesa  
aplicación: 192.168.202.9

Se realizan pruebas de ejecución del reporte original(con el código que se encuentra actualmente en producción) y se obtienen los siguiente resultados:

datos pruebas:

fecha 13-05-2010

circulo 300

ejecución de la consulta solasqldeveloper : 67 segundos  
ejecución del reporte: 02:58.65 min

se realizan pruebas con el código adjunto (consulta afinada)  
ejecución de la consulta sola(sqldeveloper) : 0,39 segundos  
ejecución del reporte: 01:24.68

Script Bases de Datos: SP\_REPORTE\_51

NINGUNA.

Creación De Clases:

NINGUNA (el mismo formato de modificación clases o jsp)

Se modificaron los siguientes archivos:

Código Anterior:

```
SELECT
/*+ INDEX_DESC(SIR_OP_TURNO)*/
    sir_op_turno.id_solicitud,
    sir_op_turno.trno_anulado,
    sir_op_turno.trno_id_workflow,
    sir_op_turno.id_turno,
    sir_op_turno.anio,
    sir_op_turno.id_proceso,
    sir_op_turno.id_circulo
FROM sir_op_pago,
    sir_op_turno,
    sir_op_solicitud
WHERE sir_op_solicitud.id_solicitud      = sir_op_turno.id_solicitud
AND sir_op_solicitud.id_solicitud      = sir_op_pago.id_solicitud
AND (sir_op_pago.pago_fecha            >= TRUNC(TO_DATE(p_dia, 'YYYY-MM-DD'), 'DD')
AND sir_op_pago.pago_fecha            < TRUNC(TO_DATE(p_dia, 'YYYY-MM-DD'), 'DD') + 1
AND TRUNC(TO_DATE(p_dia, 'YYYY-MM-DD'), 'DD') = TO_DATE(p_dia, 'YYYY-MM-DD')
OR TRUNC(sir_op_turno.trno_fecha_fin) = TO_DATE(p_dia, 'YYYY-MM-DD')
AND sir_op_turno.id_proceso          = '4')
AND (sir_op_turno.id_circulo          = p_circulo
OR sir_op_solicitud.id_circulo       = p_circulo
AND (sir_op_turno.id_proceso         = '1'
OR sir_op_turno.id_proceso          = '2'
OR sir_op_turno.id_proceso          = '5'
OR sir_op_turno.id_proceso          = '6'
OR sir_op_turno.id_proceso          = '20'
OR sir_op_turno.id_proceso          = '4')
ORDER BY sir_op_pago.id_usuario, sir_op_pago.pago_fecha,
    sir_op_turno.trno_id_workflow ;
```

Código Nuevo

```
SELECT sir_op_pago.id_usuario,
    /******
    ** AHERRERO */
    /* 26/05/2010 */
    sir_op_pago.pago_fecha,
    /******
    sir_op_turno.trno_id_workflow,
    /*+ INDEX_DESC(SIR_OP_TURNO)*/
    sir_op_turno.id_solicitud,
    sir_op_turno.trno_anulado,
    sir_op_turno.id_turno,
    sir_op_turno.anio,
    sir_op_turno.id_proceso,
    sir_op_turno.id_circulo,
    /******
    /* AHERRENO 26/05/2010
Se agrega subquery para consultar la fecha en que se realizo la anulación*/
    (SELECT MAX(H.TRHS_FECHA_REGISTRO)
        FROM sir_op_turno_historia h
        WHERE sir_op_turno.id_circulo = h.id_circulo
        AND sir_op_turno.anio      = h.anio
        AND sir_op_turno.id_proceso = h.id_proceso
        AND sir_op_turno.id_turno   = h.id_turno
    AND sir_op_turno.last_id_historia = h.id_turno_historia
        AND h.trhs_id_fase      = 'ANULADO') fecha_anulacion
/*GCABRERA 21-09-2010
SE MODIFICA LAS CONDICIONES PARA OPTIMIZAR LA CONSULTA
SE BAJÓ LA CONSULTA DE 67 segundos(sin la union que es del nuevo requerimiento a 0,39 segundos (con la
union)*/
    FROM sir_op_solicitud inner join sir_op_turno on sir_op_solicitud.id_solicitud = sir_op_turno.id_solicitud
    inner join sir_op_pago on sir_op_solicitud.id_solicitud = sir_op_pago.id_solicitud
    WHERE (sir_op_solicitud.id_circulo = p_circulo
        OR sir_op_turno.id_circulo = p_circulo)
        AND sir_op_pago.id_circulo = p_circulo
    and to_char(sir_op_turno.TRNO_FECHA_INICIO,'YYYY')=TO_CHAR(sir_op_pago.pago_fecha,'YYYY')
    AND sir_op_turno.id_proceso in ('1','2','5','6','20','4')
    AND sir_op_pago.pago_fecha >= TRUNC(TO_DATE(p_dia, 'YYYY-MM-DD'), 'DD')
    AND sir_op_pago.pago_fecha < TRUNC(TO_DATE(p_dia, 'YYYY-MM-DD'), 'DD') + 1
    --AND TRUNC(TO_DATE(:p_dia, 'YYYY-MM-DD'), 'DD') = TO_DATE(:p_dia, 'YYYY-MM-DD')
    --or TRUNC(sir_op_turno.trno_fecha_fin)      = TO_DATE(:p_dia, 'YYYY-MM-DD')
    --and sir_op_turno.id_proceso                = '4')
    /******
```

```

/** AHERRENO */
/** 26/05/2010 */
UNION
SELECT sir_op_pago.id_usuario,
sir_op_pago.pago_fecha,
sir_op_turno.trno_id_workflow,
/*+ INDEX_DESC(SIR_OP_TURNO) */
sir_op_turno.id_solicitud,
sir_op_turno.trno_anulado,
sir_op_turno.id_turno,
sir_op_turno.anio,
sir_op_turno.id_proceso,
sir_op_turno.id_circulo,
(SELECT MAX(H.TRHS_FECHA_REGISTRO)
FROM sir_op_turno_historia h
WHERE sir_op_turno.id_circulo = h.id_circulo
AND sir_op_turno.anio      = h.anio
AND sir_op_turno.id_proceso = h.id_proceso
AND sir_op_turno.id_turno   = h.id_turno
AND sir_op_turno.trno_last_id_historia = h.id_turno_historia
AND h.trhs_id_fase        = 'ANULADO')fecha_anulacion
FROM sir_op_pago,
sir_op_turno,
sir_op_solicitud,
sir_op_turno_historia h
WHERE sir_op_solicitud.id_solicitud      = sir_op_turno.id_solicitud
AND sir_op_solicitud.id_solicitud      = sir_op_pago.id_solicitud
AND sir_op_turno.trno_last_id_historia   = h.id_turno_historia
AND sir_op_turno.id_circulo            = h.id_circulo
AND sir_op_turno.anio                  = h.anio
AND sir_op_turno.id_proceso           = h.id_proceso
AND sir_op_turno.id_turno             = h.id_turno
AND h.trhs_id_fase                  = 'ANULADO'
AND TO_DATE(sir_op_pago.pago_fecha, 'YYYY-MM-DD') <> TO_DATE(h.trhs_fecha_registro, 'YYYY-MM-DD')
AND (h.trhs_fecha_registro          >= TRUNC(TO_DATE(p_dia, 'YYYY-MM-DD'), 'DD')
AND h.trhs_fecha_registro          < TRUNC(TO_DATE(p_dia, 'YYYY-MM-DD'), 'DD') + 1)
AND (sir_op_turno.id_circulo       = p_circulo
OR sir_op_solicitud.id_circulo    = p_circulo)
and to_char(sir_op_turno.TRNO_FECHA_INICIO,'YYYY')=TO_CHAR(sir_op_pago.pago_fecha,'YYYY')
AND sir_op_turno.id_proceso in ('1','2','5','6','20','4')
ORDER BY 1, 2,3;

```

Se ejecuta actualización procedimiento el día 21/09/2010 en horas de la noche. Se realizaron pruebas con fecha 13/09/2010, 13/05/2010 y 14/05/2010 en las ORIPS de Bucaramanga, piedecuesta, neiva entre otras ORIPS, en la ejecución se observó un buen tiempo en el despliegue del reporte, se da por aprobada la actualización y se deja el cambio en base de datos de producción. Se reporta solución en el acta.

## 228 Error en notas devolutivas

### **REQUERIMIENTO**

Requerimiento Glpi No 21917

Motivo Inconsistencia: Nos comunicamos con el funcionario Rodrigo Flórez quien nos informa que la nota devolutiva de inscripción parcial está generando el numero de matrícula sobreescrito sobre el comentario y matrícula inscrita parcial (Ver error)

por favor verificar por que cuando se genera la impresión de la nota devolutiva de un inscripción parcial genera la matrícula sobreescrita sobre el comentario y el turno de certificado asociado se genera en otro renglón entrecortado. (Ver imagen subrayado)

Grupo Manizales



## OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS DE MANIZALES

## NOTA DEVOLUTIVA

ID - 21917

Página: 1

Impreso el 22 de Septiembre de 2010 a las 08:01:18 am

El documento OFICIO Nro 1423 del 26-08-2010 de JUZGADO NOVENO CIVIL MUNICIPAL de MANIZALES - CALDAS fue presentado para su inscripción como solicitud de registro de documentos con Radicacion:2010-100-6-19763 vinculado a la Matricula Inmobiliaria:

y matricula inscrita parcialmente 62668

Y CERTIFICADOS ASOCIADOS: 2010-100-1-66578

Conforme con el principio de Legalidad Registral previsto por el artículo 37 del Decreto Ley 1250 de 1970 (Estatuto del Registro de Instrumentos Públicos), se rechaza sin registrar el citado documento por las siguientes razones y fundamentos de derecho:

1: SOBRE EL BIEN INMUEBLE OBJETO DE EMBARGO SE ENCUENTRA VIGENTE AFECTACIÓN A VIVIENDA FAMILIAR (ARTÍCULO 7 LEY 258 DE 1996).(400 400)

**DESARROLLO**

Se resuelve incidente se corrige clase  
gov.sir.core.negocio.modelo.imprimibles.ImprimibleNotaDevolutivaCalificacion  
que genera el imprimible...

**LINEA 120****ANTES**

```
this.imprimirLinea(ImprimibleConstantes.TITULO2,ImprimibleConstantes.MARGEN_IZQ, t,130);
```

**POR**

```
this.imprimirLinea(ImprimibleConstantes.TITULO2,t,true);
```

**LINEA 151**

```
/**
```

```
* @author Fernando Padilla
```

```
* Se modifica segun caso mantis: 6252: error en nota devolutiva
```

```
*/
```

```
if(lineas.size()!=0){
```

```
    this.imprimirLinea(ImprimibleConstantes.TITULO2,ImprimibleConstantes.MARGEN_IZQ,fraseMatriculas + line,true);
```

```
}else{
```

```
    this.imprimirLinea(ImprimibleConstantes.TITULO2,ImprimibleConstantes.MARGEN_IZQ,fraseMatriculas + line,true);
```

```
}
```

**233 Error al liquidar el valor del acto****REQUERIMIENTO**

Turno: 2010-999-6-644

Motivo Inconsistencia: El sistema no está liquidando el valor del nuevo acto ingresado en la nueva entrada.

Se realizaron pruebas funcionales y se encontró que al generar una segunda nueva entrada con un acto igual al de los turnos anteriores, para este caso el Acto Venta, el sistema está liquidando valor 0 cuando debería cobrar los derechos de registro por un valor de 37.040. El turno actual de nueva entrada 2010-999-6-644 se encuentra en CONFRONTAR X-CONFRONTADOR.999 y aparece valor pagado 0, al revisar los turnos anteriores observamos que se ingresaron los actos:

VENTA Cuantía: 8,000,000 Valor: 40,000 con el turno 2010-999-6-642 y ACLARACION Cuantía: 1 Valor: 14,280 con el turno 2010-999-6-643, los cuales fueron liquidados correctamente.

Con la actual nueva entrada 2010-999-6-644 se está ingresando el acto VENTA Cuantía: 7,408,000 Valor: 37,040 pero el sistema está liquidando valor 0, favor verificar ya que el

sistema al ingresar una segunda nueva entrada con un acto igual al de turnos anteriores está liquidando valor 0.

Grupo Snr

## DESARROLLO

### CLASE o ARCHIVO JSP:

### gov.sir.hermod.pagos.LiquidadorRegistro.java

#### Línea 716:

```
/**  
 * @author fernando padilla velez  
 * @change Mantis 6716: Error al liquidar el valor del acto,  
 * Se realiza refactory en el metodo de calcularValores, ya que,  
 * existe un bug cuando se realiza una nueva entrada y se agrega  
 * un acto que ya existia en el turno anterior. por ejemplo, dos ventas,  
 * el sistema dejaba los valores en ceros no ocasionando costos en la liquidacion  
 * para este nuevo acto.  
 */  
private void calcularValores(SolicitudRegistro s, LiquidacionTurnoRegistro liqReg, List actos)  
throws LiquidarException {  
  
    try  
    {  
        Turno turnoAnt = s.getTurnoAnterior();  
        HermodDAOFactory factory = HermodDAOFactory.getFactory();  
  
        if (turnoAnt != null)  
        {  
            TurnoPk tId = new TurnoPk();  
            tId.anio = turnoAnt.getAnio();  
            tId.idCirculo = turnoAnt.getIdCirculo();  
            tId.idProceso = turnoAnt.getIdProceso();  
            tId.idTurno = turnoAnt.getIdTurno();  
            /**  
             * @author Fernando Padilla Velez  
             * @change Acta - Requerimiento 071- Interes_Mora_Impuesto, Se valida que el  
             * el circulo tenga cobros de impuesto, esto para no calcular las multas  
             * de estos impuestos.  
             */  
            if(turnoAnt.getSolicitud().getCirculo().isCobroImpuesto()){  
                obtenerValorMultasActos(actos, s ,true, turnoAnt.getFechaInicio());  
            }  
            liqReg.setValorDerechos(valorTotalDerechos);  
            liqReg.setValorImpuestos(valorTotalImpuestos);  
            liqReg.setValorMora(valorTotalMultas);  
  
            List actosCalcularComp = new ArrayList(actos);  
  
            /**  
             * se incluye algoritmo de complejidad O3. Es necesario incluirlo  
             * pues la lógica obliga a iterar sobre sobre liquidaciones y hacer  
             * comparaciones entre actos de liquidaciones.  
             *  
             * El volumen de operaciones es bajo por tanto, no tendra impacto  
             * en desempeño  
             */  
            Turno t = factory.getTurnosDAO().getTurnoByWFIId(turnoAnt.getIdWorkflow());  
  
            tId = new TurnoPk();  
            tId.anio = t.getAnio();  
            tId.idCirculo = t.getIdCirculo();  
            tId.idProceso = t.getIdProceso();  
            tId.idTurno = t.getIdTurno();  
  
            if (t.getSolicitud() != null  
                && t.getSolicitud().getLiquidaciones() != null  
                && !t.getSolicitud().getLiquidaciones().isEmpty()){  
                LiquidacionTurnoRegistro liqAnterior =
```

```

(LiquidacionTurnoRegistro)t.getSolicitud().getLiquidaciones().get(0);

    if (liqAnterior.getActos() != null){

        for (Iterator actosAntItera = liqAnterior.getActos().iterator();
             actosAntItera.hasNext();){
            Acto actoAnterior = (Acto)actosAntItera.next();
            liqReg.setValorDerechos(liqReg.getValorDerechos()-
actoAnterior.getValor());
            liqReg.setValorImpuestos(liqReg.getValorImpuestos() -
actoAnterior.getValorImpuestos());
        }

        //se valida que los valores no sean negativos
        if (liqReg.getValor()< 0) {
            liqReg.setValor(0);
        }
        if (liqReg.getValorDerechos()< 0) {
            liqReg.setValorDerechos(0);
        }
        if (liqReg.getValorImpuestos()< 0) {
            liqReg.setValorImpuestos(0);
        }
        if (liqReg.getValorMora()< 0) {
            liqReg.setValorMora(0);
        }
        if (liqReg.getValorOtroImp()< 0) {
            liqReg.setValorOtroImp(0);
        }
    }

    double valorOtroImpAnt =
factory.getLiquidacionesDAO().getValorOtroImpuestoByTurno(tId);

    liqReg.setValorOtroImp(liqReg.getValorOtroImp() -
(float)valorOtroImpAnt);

    if (liqReg.getValorOtroImp()< 0) {
        liqReg.setValorOtroImp(0);
    }

    liqReg.setValorMora(valorTotalMultas);

    if (liqReg.getValorMora()< 0) {
        liqReg.setValorMora(0);
    }

    liqReg.setValor(liqReg.getValorDerechos() +
liqReg.getValorImpuestos() + liqReg.getValorMora() + liqReg.getValorOtroImp());

    if (liqReg.getValor()< 0) {
        liqReg.setValor(0);
    }

} else{
    obtenerValorMultasActos(actos,s,false,null);

    double valorLiquidado = valorFinalActos + valorTotalMultas +
liqReg.getValorOtroImp();
    valorLiquidado = roundValue (valorLiquidado);

    liqReg.setValor(valorLiquidado);
    liqReg.setValorDerechos(valorTotalDerechos);
    liqReg.setValorImpuestos(valorTotalImpuestos);
    liqReg.setValorMora(valorTotalMultas);
    liqReg.setValorOtroImp(liqReg.getValorOtroImp());
}

} catch (Exception e) {
    Log.getInstance().error(LiquidadorRegistro.class,e.getMessage(),e);
    throw new LiquidarException("No se pudo obtener el valor de la liquidacion",

```

```

        e);
    } catch (Throwable e) {
        Log.getInstance().error(LiquidadorRegistro.class,e.getMessage(),e);
        throw new LiquidarException("No se pudo obtener el valor de la liquidacion",
        e);
    }
}

```

## DESPLIEGUE – 2.6.4

10/11/2010

### 231 Error Proceso Correcciones - Fase Aprobar (Informativo)

#### **REQUERIMIENTO**

Turno:2010-260-3-2385

Matricula(s):260-179622

Motivo Inconsistencia: Se recibe fax informando que al avanzar el turno 260-179622 de la fase de revisar y aprobar le genera el siguiente error "No se puede imprimir el formulario no se han realizado correcciones"

Por favor verificar error anexo. Se realizaron correcciones en la el numero de la anotación cancelada.

Grupo Cúcuta

Esta inconsistencia se han reportado en los siguiente casos: 22350, 22352, 22389, 22410, 22411, 22433

Se da prioridad a la solución de este caso ya que se está presentando desde el despliegue 002\_6\_2.

#### **DESARROLLO**

##### Análisis de Requerimiento:

- Script Bases de Datos:

NINGUNA

- Creación De Clases:

NINGUNA

Se modificaron los siguientes archivos:

#### CLASE o ARCHIVO JSP

**gov.sir.core.web.acciones.correccion.AWCorreccion.java**

##### **Línea 615:**

```

/**
 * Mantis 6457: Acta - Requerimiento No 231 - Error Proceso Correcciones - Fase Aprobar,
 * Se realiza refactory en el metodo de doCorr_RevisionAprobacion_PrintForm, ya que,
 * existe un bug cuando se realiza la validacion de cambios realizados sobre los folios
 * asociados al turno. Se cambia esta validacion usando la libreria GeneralSIR para
 * realizar esta validacion.
 * @author fernando padilla velez

```

```

/*
private EvnCorreccion doCorr_RevisionAprobacion_PrintForm( HttpServletRequest request,
boolean impresionTemporal ) throws AccionWebException {

    HttpSession session = request.getSession();

    org.auriga.core.modelo.transferObjects.Usuario usuarioAuriga =
    (org.auriga.core.modelo.transferObjects.Usuario)
    request.getSession().getAttribute(SMARTKeys.USUARIO_EN_SESION);
    Turno turno = (Turno) request.getSession().getAttribute(WebKeys.TURNO);
    Fase fase = (Fase) request.getSession().getAttribute(WebKeys.FASE);

    String accionWf = "";
    ValidacionParametrosException exception = new ValidacionParametrosException();

    try {
        ValidacionesSIR validacionesSIR = new ValidacionesSIR();

        if(!validacionesSIR.isTurnoMatriculasTemporales(turno.getIdWorkflow())){
            exception.addError("No se puede imprimir el formulario porque no se han realizado
correcciones." );
            throw exception;
        }

    } catch (GeneralSIRException ex) {
        Logger.getLogger(AWCorreccion.class.getName()).log(Level.SEVERE, null, ex);
        exception.addError("No se puede imprimir el formulario porque se genera "
                + "un error al tartar de validar el turno: " + ex.getMessage());
        throw exception;
    }

    int copias= 1;

    if(fase!=null && fase.getID() !=null && fase.getID().equals(CFase.COR_REVISAR_APROBAR))
    {
        copias=2;
    } else {

        String valCopias = request.getParameter(WebKeys.NUMERO_COPIAS_IMPRESION);
        if( valCopias != null ){
            try{
                copias = Integer.valueOf(valCopias).intValue();
                if(copias < 1){
                    exception.addError("El número de copias es inválido");
                }
            }
            catch( Exception e ){
                exception.addError("El número de copias es inválido");
            }
        }
    }

    // raise if needed -----
    if( exception.getErrores().size() > 0 ) {
        throw exception;
    }
    // -----

    EvnCorreccion evn = new EvnCorreccion(usuarioAuriga,
    (gov.sir.core.negocio.modelo.Usuario) request.getSession().getAttribute(WebKeys.USUARIO),
    turno, fase, accion, accionWf);
    // evn.setEjecutadaCorreccionSimple( true );

    Circulo local_Circulo = (Circulo) session.getAttribute( WebKeys.CIRCULO );

    String local_CirculoId = local_Circulo.getIdCirculo();
    evn.setUID(request.getSession().getId());
    // cambiado: se usa la impresion por la
}

```

```

    // impresora laser; se mantiene
    // sin embargo el id de sesion
    evn.setCirculoId( local_CirculoId );

    evn.setNumeroCopias(copias);
    evn.setImpresionTemporalDeAuxiliarEnabled( impresionTemporal );

    return evn;

} // end-method: doCorr_RevisionAprobacion_PrintForm

```

**Se agrega la siguiente librería al jboss:**

La librería se encuentra en /WORK/LIB/GeneralSIR.jar y se debe colocar en la ruta del jboss /JBoss\_HOME/server/default/lib/

**Se debe modificar el archivo oracle-xa-ds.xml del jboss:**

**Se debe modificar este archivo .xml alojado en la ruta  
/JBoss\_HOME/server/default/deploy/ y se debe agregar un nuevo  
datasource como sigue:**

```

<xa-datasource>
    <jndi-name>jdbc/generalPool</jndi-name>
    <track-connection-by-tx/>
    <isSameRM-override-value>false</isSameRM-override-value>
    <xa-datasource-class>oracle.jdbc.xa.client.OracleXADatasource</xa-datasource-class>
    <xa-datasource-property
        name="URL">jdbc:oracle:thin:@(DESCRIPTION_LIST=(LOAD_BALANCE=yes)(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.200.9)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.200.7)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.200.8)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=snr.supernotariado.gov.co))</xa-datasource-property>
    <xa-datasource-property name="User">sir</xa-datasource-property>
    <xa-datasource-property name="Password">dj7810kl</xa-datasource-property>
    <exception-sorter-class-
        name>org.jboss.resource.adapter.jdbc.vendor.OracleExceptionSorter</exception-sorter-class-
        name>
    <no-tx-separate-pools/>
    <metadata>
        <type-mapping>Oracle9i</type-mapping>
    </metadata>
</xa-datasource>

```

## Creación Proyecto GeneralSIR

Script Bases de Datos:

```
\WORK\Documentos\DESARROLLO\OBJETOS BASE DE
DATOS\SIR\PAQUETES\PCK_UTILIDADES_VALIDACIONES.pks
```

```
\WORK\Documentos\DESARROLLO\OBJETOS BASE DE
DATOS\SIR\PAQUETES\PCK_UTILIDADES_VALIDACIONES.pkb
```

Creación De Clases:

```

co.com.iridium.generalSIR.comun.ConstantesSIR.java
co.com.iridium.generalSIR.comun.GeneralSIRException.java
co.com.iridium.generalSIR.comun.PropertiesSIR.java
co.com.iridium.generalSIR.comun.properties.GeneralSIR.Properties
co.com.iridium.generalSIR.dao.FabricaDAO.java

```

co.com.iridium.generalSIR.dao.ValidacionesSIRDAO.java  
co.com.iridium.generalSIR.dao.datasource.DataSourceSIR.java  
co.com.iridium.generalSIR.negocio.validaciones.ValidacionesSIR.java

## DESPLIEGUE – 2.6.3

21/09/2010

### 222 Error al reasignar en fase de calificación (Informativo)

#### REQUERIMIENTO

Requerimiento GLPI No 21749

Turno:2010-300-6-39648

Motivo Inconsistencia:

El usuario José Luis García nos informa que el turno 2010-300-6-39648 inicialmente se le había asignado en calificación al usuario CLAUDIA.RUEDA, después se le reasignó al usuario SUSANA.AWAD quien fue la que hizo la respectiva calificación y lo envió a la fase de DIGITACION, y cuando se lo fueron a devolver a SUSANA el turno retorno donde CLAUDIA.RUEDA. Quien fue el usuario que no efectuó la calificación.

Se requiere verificar ya que el turno debe llegar a la usuaria que realizó la calificación.

#### DESARROLLO

- Script Bases de Datos:
  - NINGUNA
- Creación De Clases:
  - NINGUNA
- Se modificaron los siguientes archivos:

<b>CLASE:</b> <b>gov.sir.hermod.dao.impl.jdogenie.JDOTurnosNuevosDAO.java</b>
<b>Método:</b>
<b>private void actualizarInfoTurnoEjecucion (Hashtable paramsRespuesta, String estacionAsignada, Turno turno, Usuario usuarioSir, PersistenceManager pm) throws DAOException</b>

#### Código Línea 1166:

```
boolean flag = true;
        while(itr.hasNext() && flag){
            itrTurno = (TurnoHistoria) itr.next();
            if(CFase.COR_REVISION_ANALISIS.equals(itrTurno.getFase())){
                turnoEjecucionEnh.setEstacion(itrTurno.getIdAdministradorSAS());
                estacionAsignada=itrTurno.getIdAdministradorSAS();
                flag = false;
            }
        }
```

**Por:**

```

/*
 * @author : Julio Alcazar
 * @change : se actualiza el codigo que retorna el turno al usuario que lo envió a
Solicitar información de
folios a otras dependencias (ANTIGUO SISTEMA).
Teniendo en cuenta si el turno fue reasignado antes de
iniciar el proceso de revision y analisis
CFase.COR_REVISION_ANALISIS.
* Caso Mantis : 0006176
*/
while(itr.hasNext() ){
    itrTurno = (TurnoHistoria) itr.next();
    if((CFase.COR_REVISION_ANALISIS.equals(itrTurno.getFase()) ||
(CFase.COR_REVISION_ANALISIS.equals(itrTurno.getFase()) &&
CFase.COR_REVISION_ANALISIS.equals(itrTurno.getFaseAnterior()))){

turnoEjecucionEnh.setEstacion(itrTurno.getIdAdministradorSAS());
estacionAsignada=itrTurno.getIdAdministradorSAS();
}

```

#### Método:

```

private void actualizarInfoTurnoEjecución (Hashtable paramsRespuesta, String
estacionAsignada, Turno turno, Usuario usuarioSir, PersistenceManager pm) throws
DAOException

```

#### Código Insertado Línea 1723:

```

/*
 * @author : Julio Alcazar
 * @change : variable utilizada para guardar un historial temporal
 * Caso Mantis : 0006176
*/
TurnoHistoriaEnhanced stackPos2=null;

```

#### Código Línea 1729:

```

if(CFase.CAL_CALIFICACION.equals(turnoHist.getFase()) &&
CFase.REG_REPARTO.equals(turnoHist.getFaseAnterior())){
    stackPos = turnoHist;
}

```

#### Por:

```

/*
 * @author : Julio Alcazar
 * @change : se actualiza el codigo que retorna al usuario los turnos de registro que
han sido enviado a Digitacion masiva
una vez estos finalicen el proceso. Teniendo en cuenta si el turno fue reasignado antes de
iniciar el proceso Calificacion CFase.CAL_CALIFICACION.
* Caso Mantis : 0006176
*/
if((CFase.CAL_CALIFICACION.equals(turnoHist.getFase()) &&
CFase.REG_REPARTO.equals(turnoHist.getFaseAnterior()) ||
(CFase.CAL_CALIFICACION.equals(turnoHist.getFase()) &&
CFase.CAL_CALIFICACION.equals(turnoHist.getFaseAnterior())))
{
    stackPos2 = turnoHist;
}

```

#### Código Insertado Línea 1761:

```

/*
 * @author : Henry Gómez Rocha
 * @change : se asigna la variable temporal stackPos2 a stackPos en el caso qe sea
diferente de null.
 * Caso Mantis : 0002484
*/
if (stackPos2 != null){
    stackPos = stackPos2;
}

```

### 225 Proceso Correcciones – SGD (Informativo)

#### **REQUERIMIENTO**

#### **Correo 15/09/2010**

Los turnos de corrección dados en el día de hoy en el sistema SIR no están entrando a la malla del sistema IRIS, todos los demás turnos si han llegado, por favor revisar que sucede, estos son los turnos 2010-300-3-1402 al 2010-300-3-1407. Se necesita escanear la

solicitud de la corrección para inicializar el proceso.

### **DESARROLLO**

Análisis: Se encontró error NullPointerException al tratar de realizar numeroRecibo = new Long(pago.getNumRecibo());

Solución: se corrige de la siguiente forma:

clase: gov.sir.hermod.gdocumental.integracion.PublisherIntegracion

Método: private void addTurno()

Línea: 235:

numeroRecibo = new Long(pago.getNumRecibo() != null ? pago.getNumRecibo() : "0");

## **DESPLIEGUE – 2.6.2**

**15/09/2010**

### **056 Bloqueo de turnos**

**(Informativo)**

#### **REQUERIMIENTO**

Bloqueo de Turnos.

Se requiere tratar la problemática para evitar el bloqueo de los turnos que se encuentren bloqueados por otros turnos que se encuentren en fase de entrega o finalizados, como ocurre con los turnos de correcciones que se encuentran en la fase entrega o que ya están finalizados y su folio asociado se encuentra bloqueado por este turno.

### **DESARROLLO**

#### **Análisis de Requerimiento:**

Los problemas que se han presentados por bloqueos de turno se han venido solucionando de acuerdo a la particularidad de cada uno de ellos por ejemplo en cuanto a bloqueo de turnos entre registro y corrección (este último posee prioridad), se le dio la solución general en el Acta - Requerimiento 029 - Bloqueos y Desbloqueos de Turno caso Mantis 03384 (Despliegue 2.3.1).

Actualmente se está trabajando en el bloqueo generados por datos temporales en Acta - Requerimiento No 94 - Proceso Calificación - Envío turno a proceso de correcciones y Acta - Requerimiento 136 Anotaciones Temporales Folios (Despliegue actual 2.6.2).

Se determina cerrar este caso dado que esta particularizado en los requerimientos antes mencionados.

### **057 Bloqueo de Folios**

**(Informativo)**

#### **REQUERIMIENTO**

El impacto que causa este desarrollo beneficia y evita las diferentes llamadas a la mesa de ayuda para reportar casos en donde el folio se encuentra bloqueado por un usuario diferente y no permite el avance del turno, como ejemplo los turnos que se encuentran en la fase firma, estos turnos no avanzan de fase porque el bloqueo del turno quedó con la estación del usuario que lo calificó que es diferente al usuario que tiene el rol registrador.

También permitirá eliminar estos bloqueos cuando los turnos de correcciones o registro no se pueden tomar debido a que otro usuario tiene el bloqueo del turno y este no ha generado ningún dato temporal.

## **DESARROLLO**

### **Análisis de Requerimiento:**

Debido a las sucesivas modificaciones al sistema (soluciones dadas a múltiples casos) ha disminuido este problema ya que acorde al cronograma otros problemas tienen mayor prioridad para SIR; se ha tomado la decisión de dar por cerrado este caso ya que no amerita invertir tiempo de desarrollo en él, no obstante, queda abierto el proceso de desarrollo para casos puntuales y precisos que en ultimas repercuten en la disminución del impacto de este caso en concreto.

## **094 Proceso Calificación - Envio turno a proceso de correcciones**

### **REQUERIMIENTO**

En el proceso de calificación al momento de enviar el turno a proceso de correcciones el sistema no permite avanzar el turno y muestra mensaje de error: “Existen anotaciones temporales” el aplicativo presenta esta inconsistencia para los folios que han tenido modificaciones con el proceso de corrección y que luego son asociados a turnos de registro.

## **DESARROLLO**

Después de un análisis minucioso se encontró que dependiendo del tipo de modificación que se le hiciese al folio en el rol Auxiliar de Correcciones y una vez aprobadas dichas correcciones en el rol Responsable de Correcciones, el sistema estaba dejando información en alguna de las siguientes tablas SIR\_NE\_ANOTA\_CIUDADANO\_TMP, SIR\_ANOTACION\_TMP,

SIR\_NE\_DIRECCION\_TMP,

SIR\_NE\_FOLIO\_DATOS\_TMP,

SIR\_NE\_FOLIO\_DERIVADO\_TMP,

SIR\_NE\_REL\_CANCELACION\_TMP,

SIR\_NE\_SALVEDAD\_AN\_TMP

y SIR\_NE\_SALVEDAD\_FO\_TMP.

Para evitar esto, se hicieron las modificaciones respectivas al código; consistiendo básicamente en la verificación de la información que se almacenaba en las tablas anteriormente mencionadas, debido a que el borrado de las tablas está codificado para que las tablas reúnan cierta cantidad de campos con información válida dado que de faltar algún campo o atributo el borrado no es ejecutado satisfactoriamente.

En ese orden de ideas, cabe anotar que en el caso de que todas las tablas posean información sobre un turno y folio específico se hace necesario borrar los datos de las tablas en el orden inverso en el que se detallaron anteriormente, esto debido a que si no se ejecuta el borrado en ese orden el sistema arroja una excepción dado que no se respeta la integridad referencial del modelo de base de datos del Sistema SIR.

Así mismo, se encontró durante la revisión del caso y se decidió comentar el código que no presta ningún servicio al sistema o en su defecto es redundante puesto que se encuentra duplicado. Todo esto con el fin de optimizar el rendimiento de SIR dado que se eliminan preguntas y ejecución de código innecesario.

La prueba del caso se hizo con la asesoría de Javier Rocha, Jefe Mesa de Ayuda.

- Script Bases de Datos:

- NINGUNA
- Creación De Clases:
- NINGUNA

<b>CLASE o ARCHIVO JSP:</b>
C:\WORK\SIR-APP\src\gov\sir\forseti\dao\impl\jdogenie\JDOGenieFolioDAO.java
Método:
<b>protected boolean updateFolio(FolioEnhanced folio, Folio datos, UsuarioEnhanced usuario, TurnoEnhanced turno, boolean validarTurno, PersistenceManager pm)</b>

**Código Línea 11616:**

```

if ((datos.getLindero() != null) ||
    (datos.getCodCatastral() != null) ||
    (datos.getCodCatastralAnterior() != null) ||
    (datos.getTipoPredio() != null) ||
    (datos.getRadicacion() != null) // bug 03580
    (datos.getFechaApertura() != null) // bug 03580
    (datos.getZonaRegistral() != null) // bug 03930
    (datos.getEstado() != null)) {
    this.updateFolioDatos(folio, datos, turno, validarTurno, pm);
}

```

**Por:**

```

/*
 * @author : Henry Gómez Rocha
 * @change : Se verifica que las variables tipo string no vengan con cadena vacía.
 * Caso Mantis : 0004503
 */

if ((datos.getLindero() != null) ||
    (datos.getCodCatastral() != null && !datos.getCodCatastral().isEmpty()) ||
    (datos.getCodCatastralAnterior() != null &&
    !datos.getCodCatastralAnterior().isEmpty()) ||
    (datos.getTipoPredio() != null) ||
    (datos.getRadicacion() != null && !datos.getRadicacion().isEmpty()) ||
    // bug 03580
    (datos.getFechaApertura() != null) // bug 03580
    (datos.getZonaRegistral() != null) // bug 03930
    (datos.getEstado() != null)) {
    this.updateFolioDatos(folio, datos, turno, validarTurno, pm);
}

```

Método:
<b>protected void hacerDefinitivoFolioDatos(FolioEnhanced folio, FolioDatosTMP datos, PersistenceManager pm, TurnoEnhanced turno, boolean validarTurno, UsuarioEnhanced usuario)</b>

**Código Línea 15367:**

```

if (datos.getLindero() != null) {
    set.add(datos.getNumRadicaLindero());

    if(datos.getNumRadicaLindero()== null ||
       datos.getNumRadicaLindero().equals(turno.getIdWorkflow())){
        folio.setLindero(datos.getLindero());
        datos.setLindero(null);
        datos.setNumRadicaLindero(null);
    }
}

if (datos.getCodCatastral() != null) {
    set.add(datos.getNumRadicaCodCatastral());

    if(datos.getNumRadicaCodCatastral() == null ||
       datos.getNumRadicaCodCatastral().equals(turno.getIdWorkflow())){
        folio.setCodCatastral(datos.getCodCatastral());
        datos.setCodCatastral(null);
    }
}

```

```

                datos.setNumRadicaCodCatastral(null);
            }

            if (datos.getCodCatastralAnterior() != null) {
                set.add(datos.getNumRadicaCodCatastralAnterior());

                if(datos.getNumRadicaCodCatastralAnterior() == null ||
                   datos.getNumRadicaCodCatastralAnterior().equals(turno.getIdWorkflow())){
                    folio.setCodCatastralAnterior(datos.getCodCatastralAnterior());
                    datos.setCodCatastralAnterior(null);
                    datos.setCodCatastralAnterior(null);
                }
            }

            // bug: 3580
            if (datos.getRadicacion() != null) {
                set.add(datos.getNumRadicaRadicacion());

                if(datos.getNumRadicaRadicacion()== null ||
                   datos.getNumRadicaRadicacion().equals(turno.getIdWorkflow())){
                    folio.setRadicacion( datos.getRadicacion() );
                    datos.setRadicacion(null);
                    datos.setNumRadicaRadicacion(null);
                }
            }

            if (datos.getFechaApertura() != null) {
                set.add(datos.getNumRadicaFechaApertura());

                if(datos.getNumRadicaFechaApertura()== null ||
                   datos.getNumRadicaFechaApertura().equals(turno.getIdWorkflow())){
                    folio.setFechaApertura( datos.getFechaApertura() );
                    datos.setFechaApertura(null);
                    datos.setNumRadicaFechaApertura(null);
                }
            }

            if (datos.getEstado() != null) {
                set.add(datos.getNumRadicaEstado());

                if(datos.getNumRadicaEstado()== null ||
                   datos.getNumRadicaEstado().equals(turno.getIdWorkflow())){
                    if (datos.getEstado().getIdEstado() != folio.getEstado()
                        .getIdEstado())
                    {
                        //creacion en tabla histórica de estados de folio:
                        EstadoHistoriaEnhanced hist = new EstadoHistoriaEnhanced();
                        hist.setIdEstadoHistoria(String.valueOf(folio.getLastIdHistorialEstados() +
                            1));
                        hist.setFolio(folio);
                        hist.setEstadoOrigen(folio.getEstado());
                        hist.setEstadoDestino(datos.getEstado());
                        hist.setComentario(datos.getComentarioCambioEstado());
                        hist.setComentarioAnterior(folio.getComentario());
                        hist.setUsuario(usuario);
                        hist.setFechaRegistro(new Date());
                        pm.makePersistent(hist);

                        folio.setLastIdHistorialEstados(folio.getLastIdHistorialEstados() +
                            1);
                        folio.setEstado(datos.getEstado());
                        folio.setComentario(datos.getComentarioCambioEstado());
                        datos.setEstado(null);
                        datos.setNumRadicaEstado(null);
                    }
                }
            }

            if (datos.getTipoPredio() != null) {
                set.add(datos.getNumRadicaTipoPredio());

                if(datos.getNumRadicaTipoPredio()== null ||
                   datos.getNumRadicaTipoPredio().equals(turno.getIdWorkflow()))
            }

```

```

        folio.setTipoPredio(datos.getTipoPredio());
        datos.setTipoPredio(null);
        datos.setNumRadicaTipoPredio(null);
    }
}

if (datos.getZonaRegistral() != null){
    set.add(datos.getNumRadicaTipoPredio());

    if(datos.getNumRadicaZonaRegistral()== null ||
       datos.getNumRadicaZonaRegistral().equals(turno.getIdWorkflow()))
    {
        this.updateZonaRegistralFolio(folio, datos.getZonaRegistral(), usuario, pm);
        datos.setZonaRegistral(null);
        datos.setNumRadicaZonaRegistral(null);
    }
}

```

**Por:**

```

/*
 * @author : Henry Gómez Rocha
 * @change : Comentareando en los siguientes métodos la asignación a null de los datos.
 * Caso Mantis : 0004503
 */

if (datos.getLindero() != null) {
    set.add(datos.getNumRadicaLindero());

    if(datos.getNumRadicaLindero()== null ||
       datos.getNumRadicaLindero().equals(turno.getIdWorkflow())){
        folio.setLindero(datos.getLindero());
        //datos.setLindero(null);
        //datos.setNumRadicaLindero(null);
    }
}

if (datos.getCodCatastral() != null) {
    set.add(datos.getNumRadicaCodCatastral());

    if(datos.getNumRadicaCodCatastral() == null ||
       datos.getNumRadicaCodCatastral().equals(turno.getIdWorkflow())){
        folio.setCodCatastral(datos.getCodCatastral());
        //datos.setCodCatastral(null);
        //datos.setNumRadicaCodCatastral(null);
    }
}

if (datos.getCodCatastralAnterior() != null) {
    set.add(datos.getNumRadicaCodCatastralAnterior());

    if(datos.getNumRadicaCodCatastralAnterior() == null ||
       datos.getNumRadicaCodCatastralAnterior().equals(turno.getIdWorkflow())){
        folio.setCodCatastralAnterior(datos.getCodCatastralAnterior());
        //datos.setCodCatastralAnterior(null);
        //datos.setCodCatastralAnterior(null);
    }
}

// bug: 3580
if (datos.getRadicacion() != null) {
    set.add(datos.getNumRadicaRadicacion());

    if(datos.getNumRadicaRadicacion()== null ||
       datos.getNumRadicaRadicacion().equals(turno.getIdWorkflow())){
        folio.setRadicacion( datos.getRadicacion() );
        //datos.setRadicacion(null);
        //datos.setNumRadicaRadicacion(null);
    }
}

if (datos.getFechaApertura() != null) {
    set.add(datos.getNumRadicaFechaApertura());

    if(datos.getNumRadicaFechaApertura()== null ||

```

```

        datos.getNumRadicaFechaApertura().equals(turno.getIdWorkflow())){
        folio.setFechaApertura( datos.getFechaApertura() );
        //datos.setFechaApertura(null);
        //datos.setNumRadicaFechaApertura(null);
    }

    if (datos.getEstado() != null) {
        set.add(datos.getNumRadicaEstado());
    }

    if(datos.getNumRadicaEstado()== null ||
       datos.getNumRadicaEstado().equals(turno.getIdWorkflow())){
        if (datos.getEstado().getIdEstado() != folio.getEstado(
            .getIdEstado()))
        {
            //creacion en tabla histórica de estados de folio:
            EstadoHistoriaEnhanced hist = new EstadoHistoriaEnhanced();
            hist.setIdEstadoHistoria(String.valueOf(folio.getLastIdHistorialEstados() +
                1));
            hist.setFolio(folio);
            hist.setEstadoOrigen(folio.getEstado());
            hist.setEstadoDestino(datos.getEstado());
            hist.setComentario(datos.getComentarioCambioEstado());
            hist.setComentarioAnterior(folio.getComentario());
            hist.setUsuario(usuario);
            hist.setFechaRegistro(new Date());
            pm.makePersistent(hist);

            folio.setLastIdHistorialEstados(folio.getLastIdHistorialEstados() +
                1);
            folio.setEstado(datos.getEstado());
            folio.setComentario(datos.getComentarioCambioEstado());
            //datos.setEstado(null);
            //datos.setNumRadicaEstado(null);
        }
    }

    if (datos.getTipoPredio() != null) {
        set.add(datos.getNumRadicaTipoPredio());

        if(datos.getNumRadicaTipoPredio()== null ||
           datos.getNumRadicaTipoPredio().equals(turno.getIdWorkflow()))
        {
            folio.setTipoPredio(datos.getTipoPredio());
            //datos.setTipoPredio(null);
            //datos.setNumRadicaTipoPredio(null);
        }
    }

    if (datos.getZonaRegistral()!=null){
        set.add(datos.getNumRadicaTipoPredio());

        if(datos.getNumRadicaZonaRegistral()== null ||
           datos.getNumRadicaZonaRegistral().equals(turno.getIdWorkflow()))
        {
            this.updateZonaRegistralFolio(folio, datos.getZonaRegistral(), usuario, pm);
            //datos.setZonaRegistral(null);
            //datos.setNumRadicaZonaRegistral(null);
        }
    }
}

```

#### CLASE o ARCHIVO JSP:

C:\WORK\SIR-  
APP\src\gov\sir\core\negocio\acciones\correccion\AnEdicionCiudadanoSobreTodosFolios.java

#### Método:

public EvnRespMsgEdicionCiudadanoSobreTodosFolios  
editCiudadanoOnApply( EvnMsgEdicionCiudadanoSobreTodosFolios evento )

#### Código Línea 640:

```
// actualizar el folio
forseti.updateFolio( representation2_Folio_List_Item, param_CiudadanoUpdatedBy, null, false );
```

**Por:**

```
/*
 * @author : Henry Gómez Rocha
 * @change : Seteando turno para enviarlo en lugar de enviar null.
 * Caso Mantis : 0004503
 */

TurnoPk tid = new TurnoPk();
tid.anio = evento.getTurno().getAnio();
tid.idCirculo = evento.getTurno().getIdCirculo();
tid.idProceso = evento.getTurno().getIdProceso();
tid.idTurno = evento.getTurno().getIdTurno();

// actualizar el folio
forseti.updateFolio(representation2_Folio_List_Item, param_CiudadanoUpdatedBy, tid, false );
```

**CLASE o ARCHIVO JSP:**

**C:\WORK\SIR-APP\src\gov\sir\core\negocio\acciones\registro\ANCalificacion.java**

**Método:**

**private EventoRespuesta eliminarAnotacionTemporal(EvnCalificacion evento)**

**Código Línea 3564:**

```
if(anotacion.getIdNaturalezaJuridica().equals(CNaturalezaJuridica.ACTUALIZACION_DE_LINDEROS)||

anotacion.getIdNaturalezaJuridica().equals(CNaturalezaJuridica.ACTUALIZACION_AREA)||

anotacion.getIdNaturalezaJuridica().equals(CNaturalezaJuridica.DECLARACION_DE_CONSTRUCCION_CON_SUBSIDIO_FAMILIAR_DE_VIVIENDA)||

anotacion.getIdNaturalezaJuridica().equals(CNaturalezaJuridica.DECLARACION_DE_CONSTRUCCION_EN_SUELO_PROPPIO) ||

anotacion.getIdNaturalezaJuridica().equals(CNaturalezaJuridica.DECLARACION_PARTE_RESTANTE)) {
    /**LINDERO*/
    if(numeroNatLinderos==0 &&
numeroNatLinderos2==0 && numeroNatLinderos3==0 && numeroNatLinderos4==0){
        if(numeroNatDireccion == 0){
            if(numeroArea == 0){

if(numeroAclaracion == 0 && eliminarAnotaciones){
folioDatosAux.setLindero(null);
forseti.actualizarFolioDatosTMP(folioPk.idMatricula,folioDatosAux,false);
}
}
}
}

if(anotacion.getIdNaturalezaJuridica().equals(CNaturalezaJuridica.ACTUALIZACION_DE_NOMENCLATURA)){
    /**LINDEROS*/
    if(numeroNatLinderos==0 &&
numeroNatLinderos2==0 && numeroNatLinderos3==0 && numeroNatLinderos4==0){
        if(numeroNatDireccion == 0){
            if(numeroArea == 0){

if(numeroAclaracion == 0 && eliminarAnotaciones){
folioDatosAux.setLindero(null);
forseti.actualizarFolioDatosTMP(folioPk.idMatricula,folioDatosAux,false);
}
}
}
}
```

Por:

```
/*
 * @author    : Henry Gómez Rocha
 * @change : Comentareando en los siguientes métodos la asignación a null de los datos.
 * Caso Mantis : 0004503
 */
```

if(anotacion.getNaturalezaJuridica().getIdNaturalezaJuridica().equals(CNaturalezaJuridica.ACTUALIZACION\_DE\_LINDEROS))

```

        anotacion.getIdNaturalezaJuridica().equals(CNaturalezaJuridica.ACTUALIZACION_AREA)||

        anotacion.getIdNaturalezaJuridica().equals(CNaturalezaJuridica.DECLARACION_DE_CONSTRUCCION_CON_SUBSIDIO_FAMILIAR_DE_VIVIENDA)||

        anotacion.getIdNaturalezaJuridica().equals(CNaturalezaJuridica.DECLARACION_DE_CONSTRUCCION_EN_SUELO_PROPPIO)||

        anotacion.getIdNaturalezaJuridica().equals(CNaturalezaJuridica.DECLARACION_PARTE_RESTANTE)) {
            /**LINDEROS*/
            if(numeroNatLinderos==0 &&
numeroNatLinderos2==0 && numeroNatLinderos3==0 && numeroNatLinderos4==0){
                if(numeroNatDireccion == 0){
                    if(numeroArea == 0){

if(numeroAclaracion == 0 && eliminarAnotaciones){
//folioDatosAux.setLadero(null);

forseti.actualizarFolioDatosTMP(folioPk.idMatricula,folioDatosAux,false);
}
}
}
}

if(anotacion.getIdNaturalezaJuridica().equals(CNaturalezaJuridica.ACTUALIZACION_DE_NOMENCLATURA)){
    /**LINDEROS*/
    if(numeroNatLinderos==0 &&
numeroNatLinderos2==0 && numeroNatLinderos3==0 && numeroNatLinderos4==0){
        if(numeroNatDireccion == 0){
            if(numeroArea == 0){

if(numeroAclaracion == 0 && eliminarAnotaciones){
//folioDatosAux.setLadero(null);

forseti.actualizarFolioDatosTMP(folioPk.idMatricula,folioDatosAux,false);
}
}
}
}

if(eliminarAnotacionesDir){
eliminadasDireccionesTMP = true;

forseti.actualizarFolioDatosTMP(folioPk.idMatricula,folioDatosAux,true);
}
}
}

if(anotacion.getIdNaturalezaJuridica().equals(CNaturalezaJuridica.ACCLARACION)){
    /**LINDEROS*/
    if(numeroNatLinderos==0 &&
numeroNatLinderos2==0 && numeroNatLinderos3==0 && numeroNatLinderos4==0){
        if(numeroNatDireccion == 0){
            if(numeroArea == 0 &&
eliminarAnotaciones){

//folioDatosAux.setLadero(null);
}
}
}
}

```

## **CLASE o ARCHIVO JSP:**

C:\WORK\SIR-APP\src\gov\sir\core\web\acciones\correccion\ AWModificarFolio.java

## Método:

```
protected EvnCorreccion  
doFolioEdit_Anotacion_SaveChanges( HttpServletRequest request, boolean enableCancelaciones )
```

## Código Línea 7242:

```
t2_folio.setZonaRegistral( t0_folio.getZonaRegistral() );
```

Por:

```
/*
 * @author   : Henry Gómez Rocha
 * @change   : Se comentarea la siguiente línea para evitar que se guarde información
 *              en la tabla SIR_NE_FOLIO_DATOS_TMP ya que los datos de la zona registral
 *              son diferentes de null y por tal motivo el sistema lo interpreta como
 *              si siempre se estuviesen haciendo modificaciones al folio.
 * Caso Mantis : 0004503
 */
```

```
//t2_folio.setZonaRegistral( t0_folio.getZonaRegistral() );
```

## **Método:**

**private EvnCorrecció**

### **doFolioEdit\_InfoFo**

if( false )

```
/*
 * @author : Henry Gómez Rocha
 * @change : Con el propósito de optimizar el rendimiento del aplicativo se
 *            comentara el siguiente if debido a que nunca ingresa al cuerpo
 *            del mismo.
 * Caso Mantis : 0004503
```

```

    */
    // if( false )
    // return null;

```

**Código Línea 9009:**

```

if(t2_folio.getDocumento() != null && t2_folio.getDocumento().getIdDocumento() == null){
    t2_folio.setDocumento(t0_folio.getDocumento());
}

```

**Por:**

```

if(t2_folio.getDocumento() != null && t2_folio.getDocumento().getIdDocumento() == null){
    /*
     * @author : Henry Gómez Rocha
     * @change : Se envía t1_folio.getDocumento() en lugar de
     *           t0_folio.getDocumento() ya
     *           que este último no contiene los datos de la sesión.
     * Caso Mantis : 0004503
    */
    t2_folio.setDocumento(t1_folio.getDocumento());
}

```

**Método:**

```
private void preservarInfoBasicaFolio(HttpServletRequest request)
```

**Código Línea 11875:**

```

if (request.getParameter(AWModificarFolio.FOLIO_TIPO_PREDIO) != null) {
    session.setAttribute(AWModificarFolio.FOLIO_TIPO_PREDIO,
        request.getParameter(AWModificarFolio.FOLIO_TIPO_PREDIO));
}

```

**Por:**

```

/*
 * @author : Henry Gómez Rocha
 * @change : Con el propósito de optimizar el rendimiento del aplicativo se
 *           comentarea el siguiente if debido a que ya se encuentra en la línea
 *           11856.
 * Caso Mantis : 0004503
*/
// if (request.getParameter(AWModificarFolio.FOLIO_TIPO_PREDIO) != null) {
//     session.setAttribute(AWModificarFolio.FOLIO_TIPO_PREDIO,
//         request.getParameter(AWModificarFolio.FOLIO_TIPO_PREDIO));
//}

```

## 103 Reporte 57 - Despliegue Numero Turno

### **REQUERIMIENTO**

Reporte 57 – Diario Radicador de Matriculas, en la columna turno el reporte no esta reflejando el numero del turno con el cual se dio apertura a la matricula inmobiliaria, en esta columna el reporte debe mostrar el numero del turno con el cual se dio apertura al folio de matricula. Se debe ajustar a un solo espacio las líneas.

### **DESARROLLO**

Reporte 57: Diario radicador de matriculas

- Script Bases de Datos:  
SP\_REPORTE\_57.sql

```

*****
/** AHERRENO 25/02/2010 */
/** Se agrega campo t_fol(fol).flio_radicacion ya que se estaba enviando NULL
*****
```

```

VALUES      (t_fol(fol).id_matricula,          t_fol(fol).flio_radicacion,          t_fol(fol).id_usuario,          v_municipio,
TO_CHAR(t_fol(fol).flio_fecha_apertura,'HH24:MI:SS'), TO_CHAR(t_fol(fol).flio_fecha_apertura,'DD-MM-YYYY HH24:MI:SS'),
t_fol(fol).id_zona_registral, v_ciud, v_direccion, p_sec_proceso);

```

SP\_REPORTE\_57\_Marcha\_Atras.sql

➤ Creación De Clases:  
NINGUNO

➤ Modificación de archivos:

**Reporte 57 – Diario Radicador de Matriculas**

Tamaño – Campos: \$F{MATRICULA}, \$F{TURNO}, \$F{DIRECCION},  
\$F{USUARIO}, \$F{MUNICIPIO}, \$F{ANOTACION\_PROPIETARIO}, \$F{HORA}

Antes	18
Ahora	13

**115 Impresión recibos  
(Informativo)**

**REQUERIMIENTO**

Impresión recibos. Se comunica el funcionario Carlos Cortes e informa que no salieron los recibos de los turnos 2009-242-1-1462 y 2009-242-1-1463. Se indica al funcionario Carlos Cortes verificar si la impresora local está conectada y encendida, nos confirma que está correctamente, pero se recibe una llamada de Bucaramanga y nos informa el funcionario José García que salieron dos recibos reimpresos de la oficina de Barbacoas en Bucaramanga, se indicó al funcionario José García de Bucaramanga enviar el recibo que les salió de la oficina de barbacoas para verificar información y al recibirlo se confirma que el recibo es de la oficina de Barbacoas.

Por favor verificar por qué están saliendo los recibos de Barbacoas en Bucaramanga si estos se configuran en las impresoras locales de cada cajero.

Se anexa recibo. Barbacoas. Id Glpi 12187.

**DESARROLLO**

Según mesa de ayuda este error se produjo solo en este caso, no se ha vuelto a replicar la incidencia, sin embargo se analiza el problema para averiguar su causa.

Después de cierto tiempo este error no fue posible de replicar, se hará seguimiento desde mesa de ayuda.

**121 Turnos de Antiguo Sistema**

**REQUERIMIENTO**

Turnos de Antiguo Sistema. Los turnos de registro y correcciones que son enviados a antiguo sistema desde las fases calificación para los turnos de registro y revisión y análisis para los turnos de correcciones, estos turnos al terminar el flujo de antiguo sistema y volver a las fases de las cuales fueron enviadas a este proceso, quedan con la estación del usuario que realiza el proceso de antiguo sistema y no en la estación del usuario que lo envió a este proceso.

Se requiere escalar este caso a desarrollo ya que el sistema asigna el turno al usuario que realiza el proceso de antiguo sistema, este usuario por lo general no tiene asignado el rol de calificador y/o responsable de correcciones. El turno debe siempre volver al usuario que envío el turno inicialmente a antiguo sistema.

**DESARROLLO**

Luego de replicar el caso, se encontró que al momento de hacer la revisión final de un turno desde el rol Encargado Antiguo Sistema, dicho turno no retornaba al usuario que lo envió a Solicitar información de folios a otras dependencias (ANTIGUO SISTEMA), dado que no se verificaba que después de esta fase el sistema debía retornar al usuario que lo había enviado a Antiguo Sistema; así mismo, al momento de corregir este inconveniente también se hizo necesario asignar nuevamente el bloqueo del turno al usuario que envió dicho turno a Solicitar información de folios

a otras dependencias.

La prueba de este caso se hizo con la colaboración de Lourdes Brito quien hace parte del personal de Mesa de Ayuda.

- Script Bases de Datos:
  - NINGUNA

- Creación De Clases:
  - NINGUNA

CLASE o ARCHIVO JSP:
C:\WORK\SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDOTurnosNuevosDAO.java
Método:
<b>Se agrega el siguiente Código en la Línea 1157:</b> <pre>/*  * @author : Henry Gómez Rocha  * @change : Se verifica que al momento de hacer la revisión final de un turno desde el rol Encargado Antiguo Sistema, dicho turno retorne al usuario que lo envió a Solicitar información de folios a otras dependencias (ANTIGUO SISTEMA). *Caso Mantis : 0002484 */ if(CFase.REG_REPARTO.equals(turno.getIdFaseRelacionActual()) &amp;&amp; CFase.CAL_CALIFICACION.equals(fase)){ TurnoHistoria itrTurno = null;  List turnoHistorial = turno.getHistorials(); Iterator itr = turnoHistorial.iterator(); while(itr.hasNext()){ itrTurno = (TurnoHistoria) itr.next(); if(CFase.CAL_CALIFICACION.equals(itrTurno.getFase()) &amp;&amp; CFase.REG_REPARTO.equals(itrTurno.getFaseAnterior())){ turnoEjecucionEnh.setEstacion(itrTurno.getIdAdministradorSAS()); estacionAsignada=itrTurno.getIdAdministradorSAS(); } } }</pre>

#### Se agrega el siguiente Código en la Línea 1833:

```
/*
 * @author : Henry Gómez Rocha
 * @change : Se verifica que al momento de hacer la revisión final de un
turno desde el rol Encargado Antiguo Sistema, dicho turno
retorne el bloqueo al usuario que lo envió a Solicitar información de
folios a otras dependencias (ANTIGUO SISTEMA).
*Caso Mantis : 0002484
*/
if(CFase.REG_REPARTO.equals(turno.getIdFaseRelacionActual()) && CFase.CAL_CALIFICACION.equals(faseDestino)){
turnoEnh.setModoBloqueo(CModoBloqueo.DELEGAR_CUALQUIERA);
}
```

### 122 Nota devolutiva no salio impresa (Informativo)

#### REQUERIMIENTO

Nota Devolutiva no salio impresa. Turno 2009-300-6-39097. Se comunica el funcionario Jose Luis Garcia e informa que la nota devolutiva del turno 2009-300-6-39097 no salió impresa. Se indicó al funcionario Jose Luis Garcia redireccionar las impresiones de notas devolutivas a otra impresora, enviar una impresión de prueba y verificar si la impresora está funcionando correctamente y si el administrador de impresión está configurado correctamente para las notas devolutivas a lo que nos informa que ya redireccionó las notas devolutivas a otra impresora y esta no sale, que la impresora esta funcionando bien ya que el resto de las notas devolutivas si salieron y que también han salido correctamente los formularios de calificación.

Por favor indicar ID de la nota devolutiva para que el funcionario pueda imprimirla, el turno aparece como devuelto. Bucaramanga. Id Ref 12493.

Este caso debe ser escalado a desarrollo para dar solución a este inconveniente presentado con las notas devolutivas de este turno. Se requiere crear acta de requerimiento a desarrollo para la solución de este caso.

Se reproduce este error el 26 de Octubre en la Orips de Armenia, se anexa a esta acta el caso con Id Glpi No 13222. Turno 2009-280-6-18797, Se comunica la funcionaria Gloria Buitrago de la oficina e informa que la nota devolutiva del turno 2009-280-6-18797 no salió impresa. Se verifica el turno 2009-280-6-18797 y se encuentra en mesa de control con resolución devuelto al momento en que en calificación madaron a imprimir la nota devolutiva esta no salió impresa, por lo tanto están tratando de reimprimirla y tampoco sale. Por favor enviar el PDF de la nota devolutiva para enviarsela a la oficina y puedan seguir con el trámite del turno.

### **DESARROLLO**

Este error no fue posible de replicar, se hará seguimiento desde mesa de ayuda, ya que tampoco se ha vuelto a presentar el error para su análisis.

## **124 Impresión certificados (Informativo)**

### **REQUERIMIENTO**

Se comunica el funcionario Miguel Galvis e informa que algunos certificados están saliendo impresos con los datos incompletos. Se verifican los certificados que envía el funcionario y los datos que salen incompletos son los que aparecen en la parte superior donde dice: Certificado de tradición y donde se especifica la matrícula.

Nos informa el funcionario que los certificados se están imprimiendo por las siguientes impresoras:

LP1 con IP 10.10.19.6

LP3 con IP 10.10.19.5

LP5 con IP 10.10.19.4.

Se adjuntan los certificados para verificar el caso. Pasto. Glpi Ref 12546.

### **DESARROLLO**

Este error no fue posible de replicar, se hará seguimiento desde mesa de ayuda.

## **125 Turnos de mayores valores cancelados por vencimiento1**

### **REQUERIMIENTO**

REGLAS MODULO DEL MAYOR VALOR- (MAYORES VALORES VENCIDOS).

A solicitud de Mesa de ayuda se definen las reglas para el vencimiento de los turnos que tienen cargado un mayor valor.

.-Según el art. 7 De la Resolución 4907 de 2009 de la Superintendencia de Notariado y Registro, el término para cancelar un mayor valor es de dos meses (calendario) contados a partir de la fecha de radicación del documento (ejemplo: 01 de abril de 2010 se radico el documento vence el mayor valor el 01 de Junio de 2010).

.-Transcurridos los dos meses sin cancelar el mayor valor el aplicativo enviará el turno de registro o de corrección, con mayor valor vencido a la pantalla administrativa de “Pagos de mayor valor cancelados por Vencimiento”, para que desde allí el usuario genere las notas devolutivas con la correspondiente causal de devolución, para todos los turnos que ese encuentren en esa pantalla. Como consecuencia de lo anterior y en acción simultánea

el aplicativo eliminará el ítem de liquidación de mayor valor .

.-Los turnos de registro con nota devolutiva de vencimiento de mayor valor, deben avanzar a la fase de Firma de Registrador y los turnos de corrección con nota devolutiva de vencimiento de mayor valor, deben enviarse a la fase de revisión y análisis, para negar la corrección solicitada.

.-Transcurridos los dos meses sin cancelar el mayor valor, el aplicativo SIR no debe permitir el pago del mayor valor, independientemente de la fase donde se encuentre. (aplica para todo tipo de documento radicado- registro-correcciones)

### **DESARROLLO**

Se modificaron los siguientes archivos:

CLASE:
<b>gov.sir.core.web.acciones.pagomayorvalor.AWPagoMayorValor.java</b>
<b>Código Insertado Línea 247:</b> <pre>/* JAICAZAR caso Mantis 2548 Acta - Requerimiento No 125 - Turnos de mayores valores cancelados por vencimiento  * Se define la toma de la fase se Solicitud if(th.getFase().equals(WebKeys.SOLICITUD))  */ TurnoHistoria th = null; List historial_turno = turno.getHistorials(); Iterator hist = historial_turno.iterator(); while(hist.hasNext()){     th = (TurnoHistoria) hist.next();     if(th.getFase().equals(WebKeys.SOLICITUD)){         break;     } }</pre>

### 132 Impresion en la fase Mesa Control Correcciones

#### **REQUERIMIENTO**

Cuando se avanza un turno de correcciones de la fase revisar y aprobar, el sistema avanza el turno a la fase mesa control correcciones, donde el sistema permite imprimir certificado de las matrículas que fueron corregidas o modificadas con el turno que se está tramitando siempre y cuando la aprobación de la corrección sea en la misma fecha en la cual se va a imprimir el certificado. El sistema no está imprimiendo los certificados asociados porque la fecha de aprobación e impresión del certificado no son la misma y está tomando la fecha de aprobación con la fecha en que se ejecutó la corrección.

Por favor escalar este caso a desarrollo para su solución general ya que el sistema debe validar la fecha en que se ejecutó la corrección solamente para poder expedir el certificado. Por favor crear acta de requerimiento para desarrollo.

### **DESARROLLO**

- Script Bases de Datos:
  - NINGUNA
- Creacion De Clases:
  - NINGUNA
- Se modificaron los siguientes archivos:

CLASE:
<b>gov.sir.core.negocio.acciones.correcion.AnCorr_MesaControl.java</b>
Método:

	<pre><code>private EvnRespMsgMesaControl imprimirCertificadosMesa(EvnCorreccion evento) throws EventoException</code></pre>	
	<p><b>Código Linea 1130:</b></p> <p>local_TurnoSearchContext.setValue("\$idFase", "COR_REVISAR_APROBAR");</p> <p><b>Por:</b></p> <pre><code>/*JAlcazar 04/05/2010 caso mantis 2754 Requerimiento 132 - Impresion en la fase Mesa Control Correcciones  * La fecha de aprobación corresponde a la fecha en que el turno pasa a Mesa de Control  CFase.COR_MESA_CONTROL  */ local_TurnoSearchContext.setValue("\$idFase", CFase.COR_MESA_CONTROL);</code></pre> <p><b>Código Eliminado Linea 326:</b></p> <pre><code>if( !founded_Date ) {      // caso2: actuacion administrativa; buscar idFaseAprobacion      Turno local_Turno;      // buscar turno en la db?     local_Turno = turno;      // por el momento se observa el registro historico que tiene el turno     // List&lt; TurnoHistoria &gt;     List local_TurnoHistoria;      local_TurnoHistoria = local_Turno.getHistorials();      JXPathContext local_TurnoSearchContext = jx_GetContext( local_TurnoHistoria );     local_TurnoSearchContext.getVariables().declareVariable("idFase", "");     local_TurnoSearchContext.setValue("\$idFase", "COR_ACT_EJECUTAR");      // search matches     // apply first      Iterator local_TurnoSearchContextIterator = local_TurnoSearchContext.iterate(         ".[@fase = \$idFase]");      for ( ; local_TurnoSearchContextIterator.hasNext(); ) {         TurnoHistoria element = (TurnoHistoria) local_TurnoSearchContextIterator.             next();         local_TurnoHistoriaDate = element.getFecha();         founded_Date = true;     } // for  } // if</code></pre>	
	<h3>136 Anotaciones Temporales Folios (Informativo)</h3> <p><b>REQUERIMIENTO</b></p> <p>Anotaciones Temporales. Al realizar correcciones sobre ciudadanos en las anotaciones de los folios, los turnos de correcciones avanzan correctamente e imprimen correctamente la información modificada, pero al ingresar otro turno ya sea de registro o correcciones el sistema muestra temporal la anotación o anotaciones donde se hizo edición o corrección de ciudadano y es necesario que por medio de la mesa de ayuda se eliminen estos datos temporales que quedan en los folios.</p> <p>Por favor escalar a desarrollo para corregir este tema relacionado con los datos temporales en las anotaciones ya que estos datos deben quedar definitivos en su totalidad con el turno de correcciones cuando realiza su flujo normal y no se debe mostrar ningún temporal en</p>	

los folios.

Por favor crear acta de requerimiento a desarrollo para su solución general. Acta de referencia No 078 (Despliegue 2.2.0).

### **DESARROLLO**

Dado que este caso es en últimas un sub-caso del caso Mantis No 0004503 y cuya Acta de Requerimiento es la No. 94 (Despliegue Actual) y lleva por nombre "Proceso Calificación - Envió turno a proceso de correcciones" y teniendo en cuenta que la solución del mismo da solución al presente caso, se decide cerrarlo.

## **137 Consulta matriculas y anotaciones (Informativo)**

### **REQUERIMIENTO**

Realizar la consulta de las matriculas y anotaciones en las que un ciudadano ha sido, o es propietario, con base en el listado de 5212 nombres y cedulas suministradas por la Fiscalía General de la Nación.

Se debe hacer búsqueda en el índice de propietarios, por todos los círculos, a partir de la cedula de ciudadanía, nombre y primer apellido, nombre y segundo apellido o sus combinaciones dependiendo de los datos suministrados.

Se debe hacer verificación de la mayor cantidad de información origen posible, para evitar el reporte errados de propiedades.

### **DESARROLLO**

El costo medio de cada consulta por cedula de ciudadanía con la siguiente instrucción SQL es de dos minutos.

```
select /*+ ordered index(sir_ne_anotacion_ciudadano idx_ntcd_cddn)
index(sir_ne_anotacion pk_sir_ne_anotacion)
index(sir_ne_anotacion_ciudadano pk_sir_ne_anotacion_ciudadano) */ LC.id_ciudadano,
LC.cddn_tipo_documento, LC.cddn_documento, LC.cddn_nombre,
LC.cddn_apellido1, LC.cddn_apellido2, D.drcc_especificacion, 'COLOMBIA' pais,
DP.dprt_nombre, M.mncp_nombre, V.vrda_nombre, F.id_matricula,
to_char(F.FLIO_FECHA_APERTURA, 'yyyy-mm-dd hh24:mi') flio_fecha_apertura,
a.id_anotacion, a.NTCN_ESPECIFICACION, decode(ac.ntcd_propietario, 1, 'SI', 'NO')
propietario, AC.NTCD_ROL_PERSONA as rol from (select /*+ index( sir_ne_ciudadano
idx_numero_tipo)*/
C.cddn_apellido1, C.cddn_apellido2, C.cddn_documento, c.cddn_fecha_creacion,
C.cddn_nombre, c.cddn_solicitante, c.cddn_telefono, C.cddn_tipo_documento,
C.id_circulo, C.id_ciudadano, C.id_nciudadano
from sir_ne_ciudadano C where C.cddn_documento = '123' and C.cddn_tipo_documento
= 'CC') LC, sir_ne_anotacion_ciudadano AC, sir_ne_anotacion A, sir_ne_folio F,
sir_ne_zona_registral ZR, sir_ne_vereda V, sir_ne_municipio M, sir_ne_departamento DP,
sir_ne_direccion D where AC.id_ciudadano = LC.id_ciudadano and A.id_anotacion =
AC.id_anotacion
and A.id_matricula = AC.id_matricula and f.id_matricula = a.id_matricula and f.id_circulo
!= '999' and zr.id_zona_registral = f.id_zona_registral and v.id_vereda = zr.id_vereda
and v.id_departamento = zr.id_departamento and v.id_municipio = zr.id_municipio and
m.id_municipio = v.id_municipio and m.id_departamento = v.id_departamento and
```

```

dp.id_departamento = m.id_departamento
and f.id_matricula = d.id_matricula and To_Number( d.id_direccion ) = (select max(
To_Number( ld.id_direccion )) from sir_ne_direccion ld where ld.id_matricula =
f.id_matricula) order by f.id_matricula, a.id_anotacion;

```

Para la búsqueda por nombre se usa la siguiente instrucción SQL con un costo medio de 1.7 minutos por consulta.

```

select /*+ ordered index(sir_ne_anotacion_ciudadano idx_ntcd_cddn)
index(sir_ne_anotacion pk_sir_ne_anotacion) */ LC.id_ciudadano,
LC.cddn_tipo_documento, LC.cddn_documento, LC.cddn_nombre, LC.cddn_apellido1,
LC.cddn_apellido2, D.drcc_especificacion, 'COLOMBIA' pais, DP.dprt_nombre,
M.mncp_nombre, V.vrda_nombre, F.id_matricula,
to_char(F.FLIO_FECHA_APERTURA, 'yyyy-mm-dd hh24:mi') flio_fecha_apertura,
a.id_anotacion, a.NTCN_ESPECIFICACION, decode(ac.ntcd_propietario, 1, 'SI', 'NO')
propietario, AC.NTCD_ROL_PERSONA as rol from (select /*+ index_combine(
sir_ne_ciudadano idx_cddn_ape1 idx_cddn_ape2 idx_cddn_nombre)*/ C.cddn_apellido1,
C.cddn_apellido2, C.cddn_documento, c.cddn_fecha_creacion, C.cddn_nombre,
c.cddn_solicitante, c.cddn_telefono, C.cddn_tipo_documento, C.id_circulo,
C.id_ciudadano, C.id_nciudadano
from sir_ne_ciudadano c where c.cddn_apellido1 = 'JORGE LUIS' and c.cddn_apellido2
= 'MADRID' and c.cddn_nombre = 'LOPEZ') LC, sir_ne_anotacion_ciudadano AC,
sir_ne_anotacion A, sir_ne_folio F, sir_ne_zona_registral ZR, sir_ne_vereda V,
sir_ne_municipio M, sir_ne_departamento DP, sir_ne_direccion D where
AC.id_ciudadano = LC.id_ciudadano and A.id_anotacion = AC.id_anotacion
and A.id_matricula = AC.id_matricula and f.id_matricula = a.id_matricula and f.id_circulo
!= '999' and zr.id_zona_registral = f.id_zona_registral and v.id_vereda = zr.id_vereda and
v.id_departamento = zr.id_departamento and v.id_municipio = zr.id_municipio and
m.id_municipio = v.id_municipio and m.id_departamento = v.id_departamento
and dp.id_departamento = m.id_departamento and f.id_matricula = d.id_matricula and
To_Number( d.id_direccion ) = (select max( To_Number( ld.id_direccion)) from
sir_ne_direccion ld where ld.id_matricula = f.id_matricula ) order by f.id_matricula,
a.id_anotacion;

```

Se debe construir un paquete PL/SQL que almacene el resultado de la consulta en una tabla temporal del esquema adminsir, para su posterior exportación a medio magnético y envió a SNR

Tiempo estimado de desarrollo, ejecución, verificación y generación de archivo final:  
100 horas

La respuesta al Oficio 2009EE22852 de octubre de 2009, que generó la consulta, fue radicada con el oficio 2010ER748 de enero 7 de 2010; y radicado en PWC con el radicado 51434 de Enero 8 de 2010.

#### 141 Pantallas Administrativas ver detalles de turno (Informativo)

##### **REQUERIMIENTO**

Cuando un turno de documento ha tenido más de una entrada el sistema debe encadenar los turnos de manera tal que cuando se consulte cualquiera de ellos se observe el listado de los turnos anteriores o posteriores al consultado, tanto de registro como de devoluciones

dinero.

Por ejemplo el turno 2008-350-6-21205 al consultarla el sistema no muestra los turnos de nuevas entradas 2009-350-6-6982 y 2009-350-6-10146 para el mismo documento, únicamente muestra el turno de devolución de dinero con el cual se encuentra asociado 2009-350-4-147 todos están relacionados para el folio de matrícula 350-185795.

Al consultar el turno 2009-350-6-6982 muestra el turno siguiente y anterior y no refleja el turno de devoluciones de dinero que está vinculado al turno 2008-350-6-21205.

Al consultar el turno 2009-350-6-10146 muestra los dos turnos anteriores 6982, 21205 y no refleja el turno de devolución asociado.

### ***DESARROLLO***

#### **Análisis de Requerimiento:**

Se realizaron pruebas en los ambientes de desarrollo y producción para corroborar el inconveniente presentado en la Orip de Ibagué.

Se evidenció que este problema no se está presentando, al realizar las pruebas en la pantalla administrativa-> ver detalles de turno se observó la información correspondiente a los turnos relacionados tanto los turnos anteriores y siguientes al turno trabajado. Se decide en conjunto con el personal de mesa de ayuda cerrar el caso.

### **142 Radicación nuevas entradas validaciones con turnos devoluciones de dinero (Informativo)**

#### **REQUERIMIENTO**

Cuando un turno de documento ha tenido más de una entrada, el sistema debe verificar y encadenar los turnos asociados a éste, de tal manera que cuando se ingrese nuevamente el turno para registro o certificado, debe VALIDAR si sobre alguno de los turnos asociados se ha vinculado un turno de devolución de dinero en trámite o finalizado y no debe permitir radicar la nueva entrada del turno de registro, salvo si el turno de devolución tiene el estado de resolución NEGADA.

### ***DESARROLLO***

#### **Análisis de Requerimiento:**

Luego de crear los turnos 2010-999-6-156 y 2010-999-4-23 en el ambiente de desarrollo y los turnos 2010-999-6-319 y 2010-999-4-51 en el ambiente de producción no se encontró el problema del caso actual. Se hizo la respectiva búsqueda de los casos resueltos que se encuentran relacionados con el presente caso y se encontró que la solución dada al caso mantis 4275 y cuya Acta de Requerimiento es la No 195 (Despliegue 2.6.1) y lleva por nombre "La nueva entrada no asocia el certificado" resolvió este problema.

Las pruebas en SIR fueron realizadas con la colaboración de Javier Rocha y Sergio Gonzales quienes hacen parte del personal de Mesa de Ayuda.

### **144 Notas Informativas (Informativo)**

#### **REQUERIMIENTO**

Se realizan pruebas en el aplicativo para revisar las notas informativas creadas en los

diferentes procesos del sistema para realizar la impresión de las mismas. Al tratar de realizar la impresión de estas notas informativas el sistema no las imprime y el administrador de impresión está bien configurado. Por favor crear acta de requerimiento a desarrollo para revisar la impresión de las notas informativas del sistema ya que no se están imprimiendo cuando se realiza la creación de las notas desde las diferentes fases del sistema, este caso debe estar relacionado al acta 087 con el cual fueron revisadas las impresiones de las notas informativas del rol responsable de correcciones. Id Glpi 13699.

### **DESARROLLO**

Se hicieron las pruebas respectivas en el ambiente de Desarrollo y Producción junto con Javier Rocha y Sergio González (Personal de Mesa de Ayuda) y no se encontró el problema manifestado en este caso. Como constancia se dejan los turnos creados en su orden en el ambiente de Desarrollo y Producción 2010-999-3-117 y 2010-999-6-306.

### **148 Digitador masivo – Direcciones - Cabida y Linderos (Informativo)**

#### **REQUERIMIENTO**

Se efectuaron pruebas con el turno de documento 2009-368-6-2519 al cual en el proceso de calificación se segregaron las matrículas inmobiliaria 368-48225, 368-48226, 368- 48227, luego se envió el turno a Digitación Masiva en esta fase se Agregó una dirección para las tres matrículas , después se eliminó esta dirección en la matrícula 368-48225, el sistema muestra error que el usuario no es dueño del bloqueo y aparentemente se visualiza que es eliminada la dirección, sin embargo al retomar nuevamente el folio se observa la dirección eliminada, así mismo cuando se trata de reformas de Reglamentos de propiedad horizontal y se ingresa en el campo de linderos el coeficiente actual o se actualiza información se observa la inconsistencia mencionada .

### **DESARROLLO**

El problema que origina el presente caso se encuentra resuelto por la solución dada al Requerimiento Nro. 117 "Vista Anotaciones en Calificación", luego de sucesivas pruebas se encontró que al momento de cambiar sesión entre el rol Calificador - Digitador - Calificador se estaban perdiendo datos que ocasionaban que hubiese inconsistencia en la información almacenada en la base de datos, dado que actualmente este problema no se presenta en SIR no es necesario hacer ninguna modificación al código fuente. Se hicieron las respectivas pruebas con la asesoría de Lourdes Brito, Sergio Gonzales y Javier Rocha quienes hacen parte del personal de mesa de ayuda.

### **150 Ejecutar correcciones modificación folios (Informativo)**

#### **REQUERIMIENTO**

Se efectuó prueba con el turno 2009-368-6-2513 con matrícula 368-9644 con certificado asociado 2009-368-1-1533 se agregó una anotación en el folio asociado se avanzó hasta finalizarlo, luego este número de matrícula se asoció al turno de corrección 2009-368-3- 205 en la fase de revisión y análisis se le asignó el permiso de ciudadanos y número de documento , en la fase de ejecutar corrección se corrigió en la anotación 5 el nombre de María Transito por María Carmen y se eliminó el ciudadano de Banco Bogotá, a su vez el nombre de María Transito se encontraba en el folio 368-10891 anotación 4 el cual fue corregido, así mismo en el folio de matrícula 368-9644 en anotación 5 se corrigió el número de oficio 436 del 09/04/2008 por el 529 del 10/04/2008 .El turno de corrección se avanzó hasta la fase de entrega se finalizó y no se observó inconveniente alguno para avanzarlo .

En esta prueba se observa que el sistema realiza cambios en el folio 368-10891 en la anotación 04 respecto al número de oficio y fecha en el cual en ningún momento se efectúo modificación en estos campos para este folio, se debe corregir esta inconsistencia en el sistema.

### **DESARROLLO**

Se hicieron las pruebas correspondientes tanto en el ambiente de producción como en el de desarrollo para replicar el problema planteado, para tal fin se deja constancia de los turnos creados en el ambiente de producción 2010-999-6-358 folio 999-6347, 2010-999-6- 354 folio 999-876, 2010-999-3-246, 2010-999-3-247 en los cuales se modificaron los siguientes datos cc 49392919 - casas armando y escritura 245-oficio 123456 y los turnos creados en el ambiente de desarrollo 2010-999-3-235, 2010-999-3-236, 2010-999-3-237 y folios 999-142 y 999-175.

Las pruebas llevadas a cabo no arrojaron el problema manifestado y el personal de mesa de ayuda no tiene reportes de que dicho problema se haya vuelto a presentar. Debido a las múltiples modificaciones realizadas al código fuente de SIR para dar solución a los diferentes requerimientos que surgen en el día a día es de entender que lo más acertado es suponer que alguna de dichas soluciones haya corregido el presente caso y por tal razón este problema no se presenta actualmente en SIR.

Las pruebas fueron hechas con la asesoría de Lourdes Brito quien hace parte del personal de Mesa de Ayuda.

## **151 Entrega consulta compleja o exenta**

### **REQUERIMIENTO**

Analizando el proceso consultas (compleja o exenta) en el rol usuario operativo consultas, se radican consultas por la opción iniciar, estas pasan a la fase consulta compleja o exenta, al seleccionar el turno, en seguida se hace la consulta por los criterios requeridos y se anexa si hay resultados, después de ejecutar y anexar todas búsquedas y dar clic con el botón aceptar, el turno debe quedar en la fase “ENTREGADO”. Teniendo en cuenta esta observación se debe eliminar la opción “entrega consulta compleja o exenta” del rol usuario operativo consultas, además la opción de entrega masiva de consultas en pantallas administrativas se eliminaría, por lo tanto debe crearse un procedimiento para finalizar todos los turnos de consultas complejas o exentas que se encuentran en la fase de “ENTREGAR CONSULTA COMPLEJA O EXENTA” y dejarlos en la fase de “ENTREGADO”.

### **DESARROLLO**

Se realizaron los ajustes pertinentes. Sin embargo, se hace la aclaración que la opción de entrega masiva de consultas en pantallas administrativas no se eliminó dado que desde esta pantalla también es posible entregar consultas simples con valor.

Los siguientes scripts deben ejecutarse antes de probar los cambios hechos al aplicativo, para que entregue todos los turnos que aparecen en la fase “ENTREGAR CONSULTA COMPLEJA O EXENTA” dado que esta fase ya no hará parte del proceso.

- @author: Henry Gómez Rocha.
- @change: Este update cambia el flujo del aplicativo para que una vez
- terminada la fase “CONSULTA COMPLEJA O EXENTA” del rol
- “USUARIO OPERATIVO CONSULTAS” los turnos queden
- entregados.
- Mantis: 5166 Acta - Requerimiento No 151.

```

update sir.sir_wf_procesos set id_fase_final = 'END' where id_fase_final =
'CON_ENTREGAR_COMPLEJA'

-- @author: Henry Gómez Rocha y Guillermo Cabrera.
-- @change: Este procedimiento entrega todos los turnos que se
--          encuentran en la fase "ENTREGAR CONSULTA COMPLEJA O
--          EXENTA" del rol "USUARIO OPERATIVO CONSULTAS".
-- Mantis: 5166 Acta - Requerimiento No 151.

create or replace
PROCEDURE SP_TERMINA_TURNOS_CONSULTAS_EX
IS
CURSOR C_TURNOS(P_CIRCULO VARCHAR2)IS
SELECT TU.TRNO_ID_WORKFLOW, TU.TRNO_LAST_ID_HISTORIA, TU.ANIO,
TU.ID_CIRCULO, TU.ID_PROCESO,
TU.ID_TURNO
FROM SIR.SIR_OP_TURNO TU INNER JOIN
SIR.SIR_OP_TURNO_EJECUCION TE ON TU.TRNO_ID_WORKFLOW=TE.TREJ_ITEM_KEY
WHERE TREJ_FASE='CON_ENTREGAR_COMPLEJA'
AND tu.id_circulo = P_CIRCULO
AND TU.id_proceso = '2';
C_CIRCULOS SYS_REFCURSOR;
D_TURNOS C_TURNOS%ROWTYPE;
P_CIRCULO VARCHAR2(20);
N_USUARIO NUMBER;
ERROR_NUM NUMBER:=0;
ERROR_MSG VARCHAR2(1000):=NULL;
BEGIN

OPEN C_CIRCULOS FOR
    SELECT CI.ID_CIRCULO
    FROM SIR.SIR_NE_CIRCULO CI
    WHERE CI.CRCL_FECHA_PROD IS NOT NULL;
LOOP
    FETCH C_CIRCULOS INTO P_CIRCULO;
    EXIT WHEN C_CIRCULOS%NOTFOUND;
    OPEN C_TURNOS(P_CIRCULO);
    LOOP
        FETCH C_TURNOS INTO D_TURNOS;
        EXIT WHEN C_TURNOS%NOTFOUND;

        BEGIN
            SELECT TH.ID_USUARIO INTO N_USUARIO FROM
SIR.SIR_OP_TURNO_HISTORIA TH
            WHERE TH.ANIO= D_TURNOS.ANIO
            AND TH.ID_CIRCULO = D_TURNOS.ID_CIRCULO
            AND TH.ID_PROCESO=D_TURNOS.ID_PROCESO
            AND TH.ID_TURNO=D_TURNOS.ID_TURNO
            AND TH.ID_TURNO_HISTORIA = D_TURNOS.TRNO_LAST_ID_HISTORIA;

            UPDATE SIR.SIR_OP_TURNO TU
            SET TU.TRNO_ID_FASE = 'FINALIZADO',
            TU.TRNO_LAST_ID_HISTORIA = TU.TRNO_LAST_ID_HISTORIA+1,
            TU.TRNO_FECHA_FIN=SYSDATE
            WHERE TU.TRNO_ID_WORKFLOW = D_TURNOS.TRNO_ID_WORKFLOW;

            UPDATE SIR_OP_TURNO_EJECUCION TE
            SET TE.TREJ_FASE = 'FINALIZADO',

```

```

TE.TREJ_ESTACION = 'FINALIZADO',
TE.TREJ_NOTIFICATION_WF=0,
TE.TREJ_ESTADO = 2
WHERE TE.TREJ_ITEM_KEY = D_TURNOS.TRNO_ID_WORKFLOW;

INSERT INTO SIR.SIR_OP_TURNO_HISTORIA TUH
(TUH.ID_TURNO_HISTORIA, TUH.ID_TURNO,TUH.ID_USUARIO,
TUH.TRHS_ID_FASE,TUH.TRHS_FECHA_REGISTRO,
TUH.ID_CIRCULO, TUH.ID_PROCESO,
TUH.ANIO,TUH.TRHS_ACTIVO, TUH.TRHS_FECHA_CREACION,
TUH.TRHS_RESPUESTA, TUH.TRHS_STACK_POS,
TUH.TRHS_ID_FASE_ANTERIOR,TUH.TRHS_ID_ADMINISTRADOR_SAS,TUH.ID_USUARIO
1)
VALUES(D_TURNOS.TRNO_LAST_ID_HISTORIA+1, D_TURNOS.ID_TURNO,
N_USUARIO, 'FINALIZADO', SYSDATE, D_TURNOS.ID_CIRCULO,
D_TURNOS.ID_PROCESO,D_TURNOS.ANIO,
1, SYSDATE, NULL, 0, NULL, 'FINALIZADO',NULL);
COMMIT;
EXCEPTION
WHEN OTHERS THEN
    ERROR_NUM:=SQLCODE;
    ERROR_MSG:=SQLERRM;
    DBMS_OUTPUT.PUT_LINE('Error al
entregar el turno: '||D_TURNOS.TRNO_ID_WORKFLOW||' '|SQLCODE||'|'||SQLERRM);
    ROLLBACK;
END;
END LOOP;
CLOSE C_TURNOS;
END LOOP;
CLOSE C_CIRCULOS;
END;

```

#### CLASE o ARCHIVO JSP:

C:\WORK\SIR-APP\src\gov\sir\core\web\navegacion\comun\ControlNavegacionSeguridad.java

#### Método:

**public String procesarNavegacion(HttpServletRequest request)**

Se adicionan las siguientes líneas al método (Línea Nro. 182).

```

/**
 * @author: Henry Gómez Rocha.
 * @change: Se remueve del rol "USUARIO OPERATIVO CONSULTAS" la fase
 *          "ENTREGAR CONSULTA COMPLEJA O EXENTA".
 * Mantis: 5166 Acta - Requerimiento No 151.
 */

```

```

if(CRol.SIR_ROL_USUARIO_OPERATIVO_CONSULTAS.equals((Rol)
session.getAttribute(WebKeys.ROL)).getRoleId())){
    fases.remove(1);
}

```

#### CLASE o ARCHIVO JSP:

C:\WORK\SIR-APP\src\gov\sir\core\negocio\modelo\constantes\CRol.java

Se adicionan las siguientes líneas al método (Línea Nro. 57).

```

/**
 * @author: Henry Gómez Rocha.
 * @change: Se crea la variable "SIR_ROL_USUARIO_OPERATIVO_CONSULTAS" para poder remover
 *          del rol "USUARIO OPERATIVO CONSULTAS" la fase "ENTREGAR CONSULTA COMPLEJA O
 *          EXENTA".
 * Mantis: 5166 Acta - Requerimiento No 151.
 */
public static final String SIR_ROL_USUARIO_OPERATIVO_CONSULTAS =
    "SIR_ROL_USUARIO_OPERATIVO_CONSULTAS";

```

## 152 Notas informativas Mayor Valor (Informativo)

### **REQUERIMIENTO**

En las notas informativas de mayor valor en la fase de calificación, revisión y análisis del proceso de correcciones -cuando se selecciona tipo de nota 5010-Mayor Valor (Tiempo Limite de pago) el sistema muestra un cuadro con texto donde fundamenta el pago del mayor valor, en este texto se debe ajustar la normatividad de tarifas, lo correcto es Decreto No. 2280 de 2008 –Resolución No.0035 de Enero 09 de 2009 y no Decreto No.1428 del 26 de julio de 2000- Resolución No.5123 de Noviembre 9 de 2000.

### **DESARRROLLO**

Se resuelve modificar este tipo de nota informativa.

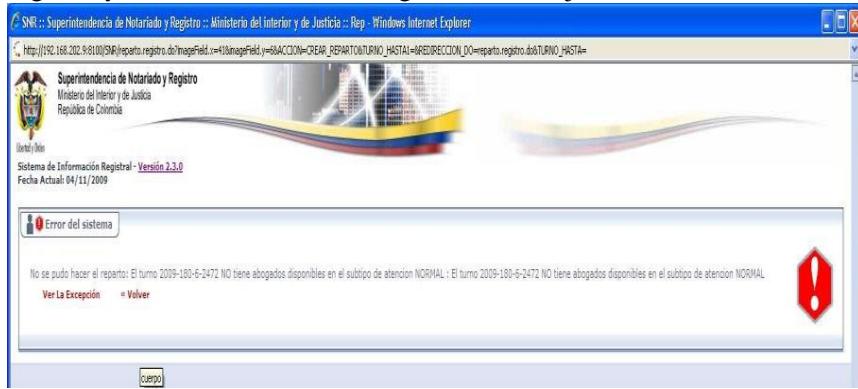
SQL:

```
update sir.sir_op_tipo_nota
set t.tpna_descripcion = 'FALTA PAGO DE DERECHOS DE REGISTRO. SENOR
USUARIO TIENE UN PLAZO LIMITE DE DOS (2) MESES, A PARTIR DE LA FECHA
DE RADICACION DEL DOCUMENTO, PARA CANCELAR EL MAYOR
VALOR. (Decreto No. 2280 de 2008 –Resolución No.0035 de Enero 09 de 2009 y no
Decreto No.1428 del 26 de julio de 2000- Resolución No.5123 de Noviembre 9 de 2000.)'
where t.id_tipo_nota='5010';
```

## 155 Reparto registro -Subtipos de atencion

### **REQUERIMIENTO**

Se efectúo prueba seleccionando subtipos de atención para calificadores de copias, gran tamaño, testamentos, reglamentos para usuarios activos, se realizo el reparto de turnos de registro y el sistema muestra el siguiente mensaje :



El mensaje debe ajustarse: no debe mostrar los turnos, sino debe decir algo general, “No existen subtipos de atención para calificadores del tipo (copias, gran tamaño, reglamento, testamento, normal según sea el caso) “.

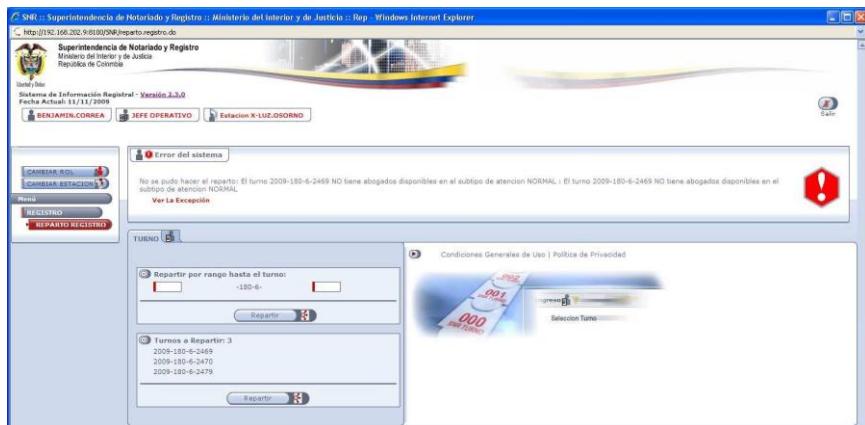
En ambiente de pruebas se efectuó reparto de la Orip de Quibdo con usuarios inactivos y seleccionando todos los subtipos de atención para calificadores a excepción de la opción normal o seleccionando únicamente normal, el sistema muestra el mensaje coherente sin embargo lo repite y genera confusión y permite volver a la pantalla donde se realiza el reparto mostrando el siguiente mensaje “No se pudo hacer el reparto: El subtipo de atención NORMAL NO tiene calificadores con estaciones disponibles, se debe habilitar por lo menos un calificador con su respectiva estación : El subtipo de atención NORMAL NO tiene calificadores con estaciones disponibles, se debe habilitar por lo menos un

calificador con su respectiva estación “

**En este texto se debe corregir la palabra calicadores por calificadores.**

“No se pudo hacer el reparto: No existen subtipos de atención para calicadores del tipo: GRAN TAMAÑO. : No existen subtipos de atención para calicadores del tipo: GRAN TAMAÑO”

Se realizo prueba de reparto de documentos para usuario activos y sin seleccionar subtipos de atención para calicadores el sistema no permite realizar el reparto validez que es correcta y muestra mensaje “No se pudo hacer el reparto: El turno 2009-180-6- 2469 NO tiene abogados disponibles en el subtipo de atención NORMAL : El turno 2009- 180-6-2469 NO tiene abogados disponibles en el subtipo de atención NORMAL” en este mensaje se observa que el sistema lo duplica lo correcto es que no debe repetir el texto del mensaje.



Se requiere que el mensaje sea general como se solicito anteriormente, no debe mencionar ningún turno.

Se expuso en reunión de seguimiento del Aplicativo SIR respecto a las inconsistencias encontradas en la revisión del ajuste realizado a la funcionalidad de Reparto Registro cuando no se seleccionan subtipos de atención y se intenta hacer reparto con usuarios activos e inactivos y que el requerimiento como tal se realizó, sin embargo se acordó la prioridad de desarrollar los ajustes a las inconsistencias halladas en la revisión del despliegue de esta fecha .

**DESARROLLO**

- Script Bases de Datos:
  - NINGUNA
- Creación De Clases:
  - NINGUNA

<b>CLASE o ARCHIVO JSP:</b>	
<b>C:\WORK\SIR-APP\src\gov\sir\core\negocio\acciones\registro\ANReparto.java</b>	
<b>Método:</b>	
<b>Código Línea 669:</b>	
throw new EventoException("No se pudo hacer el reparto: " + t.getMessage(),t);	
<b>Por:</b>	
/* * @author : Henry Gómez Rocha * @change : Se modifica el número de parametros enviados para evitar que se imprima dos veces el mismo mensaje.	

```

    * Caso Mantis : 0005012
    */
throw new EventoException("No se pudo hacer el reparto: " + t.getMessage());

```

### Código Línea 771:

```

if(abogados==null){
    throw new EventoException("El turno "+t.getIdWorkflow()+" NO tiene abogados disponibles en el
    subtipo de atencion "+sr.getSubtipoAtencion().getNombre());
}

if(abogados.size()==0){
    throw new EventoException("El turno "+t.getIdWorkflow()+" NO tiene abogados disponibles en el
    subtipo de atencion "+sr.getSubtipoAtencion().getNombre());
}

```

#### Por:

```

/*
 * @author   : Henry Gómez Rocha
 * @change   : Se unen dos if y así mismo, se modifica el mensaje evitando que se muestre en la
             excepción el turno y a su vez se hace más claro el mensaje final para el usuario.
 * Caso Mantis : 0005012
 */
if(abogados==null || abogados.size()==0){
    throw new EventoException("No existen subtipos de atención para calificadores del tipo
    "+sr.getSubtipoAtencion().getNombre());
}

```

### Código Línea 863:

```

while(llaves.hasNext()){
    SubtipoAtencion subtipo = (SubtipoAtencion)llaves.next();
    Long turnosPorSubtipo = (Long)numeroTurnosPorSubtipo.get(subtipo);
    if(mapSubtipoUsuarios.containsKey(subtipo)){
        List abogados = (List)mapSubtipoUsuarios.get(subtipo);
        numeroAbogados = abogados.size();
        if(numeroAbogados==0){
            throw new EventoException("El subtipo de atención "+subtipo.getNombre()+" NO tiene
            calificadores con estaciones disponibles, se debe habilitar por lo menos un calificador con su
            respectiva estación");
        }
        promedio = turnosPorSubtipo.longValue() / numeroAbogados;
        if(turnosPorSubtipo.longValue()%numeroAbogados!=0){
            promedio++;
        }
    }
    else{
        //El -1 indica que no existen usuarios del subtipo de atención indicado
        //que puedan atender turnos de ese subtipo de atención
        promedio = -1;
    }

    //Se ingresa el promedio en la tabla
    rta.put(subtipo, new Long(promedio));
}

```

#### Por:

```

/*
 * @author   : Henry Gómez Rocha
 * @change   : Se crea la variable evitaDosImpresiones con el fin de evitar
             que el mismo mensaje sea presentado al usuario final dos veces.
 * Caso Mantis : 0005012
 */
boolean evitaDosImpresiones = true;
while(llaves.hasNext()){
    SubtipoAtencion subtipo = (SubtipoAtencion)llaves.next();
    Long turnosPorSubtipo = (Long)numeroTurnosPorSubtipo.get(subtipo);
    if(mapSubtipoUsuarios.containsKey(subtipo)){
        List abogados = (List)mapSubtipoUsuarios.get(subtipo);
        numeroAbogados = abogados.size();
        if(numeroAbogados==0 && evitaDosImpresiones){
            evitaDosImpresiones = false;
            throw new EventoException("El subtipo de atención "+subtipo.getNombre()+" NO tiene
            calificadores con estaciones disponibles, se debe habilitar por lo menos un calificador con su
            respectiva estación");
        }
        promedio = turnosPorSubtipo.longValue() / numeroAbogados;
        if(turnosPorSubtipo.longValue()%numeroAbogados!=0){
            promedio++;
        }
    }
}

```

```

    }
    else{
        //El -1 indica que no existen usuarios del subtipo de atención indicado
        //que puedan atender turnos de ese subtipo de atención
        promedio = -1;
    }

    //Se ingresa el promedio en la tabla
    rta.put(subtipo, new Long(promedio));
}

```

### Código Línea 948:

```

if(abogados == null){
    throw new EventoException("No existen subtipos de atención para calicadores del tipo: " +
        sr.getSubtipoAtencion().getNombre());
}

Por:
/*
 * @author : Henry Gómez Rocha
 * @change : Se corrige la palabra calicadores por calificados.
 * Caso Mantis : 0005012
 */
if(abogados == null){
    throw new EventoException("No existen subtipos de atención para calificados del tipo: " +
        sr.getSubtipoAtencion().getNombre());
}

```

## 156 Reporte 6- Diario Radicador de Certificados del dia

### REQUERIMIENTO

Cuando se anulan turnos de certificados asociados a turnos de registro en la columna de turnos asociados debe listarse el turno de registro con el que se encontraba asociado.

### DESARROLLO

- Script Bases de Datos:

**SP\_REPORTE\_06.sql**

```

*****
/** AHERREN 25/02/2010 */
/** En el campo Asociado tenian la validacion:
    DECODE(t_tur(tur).trno_anulado,'S','ANULADO',v_asociado)
    en la cual estaba reemplazando el Numero del turno asociado por la palabra
    anulado se elimina validacion por requerimiento 156
    y se deja solo el numero del turno asociado
*****
```

```

VALUES(TO_CHAR(t_tur(tur).trno_fecha_inicio,'HH24:MI:SS'), v_turno_ant, t_tur(tur).trno_id_workflow, v_matricula,
v_nombre, DECODE(t_tur(tur).trno_anulado, 'S',NULL,t_sol(sol).slct_numero_copias_cr), t_liq(lig).lqdc_valor,
t_pag(pag).id_usuario , v_asociado , t_pag(pag). pago_num_recibo,
DECODE(t_tur(tur).trno_anulado,'S','ANULADO',v_tipo), DECODE(t_tur(tur).trno_anulado,'S','ANULADO',v_estado),
v_num_reimp, DECODE(t_tur(tur).trno_anulado,'S','ANULADO',NULL), p_sec_proceso,t_sol(sol).id_circulo);

```

**SP\_REPORTE\_06\_Marcha\_Atras.sql** (Archivo Adjunto)

- Creación De Clases:

NINGUNO

- Modificación de archivos:

NINGUNO

## 157 Reporte 08 Diario Radicador de Consultas

### REQUERIMIENTO

Se realizo prueba generando turnos de consultas con tarifa 2009-068-2-88, 2009-068-2-89, 2009-068-2-90, 2009-068-2-91, 2009-068-2-92 y turnos de consultas exentas 2009-068-2- 95, 2009-068-2-96, 2009-068-2-97 se probó tratando de anular turnos de consulta exentos antes y después de ejecutar la consulta y el sistema valida correctamente que no se pueden anular, se anularon los turnos 2009-068-2-93 y 2009-068-2-94, se generó el reporte 08 DIARIO RADICADOR DE CONSULTAS y se observa que el sistema no suma el valor de los turnos anulados con los demás valores en la columna de recaudo. En la

columna de solicitante seguido del nombre se observa el texto “turno anulado” frente a los turnos que se anularon.

Para mejorar la comprensión de la información reflejada en el reporte 08 la SNR gestiona formulario de requerimiento para crear una columna con título Tipo de Consulta y en esta se refleje el tipo de consulta “ Exenta –EXENTA, Normal-NORMAL

## **DESARROLLO**

- Script Bases de Datos:

SP\_Reporte\_08.sql

```
*****
/** AHERRENO 25/02/2010 */
/** Requerimiento 157 */
/** En el campo Observaciones estaba asociando al campo slct_comentario
de la tabla solicitud el cual no era tenido en cuenta en el reporte
por tal razón se asocio a observaciones el campo id_tipo_consulta
de la tabla SIR_OP_TIPO_SOLICITUD; campo necesario para el reporte.
*****
```

VALUES  
(TO\_CHAR(t\_tur(tur).trno\_fecha\_inicio,DD-MM-YYYY HH:MI:SS), TO\_CHAR(t\_tur(tur).trno\_fecha\_inicio,HH:MI:SS  
AM), t\_tur(tur).trno\_id\_workflow, t\_sol(sol).id\_tipo\_consulta, v\_nombre, t\_sol(sol).slct\_num\_max\_busquedas\_cn,  
NVL(t\_liq(liq).lqdc\_valor,0), t\_pag(pag). pago\_num\_recibo, t\_tur(tur).trno\_anulado, p\_sec\_proceso,t\_sol(sol).id\_usuario);

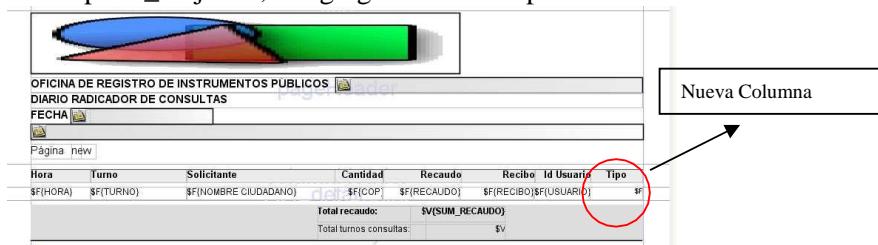
SP\_Reporte\_08\_Marcha\_Atras.sql (Archivo Adjunto)

- Creación De Clases:

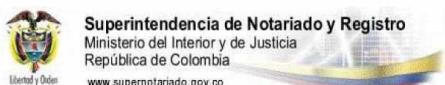
NINGUNO

- Modificación de archivos:

Modificación archivo reporte\_08.jrxml; se agrega columna Tipo



Se realizan pruebas con el circulo Bucaramanga, se radican turnos de consultas.



OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS BUCARAMANGA - 300

DIARIO RADICADOR DE CONSULTAS

FECHA 25-05-2010

Impreso el 25 de Mayo de 2010 a las 14:34:18 PM

Página 1

Hora	Turno	Solicitante	Cantidad	Recaudo	Recibo	Id Usuario	Tipo
01:53:12 PM	2010-300-2-18263	PRUEBA	0	0	70111170	52722	EXENTA
02:11:26 PM	2010-300-2-18264	PRUEBAS	1	1,100	70111171	52722	SIMPLE
02:14:50 PM	2010-300-2-18265	PRUEBAS - TURNO ANULADO	4	0	70111172	52722	COMPLEJA
02:15:16 PM	2010-300-2-18266	PRUEBAS	1	1,100	70111173	52722	CONSTANCIA
Total recaudo:							2,200
Total turnos consultas:							4

**160 Reporte 25 Y 51 Turnos anulados de fechas de radicacion anteriores**

## **REQUERIMIENTO**

Se realizan pruebas con el circulo de Bucaramanga usuario jose.garcia con el turno 2009- 300-6-38085, el cual se radico 23/10/2009 y fue anulado el 28/10 de 2009, se genero el Reporte No. 25 informe recaudo del día tiracaja del 23/10/2009, luego se procedió a generar el Reporte 51, resumen diario de ingresos y egresos, en los cuales se refleja la información de los turnos anulados, es decir cumple con el requerimiento que los turnos anulados reflejan en los reportes mencionados en la fecha de su radicación.

- En el reporte 25 muestra la columna de fecha de anulación y muestra los turnos anulados del día de su radicación y se requiere que en este resumen muestre los turnos anulados este día de fechas de radicación anteriores y que se agregue una columna donde muestre fecha de radicación.
- En el reporte 51 en el resumen por usuario y resumen general por conceptos debe crearse un ítem que diga turnos anulados tanto de (documentos, certificados, consultas, fotocopias, correcciones, devoluciones)
  - cer-anul-rad-fech-ant
  - con-anul-rad-fech-ant
  - cor-anul-rad-fech-ant
  - dev-anul-rad-fech-ant
  - fot-anul-rad-fech-ant
  - doc-anul-rad-fech-ant

## **DESARROLLO**

### **Reporte 25. (Informe Recaudos del día por usuario)**

Se agrega la columna de fecha de radicación y muestra los turnos anulados este día, de fechas de radicación anteriores.

Total Efectivo:	\$ 859,850								
Total Consignación:	\$ 0								
Total Cheque:	\$ 0								
Total Oficina Vr:	\$ 859,850								
Total Otras ORIPS Vr:	\$ 0								
<b>Vr. total aplicado:</b>	<b>\$ 859,850</b>								
<b>Resumen General por Bancos</b>									
Banco	Valor								
Total:									
<b>Resumen General por Bancos - Devoluciones</b>									
Banco	Valor								
Total:									
<b>Turnos Anulados</b>									
Radicación	Recibo	Vr. Aplicado	Banco - No Doc.	Vr.Docu.Pago	Vr. Liquidado	F. Radica.	F. Anula.	Hora Pago	Proc.
		Efectivo	Consignación	Cheque/ Nota Débito					
2010-300-6-12466	90000215	\$ 0	\$ 0	\$ 0	\$ 0	\$ 1	25/05/2010	28/05/2010	09:29:03 REG
2010-300-1-71125	90000216	\$ 0	\$ 0	\$ 0	\$ 0	\$ 1	25/05/2010	28/05/2010	09:29:03 CER-MSV
2010-300-6-12468	90000219	\$ 225,000	\$ 0	\$ 0	\$ 225,000	\$ 225,000	25/05/2010	28/05/2010	12:36:21 REG
<b>Subtotales Turnos Anulados:</b>		<b>\$ 225,000</b>	<b>\$ 0</b>	<b>\$ 0</b>	<b>\$ 225,000</b>	<b>\$ 225,000</b>			

### ➤ Script Bases de Datos:

Se modifica el script del subreporte (SUBREPORTE\_25\_TURNOSANULADOS), al cual se le agrega un UNION validando la fecha de registro

```
h.trhs_fecha_registro BETWEEN TO_DATE ($P{P_PAGO_FECHA}, 'YYYY-MM-DD') AND TO_DATE ($P{P_PAGO_FECHA})
```

### **Reporte 51 (Resumen diario Ingresos y Egresos)**

### ➤ Análisis:

- cer-anul-rad-fech-ant

Los turnos de Certificados pueden ser anulados

- con-anul-rad-fech-ant

1. Simple: El turno se puede anular siempre y cuando no se realicen b usquedas.
  2. Compleja: El turno se puede anular siempre y cuando no se realicen b usquedas.
  3. Exenta: No se puede anular
  4. No propiedad: El turno se puede anular siempre y cuando no se realicen b usquedas.

- cor-anul-rad-fech-ant

El turno no debe superar la fase de REVISION Y ANALISIS para poder ser anulado

- dev-anul-rad-fech-ant

El turno no debe superar la fase de ANALISIS para poder ser anulado

- fot-anül-rad-fecha-nt

Los turnos de fotocopias no se van a ver reflejados en los reportes ejm: el día 28/05/2010 se anulo el turno 2010-300-5-14 y no se ve reflejado por que el turno se deja anular siempre y cuando no halla superado la fase de liquidación y el reporte verifica la fecha del pago por tal razón no se va a ver reflejado

- ### • doc-anul-rad-fech-ant

El turno no debe superar la fase de REGISTRO para poder ser anulado.

#### ➤ Script Bases de Datos:

## Se modifica el procedimiento SP REPORTE 51

```

***** AHERRENO 27/05/2010 *****/
/* Se agrega condición para que evalúe los turnos que se
anulen en el día de la consulta y estos se hallan
radicado en fechas anteriores *****/
***** */

IF t(a).trno_anulado = 'S' AND TO_CHAR(p(d).pago_fecha,'YYYY-MM-DD') <> TO_DATE(p_dia,'YYYY-MM-DD') AND
TO_DATE(t(a).fecha_anulacion,'YYYY-MM-DD') <> TO_DATE(t(a).pago_fecha,'YYYY-MM-DD') THEN

    SELECT srio_nombre_de_usuario,
           srio_nombre
      ||'
      ||srio_apellido1
      ||
      ||srio_apellido2
   INTO usu
  FROM SIR_OP_USUARIO
 WHERE id_usuario = p(d).id_usuario;

-- CASE l(b).id_proceso

-- WHEN 1 THEN
--     v_concepto      := 'Cer-anul-rad-fech-ant';
-- WHEN 2 THEN
--     v_concepto      := 'con-anul-rad-fech-ant';
-- WHEN 3 THEN

```

```

v_concepto      := 'cor-anul-rad-fech-ant';

WHEN 4 THEN
v_concepto      := 'dev-anul-rad-fech-ant';
WHEN 5 THEN
v_concepto      := 'fot-anul-rad-fech-ant';
WHEN 6 THEN
v_concepto      := 'doc-anul-rad-fech-ant';
END CASE;
efectivo.valor_aplicado   :=0;

efectivo.valor_documento    :=0;
consignacion.valor_aplicado :=0;
consignacion.valor_documento :=0;
cheque.valor_aplicado      :=0;
cheque.valor_documento     :=0;
act.acto_cuantia           :=0;
act_val_imp.acto_valor_derechos :=0;
act_val_imp.acto_valor_impuestos :=0;
l(c).lqdc_valor_derechos_rg  :=0;
l(c).lqdc_valor              :=0;
l(c).lqdc_valor_derechos_rg  :=0;
l(c).lqdc_valor_otro_impuesto_rg :=0;
l(c).lqdc_valor_impuestos_rg :=0;
/*****************************************/
/*****************************************/

```

## 166\_141\_Flujo de turnos de registro fase testamento.doc

### **REQUERIMIENTO**

Flujo de turnos de registro fase testamento. Cuando un turno de registro tiene el subtipo de solicitud como testamento, avanza de la fase confrontación a la fase testamento y este mismo turno se envía nuevamente a confrontación para cambiar el subtipo de solicitud a normal y enviarlo a reparto de registro, después de ser repartido avanza a calificación. Si el calificador necesita enviarlo a corrección de encabezado, al avanzarlo nuevamente de la fase corrección de encabezado a calificación el sistema envía el turno a Testamento con la estación del calificador que había enviado el turno a corrección de encabezado. Por favor escalar este caso a desarrollo y crear acta de requerimiento a desarrollo para su solución general ya que el sistema avanza erradamente los turnos de fase, el turno de registro cuando se envía a corrección de encabezado a cualquier otra dependencia, al terminar estos procesos el turno debe volver nuevamente a calificación y su estación original.

Tomar como referencia Id Glpi 17420.

### **DESARROLLO**

Luego de sucesivas pruebas se encontró que debido a un ajuste llevado a cabo por el antiguo personal que le daba soporte a SIR se cometió el error de siempre retornar al rol Testamento desde el rol Usuario Operativo Registro indistintamente desde donde se hubiera enviado el proceso a este último rol.

Sólo fue necesario hacer esta corrección en el código fuente para que el sistema SIR funcione de la manera correcta. Las pruebas respectivas del caso las hizo Javier Rocha quien es el Jefe del Personal de Mesa de Ayuda.

Como nota se deja constancia de que este mismo inconveniente se presenta cuando se escoge la opción Enviar a Confrontación desde el rol Calificador. Se sugiere se abra acta de requerimiento para que en un posterior desarrollo se solucione dicha situación.

#### CLASE o ARCHIVO JSP:

C:\WORK\SIR-APP\src\gov\sir\core\negocio\acciones\registro\ANCalificacion.java

Método:

**Se agrega el siguiente Código en la Línea 6156:**

```
/*
 * @author : Henry Gómez Rocha
 * @change : Se verifica que al momento de finiquitar el proceso de corrección
 *            encabezado desde el rol usuario operativo registro este retorno
 *            a la estación y rol que lo envió a dicho proceso, para nuestro caso,
 *            rol calificador.
 * Caso Mantis : 0003331
 */
else if(turnoHistoriaTemp.getFase().equals(CFase.CAL_CALIFICACION)){
    testamento=false;
    estacionRevisionTestamento=turnoHistoriaTemp.getIdAdministradorSASO;
    break;
}
```

## **169 Error impresion mesa control correcciones (Informativo)**

### ***REQUERIMIENTO***

APERTURA TICKET->

Item Afectado.....

Turno: 2010-360-3-4

Matricula(s): 360-11714

Motivo Inconsistencia: Al momento de imprimir el Certificado de correcciones el sistema notifica error. El turno 2010-360-3-4 se encuentra en la fase de Mesa de Control Correcciones y al momento de imprimir el certificado de correcciones el sistema notifica error. Se adjunta imagen de error.

Impacto.....

Cantidad de Usuarios Afectados: Uno

GRUPO GUAMO

### ***DESARROLLO***

#### ***SOLUCION PUNTUAL***

*Descripción técnica:*

El folio presenta inconsistencia con un dato en la salvedad del folio.

*Descripción solución:*

Se actualiza el dato inconsistente para que se puede imprimir. El usuario trata de imprimir pero ahora se presenta otro error con la fecha límite de impresión. Se solicita que cuando se imprima el formulario nos avisen para dejar los datos originales.

```
UPDATE "SIR"."SIR_NE_SALVEDAD_FO" SET SFLO_RADICACION = 'S/N'
WHERE ID_MATRICULA ='360-11714' AND ID_SALVEDAD_FO = '1'
```

```
UPDATE "SIR"."SIR_NE_SALVEDAD_FO" SET SFLO_RADICACION = 'S/N'
WHERE ID_MATRICULA ='360-11714' AND ID_SALVEDAD_FO = '1'
```

```
UPDATE "SIR"."SIR_NE_SALVEDAD_FO" SET SFLO_RADICACION = 'S/N'
WHERE ID_MATRICULA ='360-11714' AND ID_SALVEDAD_FO = '1'
```

Actualizacion antes de la impresión:

```

update sir_op_turno_historia t
set t.trhs_fecha_registro = to_date('26/01/2010 09:40:58', 'dd/mm/yyyy hh24:mi:ss')
where t.anio = '2010'
and t.id_circulo = '360'
and t.id_proceso = 3
and t.id_turno = '4'
and t.id_turno_historia = '6'

update sir_op_turno_historia t
set t.trhs_fecha_registro = to_date('26/01/2010 09:40:59', 'dd/mm/yyyy hh24:mi:ss')
where t.anio = '2010'
and t.id_circulo = '360'
and t.id_proceso = 3
and t.id_turno = '4'
and t.id_turno_historia = '7'

update sir_op_turno_historia t
set t.trhs_fecha_registro = to_date('26/01/2010 09:41:00', 'dd/mm/yyyy hh24:mi:ss')
where t.anio = '2010'
and t.id_circulo = '360'
and t.id_proceso = 3
and t.id_turno = '4'
and t.id_turno_historia = '8'

```

Restauración de los datos originales después de la impresión:

```

update sir_op_turno_historia t
set t.trhs_fecha_registro = to_date('21/01/2010 11:58:40', 'dd/mm/yyyy hh24:mi:ss')
where t.anio = '2010'
and t.id_circulo = '360'
and t.id_proceso = 3
and t.id_turno = '4'
and t.id_turno_historia = '6'

update sir_op_turno_historia t
set t.trhs_fecha_registro = to_date('25/01/2010 09:40:58', 'dd/mm/yyyy hh24:mi:ss')
where t.anio = '2010'
and t.id_circulo = '360'
and t.id_proceso = 3
and t.id_turno = '4'
and t.id_turno_historia = '7'

update sir_op_turno_historia t
set t.trhs_fecha_registro = to_date('25/01/2010 09:42:32', 'dd/mm/yyyy hh24:mi:ss')
where t.anio = '2010'
and t.id_circulo = '360'
and t.id_proceso = 3
and t.id_turno = '4'
and t.id_turno_historia = '8'

```

Nota: Se abre acta ya que este caso se a replicado, otro de los casos en el que se presento es el ID GLPI 13410, MANTIS 2934.

Revisando y haciendo un análisis se verifico en el aplicativo y se pudo observar que este caso ya fue solucionado anteriormente con la solución dada al acta numero 140 - Salvedades en folios migrados y que salió en el despliegue de la versión 2.3.1. Por lo tanto se da como cerrado este caso.

### **171 Genero error al ingresar liquidación (Informativo)**

#### ***REQUERIMIENTO***

APERTURA TICKET->

Motivo Inconsistencia: Se comunica Sra. Mónica Robledo informando que desde que esta trabajando por el nodo 22 en ocasiones repetidas al liquidar el turno de registro el sistema le genera el siguiente error. "NO FUE POSIBLE MODIFICAR LA LIQUIDACIÓN DEL REGISTRO" Si sale del rol y vuelve a ingresar puede continuar con el proceso. Por favor revisar el proceso de radicación de turnos de registro para corregir este error ya que no es frecuente y se debe corregir este problema.

Impacto.....

Cantidad de Usuarios Afectados:1

GRUPO PEREIRA

#### ***DESARROLLO***

Después de probar en los ambientes de producción 192.168.202.22:7778/SNR y calidad 192.168.202.9:8100/SNR con el círculo pruebas no se replica el inconveniente presentado a la usuaria de Pereira. La mesa de ayuda hará seguimiento al caso, se da como cerrado este requerimiento ya que no se replica en las pruebas.

### **172 Impresion en desorden de certificados masivos (Informativo)**

#### ***REQUERIMIENTO***

APERTURA TICKET->

Item Afectado.....

Turno:2010-028-20-19

Matricula(s):028-3015,8585,14109,17142

Motivo Inconsistencia:

El usuario Aníbal Valencia nos informa que al generar este turno masivo los turnos de certificados generados fueron 2010-028-1-583 al 587 pero el sistema imprimió en desorden por lo tanto se saltan los recibos físicos, por ejemplo el turno 2010-028-1-586 salio con el consecutivo 41374524 cuando debió salir con consecutivo 41374526.

Impacto.....

Cantidad de Usuarios Afectados:1

GRUPO SONSON

#### ***DESARROLLO***

Luego de diversas pruebas para replicar el caso en el ambiente de desarrollo y producción, contando con el soporte del personal de mesa de ayuda y personal de diferentes ORIPS

que amablemente nos prestaron su colaboración, no se encontró dicho problema. Haciendo una investigación rigurosa sobre las posibles causas de este problema se encontró que es posible que se haya originado debido a un error en la secuencia de impresión de la impresora que se usó para imprimir dichos recibos, por lo tanto, se sugiere se haga mantenimiento a dicho.

## 180 Error al anular el turno

### **REQUERIMIENTO**

Turno: 2010-999-1-64 - Matricula(s): 999-7000

Motivo Inconsistencia: Después de reimprimir el certificado el sistema está permitiendo anular el turno.

Se realizan pruebas en SIR generando un certificado inmediato 2010-999-1-64, se reimprime, se intenta anular y el sistema lo anula satisfactoriamente pero por regla de negocio no debería permitir anularlo.

Id Glpi 15769

Turno:2010-999-6-31 - Matricula(s):999-814

Motivo Inconsistencia: El turno 2010-999-6-31 se devolvió de firma a calificación y el sistema permite anular el turno sin inconvenientes.

Por regla de negocio del sistema los turnos que sean avanzados de la fase mesa de control por primera vez no pueden ser anulados en el sistema, se radica el turno 2010-999-6-31 y este se avanza a la fase firma de registro, se devuelve a calificación, se anula, al momento de realizar el proceso el sistema notifica operación exitosa y deja que el turno quede anulado.

Por favor revisar por que el sistema está permitiendo anular turnos después de pasar por la fase mesa de control por primera vez.

Se adjuntan pantallazos.

Id Glpi 15768 - Grupo SNR

### **DESARROLLO**

➤ Script Bases de Datos:

- NINGUNA

➤ Creacion De Clases:

NINGUNA

Se modificaron los siguientes archivos:

CLASE:
gov.sir.core.negocio.acciones.administracion.AnTrasladoTurno.java
Método:
private EvnRespTrasladoTurno anularTurno(EvnTrasladoTurno evento) throws EventoException

**Código Insertado Línea 1130:**

```
/*JALcazar Caso Mantis 3883 Requerimiento No 180 - Error al anular turno
 * Se incluyen 2 nuevas validaciones:
 * 1) PROCESO CERTIFICADOS INMEDIATOS validar que no se permita anular turnos que *realizaron reimpresiones
 * 2) PROCESO REGISTRO validar que no se permita anular turnos de registro que se avanzaron a *la mesa de control
 */
if (turno.getIdProceso() == Long.parseLong(CProceso.PROCESO_CERTIFICADOS)){
    Solicitud solicitud = turno.getSolicitud();
    SolicitudCertificado solicitudCert = (SolicitudCertificado)solicitud;
    int numImpresionesCert = solicitudCert.getNumImpresiones();

    LiquidacionTurnoCertificado liqui = new LiquidacionTurnoCertificado();
    List liquidaciones = solicitud.getLiquidaciones();
    for(int i=0;i<liquidaciones.size();i++){
        double id = new Double(((LiquidacionTurnoCertificado)liquidaciones.get(i)).getIdLiquidacion()).doubleValue();
        if(id==solicitud.getLastIdLiquidacion()){
            liqui = (LiquidacionTurnoCertificado)liquidaciones.get(i);
        }
    }

    if(((liqui.getTipoCertificado().getIdTipoCertificado().equals(CTipoCertificado.TIPO_INMEDIATO_ID)))&&
    numImpresionesCert >1){
        throw new HermodException ("No se puede anular turnos de certificados que realizaron al menos una reimpresion. ");
    }
}

if (turno.getIdProceso() == Long.parseLong(CProceso.PROCESO_REGISTRO)){
    boolean avanza_mesa_control = false;
    List historial = turno.getHistorials();
    for(int i=0;i<historial.size();i++){
        TurnoHistoria turnoHistoria = (TurnoHistoria)historial.get(i);
        if ((turnoHistoria.getFase().equals(CFase.REG_MESA_CONTROL))){
            avanza_mesa_control = true;
            break;
        }
    }

    if(avanza_mesa_control){
        throw new HermodException ("No se puede anular turnos de registro que se avanzaron a la mesa de control. ");
    }
}
```

**183 visualización de la imagen en el Sistema de Gestión Documental (SGD)  
(Informativo)****REQUERIMIENTO**

El sistema SIR se debe permitir la visualización de la imagen que ha sido grabada en el Sistema de Gestión Documental (SGD)

- 1- Para los turnos radicados, el sistema SIR debe permitir la visualización de las imágenes del documento principal ( escritura, oficio, sentencia, entre otros), dentro de todas las fases surtidas en los diferentes procesos que se llevan a cabo en una ORIP (Consultas masivas, reparto notarial, registro, certificados de antiguo sistema, certificados de pertenencia, certificados de ampliación de tradición, correcciones), de igual forma debe permitir la consulta de las imágenes del documento principal de anotaciones que ya se encuentran definitivas y que obviamente hayan sido digitalizadas con anterioridad.
- 2- De igual forma en el momento que un usuario realice una consulta por las opciones ver detalles de folio y ver detalles del turno, el sistema debe permitir consultar la imagen del documento principal y que se encuentre digitalizado.
- 3- Los Parámetros por los cuales se debe permitir la consulta son número de matrícula y número de turno.

- 4- La consulta se debe permitir a nivel local, teniendo en cuenta que el SGD es local, las imágenes deben ser traídas del repositorio de SGD, para lo cual el sistema de gestión documental retornará una URL donde se encuentran almacenadas las imágenes requeridas

### **DESARROLLO**

No se puede desarrollar el componente en el ambiente SIR ya que esta utilidad se basaba en un api que los constructores de SGD deberían haber construido y entregado a SNR para su uso. Dicha API no fue entregada.

## 192 Reimpresión de Recibo de Fotocopias

### **REQUERIMIENTO**

Motivo Inconsistencia: En la reimpresión del recibo de fotocopias no sale la información correcta.

Se abre acta de requerimiento a solicitud del coordinador de mesa de ayuda.

### **DESARROLLO**

- Script Bases de Datos:

- NINGUNA

- Creacion De Clases:

- NINGUNA

Se modificaron los siguientes archivos:

CLASE:
gov.sir.core.negocio.acciones.administracion.ANImprimirFolio.java
Método:
private EventoRespuesta reImprimirRecibos(EvnImprimirFolio evento) throws EventoException{

### **Código Insertado Linea 268:**

```
/* JAlicazar caso Mantis 05014 Acta - Requerimiento No 192 - Reimpresión de Recibo de Fotocopias
 * turno.getSolicitud() tipo de solicitud para reimpresion Recibo Fotocopia.
 */
SolicitudFotocopia solicitud_fotocopia = null;
if (turno.getSolicitud()instanceof SolicitudFotocopia ){
    solicitud_fotocopia = (SolicitudFotocopia) turno.getSolicitud();
}
```

### **Código Insertado Linea 268:**

```
/* JAlicazar caso Mantis 05014 Acta - Requerimiento No 192 - Reimpresión de Recibo de Fotocopias
 * reorganizacion del metodo de la informacion comun en el proceso de recibo de fotocopia y los demas tipos de
 * recibo.
 */
Circulo circuloPortal = hermod.getCirculoPortal();
String idCirculoPortal = null;
if (circuloPortal != null )
{
    idCirculoPortal = circuloPortal.getIdCirculo();
}
// Bug 5253 -----
gov.sir.core.negocio.modelo.Usuario local_UsuarioGeneraRecibo;
String local_UsuarioGeneraReciboMsg;
local_UsuarioGeneraRecibo = user;
local_UsuarioGeneraReciboMsg = print_FootUtils_BuildUserName( local_UsuarioGeneraRecibo );
```

### **Código Insertado Linea 347:**

```

/* JAlcazar caso Mantis 05014 Acta - Requerimiento No 192 - Reimpresión de Recibo de Fotocopias
 * Se divide el procedo de recibo de fotocopia y los demás tipos de recibo.
 */
if(!(turno.getIdProceso() == Long.valueOf(CProceso.PROCESO_FOTOCOPIAS)) )

```

**Código Insertado Linea 411:**

```

/* JAlcazar caso Mantis 05014 Acta - Requerimiento No 192 - Reimpresión de Recibo de Fotocopias
 * else-> proceso de construcion del objeto imprimible de recibo de fotocopia.
 */
bundle = new Bundle(impRec);
}

else {
    ImprimibleFotocopiaPago_SolicitudData impRec = new ImprimibleFotocopiaPago_SolicitudData(turno,
    solicitud_fotocopia, circulo, liquidacion, pago, null, CTipoImprimible.RECIBO);
    impRec.setReimpresion(true);
    impRec.setUsuarioGeneraRecibo( local_UsuarioGeneraReciboMsg );
    impRec.setFechaImpresionServidorFormatted(fechaImpresion);
    bundle = new Bundle(impRec);
}

```

**CLASE:**

**gov.sir.core.negocio.modelo.imprimibles.ImprimibleFotocopiaPago\_SolicitudData.java**

**Método:**

**public void run()**

**Código Insertado Linea 35:**

```

/* JAlcazar caso Mantis 05014 Acta - Requerimiento No 186 - Reimpresion de consultas
 * Constante que indica que es una reimpresión.
 */
private boolean reimpresion;

```

**Código Insertado Linea 120:**

```

/* JAlcazar caso Mantis 05014 Acta - Requerimiento No 192 - Reimpresión de Recibo de Fotocopias
 * Imprime nota si es una reimpresión.
 */
if(this.reimpresion){
    this.textoReimpresion();
}

```

**Código Insertado Linea 258:**

```

/* JAlcazar caso Mantis 05014 Acta - Requerimiento No 192 - Reimpresión de Recibo de Fotocopias
 * void textoReimpresion() Imprime nota reimpresión.
 */
private void textoReimpresion() {

    String textoReimpresion = "Esta es una reimpresión del recibo ";
    this.imprimirLinea(ImprimibleConstantes.PLANO, "");
    this.imprimirLinea(ImprimibleConstantes.PLANO, textoReimpresion +
this.pago.getLastNumRecibo());
}
/* JAlcazar caso Mantis 05014 Acta - Requerimiento No 192 - Reimpresión de Recibo de Fotocopias
 * Set reimpresión.
 */
public void setReimpresion(boolean b) {
    reimpresion = b;
}

```

**193 Error en el valor del acto del formulario de segregacion**

**REQUERIMIENTO**

Turno:2010-314-6-2034

Matricula(s):314-12156 y 314-53002

Motivo Inconsistencia:

El usuario Alberto García nos informa que al realizar la segregación del folio 314-12156 en el formulario de calificación el valor del acto de la anotación 20 solo aparece un

digito (2) cuando el valor que debería aparecer es 256.233.174.

Se requiere verificar, se anexa imagen del formulario de calificación. Este caso se resolvió puntualmente, es necesario escalar este caso a desarrollo para su solución general ya que el sistema no deja en la anotación generadora del folio matriz el valor completo de la naturaleza jurídica. Por favor crear acta de requerimiento a desarrollo.

Grupo Piedecuesta.

### **DESARROLLO**

El problema se presenta en el procedimiento almacenado SIR.SP\_SG, al realizar el siguiente formato al valor ingresado en la anotación

SUBSTR(A.WBNT\_VALOR,1,INSTR(A.WBNT\_VALOR,'.')-1)

**SOLUCIÓN:** Se modifica y se deja A.WBNT\_VALOR.

Script Bases de Datos: SIR.SP\_SG

<b>PROCEDIMIENTO ALMACENADO</b>
<b>SIR. SP_SG</b>
<p><b>Código Modificado</b></p> <pre>/* @AUTOR: GUILLERMO CABRERA @CAMBIO: SE COMENTAREA EL CODIGO --SUBSTR(A.WBNT_VALOR,1,INSTR(A.WBNT_VALOR,'.')-1), Y SE DEJA A.WBNT_VALOR ya que al sacar la subcadena esta recortando erradamente el valor. @MANTIS: 4970. */ INSERT INTO SIR_NE_ANOTACION_TMP     (ID_ANOTACION_TMP, ID_MATRICULA, ID_NATURALEZA_JURIDICA,     NTTM_FECHA_REGISTRO, NTTM_COMENTARIO,     NTTM_VALOR, ID_DOCUMENTO, ID_TIPO_ANOTACION,     NTTM_RADICACION, NTTM_FECHA_RADICACION,     NTTM_ESPECIFICACION, NTTM_ORDEN,     NTTM_LAST_ID_SALVEDAD, ID_ESTADO_AN, NTTM_LAST_ID_CANCELACION_TMP,     NTTM_ORDEN_LPAD, ID_USUARIO,     NTTM_HEREDADO, NTTM_ID_WORKFLOW) (SELECT V_LAST_ID_ANOTA_TMP+1, V_ID_MATRICULA_PADRE, A.ID_NATURALEZA_JURIDICA, SYSDATE, A.WBNT_COMENTARIO, /* @AUTOR: GUILLERMO CABRERA @CAMBIO: SE COMENTAREA EL CODIGO -- SUBSTR(A.WBNT_VALOR,1,INSTR(A.WBNT_VALOR,'.')-1), Y SE DEJA A.WBNT_VALOR ya que al sacar la subcadena esta recortando erradamente el valor. @MANTIS: 4970. */ --SUBSTR(A.WBNT_VALOR,1,INSTR(A.WBNT_VALOR,'.')-1), A.WBNT_VALOR, A.ID_DOCUMENTO, 4, A.NUM_RADICACION, A.FECHA_RADICACION, (SELECT N.NTJR_NOMBRE FROM SIR_NE_NATURALEZA_JURIDICA N WHERE N.ID_NATURALEZA_JURIDICA = A.ID_NATURALEZA_JURIDICA), N_ORDEN_ANOTA+1, 0, 'V', 0, V_CADENA_CEROS  TO_CHAR(N_ORDEN_ANOTA+1), V_ID_USUARIO, 0, P_ID_WORKFLOW FROM SIR_OP_WEB_ANOTACION A WHERE A.ID_SOLICITUD = V_ID_SOLICITUD AND A.ID_WEB_SEGENG = V_ID_WEB_SEGENG;  /* @AUTOR: GUILLERMO CABRERA @CAMBIO: SE COMENTAREA EL CODIGO --SUBSTR(A.WBNT_VALOR,1,INSTR(A.WBNT_VALOR,'.')-1), Y SE DEJA A.WBNT_VALOR ya que al sacar la subcadena esta recortando erradamente el valor.</pre>

```

@MANTIS: 4970.
*/
INSERT INTO SIR_NE_ANOTACION_TMP
  (ID_ANOTACION_TMP, ID_MATRICULA, ID_NATURALEZA_JURIDICA,
   NTTM_FECHA_REGISTRO, NTTM_COMENTARIO,
   NTTM_VALOR, ID_DOCUMENTO, ID_TIPO_ANOTACION,
   NTTM_RADICACION, NTTM_FECHA_RADICACION,
   NTTM_ESPECIFICACION, NTTM_ORDEN,
   NTTM_LAST_ID_SALVEDAD, ID_ESTADO_AN, NTTM_LAST_ID_CANCELACION_TMP,
   NTTM_ORDEN_LPAD, ID_USUARIO,
   NTTM_HEREDADO, NTTM_ID_WORKFLOW)
  (SELECT V_ID_ANOTACION_DEF, V_ID_MATRICULA_PADRE,
   A.ID_NATURALEZA_JURIDICA, SYSDATE, A.WBNT_COMENTARIO,
   /*
    * @AUTOR: GUILLERMO CABRERA
    * @CAMBIO: SE COMENTAREA EL CODIGO --
    SUBSTR(A.WBNT_VALOR,1,INSTR(A.WBNT_VALOR,'.')-1), Y SE DEJA
    A.WBNT_VALOR ya que al sacar la subcadena esta recortando erradamente el valor.
    @MANTIS: 4970.
    */
   --SUBSTR(A.WBNT_VALOR,1,INSTR(A.WBNT_VALOR,'.')-1),
   A.WBNT_VALOR
   A.ID_DOCUMENTO, 4, A.NUM_RADICACION, A.FECHA_RADICACION,
   (SELECT N.NTJR_NOMBRE FROM SIR_NE_NATURALEZA_JURIDICA N
    WHERE N.ID_NATURALEZA_JURIDICA = A.ID_NATURALEZA_JURIDICA),
   V_ID_ANOTACION_DEF, 0, 'V', 0, V_ORDEN_ANOTACION_DEF,
   V_ID_USUARIO, 0, P_ID_WORKFLOW
   FROM SIR_OP_WEB_ANOTACION A
   WHERE A.ID_SOLICITUD = V_ID_SOLICITUD
   AND A.ID_WEB_SEGENG = V_ID_WEB_SEGENG);

```

## 197 Reimpresion de consultas

### **REQUERIMIENTO**

Turno:2010-999-2-93

Motivo Inconsistencia: La oficina de Purificación reporta que no le está saliendo la reimpresión de las consultas de no propiedad Se verifica en pruebas con el turno 2010- 999-2-93 y efectivamente genera el recibo y la impresión de la consulta pero no le genera la reimpresión

Grupo SNR

### **DESARROLLO**

CLASE:
<b>gov.sir.core.negocio.modelo.imprimibles. ImprimibleConstanciaNoPropiedad.java</b>
<b>Código Insertado Línea 408:</b>
<pre> /* JAlcazar caso Mantis 04493 Acta - Requerimiento No 197 - Reimpresion de consultas  * se crea el metodo get de la propiedad turno  */ public Turno getTurno() {     return turno; } </pre>
<b>CLASE:</b>
<b>gov.sir.print.server.Server.java</b>
<b>Método:</b>
<b>public void run()</b>

### Código Insertado Línea 471:

```
/* JAlcazar caso Mantis 04493 Acta - Requerimiento No 197 - Reimpresion de consultas
 * set del turno al imprimible
 * imp.setTurno(impConstanciaNoProp.getTurno().getIdWorkflow())
 */
if (imprimibleTemp instanceof ImprimibleConstanciaNoPropiedad ) {
    ImprimibleConstanciaNoPropiedad impConstanciaNoProp = (ImprimibleConstanciaNoPropiedad)
imprimibleTemp;
    if (impConstanciaNoProp != null && impConstanciaNoProp.getTurno() != null &&
impConstanciaNoProp.getTurno().getIdWorkflow() != null) {

        imp.setTurno(impConstanciaNoProp.getTurno().getIdWorkflow());
    }
}
```

## 201 Impuesto del Meta - Puerto Lopez

### REQUERIMIENTO

Motivo Inconsistencia: Se comunica la Sra. Margarita Ortiz informando que en el momento de generar la liquidación del documento no está cobrando el impuesto. Ver imagen.

Grupo Puerto López

The screenshot shows a web browser window with the URL <http://192.168.202.22:7778/>. The page title is "Superintendencia de Notariado y Registro :: Ministerio del". The main content area has a blue header "Liquidación" with a green circular icon. Below it, there's a section titled "Actos Asociados" with a sub-section "Nuevo Acto". The "Nuevo Acto" form contains fields for "Tipo" (set to "i"), "Cantía o autovalor" (set to "20000000"), "Descripción" (set to "VENTA - 1"), "Tarifa" (set to "NORMAL" and "PORCENTUAL"), and a checkbox "Impuesto?" which is unchecked. Below this, there's a section "Otros Impuestos" with a "Valor" field and a "Descripción" field. At the bottom, there's a "Datos Solicitante" section with fields for "Tipo de Identificación" (set to "SECUENCIA"), "Nombres / Razón Social", "Número" (set to "1"), and "Teléfono". A red error message box at the top left says "Error en la Validación de Datos del Formulario" and "Los Siguientes Errores han sido encontrados en el Formulario: Tipo de tarifa inválida".

### DESARROLLO

En el código se encuentra quemados los códigos de las oficinas del meta, volviéndose difícil de mantener al momento de migrar nuevas oficinas del meta.

Se corrige los números de los códigos de los círculos habilitados para el impuesto del meta, ahora serán llamados desde la propiedad `gov.sir.hermod.circulos.impuesto.meta` del archivo `gov.sir.hermod.properties` dentro de `mapas.jar`.

### CLASE o ARCHIVO JSP:

C:\WORK\SIR-

APP\src\gov\sir\core\web\helpers\registro\AgregarActosHelper.java

### Método:

```
public void drawGUI(HttpServletRequest request, JspWriter out)
throws HelperException, IOException
```

### Línea 79:

```
/**
```

```

        * @author: Fernando Padilla Velez
        * @change: Se modifica para el caso MANTIS 4729, Acta - Requerimiento No 202 -
      Impuesto Meta (Puerto Lopez),
        *           Se corrige los numeros de los codigos de los circulos habilitados para el
      impuesto del meta, ahora serán
        *           llamados desde la propiedad gov.sir.hermod.circulos.impuesto.meta del
      archivo gov.sir.hermod.properties dentro de mapas.jar
        */
    HermodProperties hp = HermodProperties.getInstancia();
    String circulos =
hp.getProperty(HermodProperties.HERMOD_CIRCULOS_IMPUESTO_META);
    for (TipoImpuesto t : tipoImp) {
        for (TipoImpuestoCirculo tt : tipoImpCir) {
            Date fechaInicioVigencia =
gov.sir.core.util.DateFormatUtil.parse(gov.sir.core.util.DateFormatUtil.format(tt.getFechaInicio
Vigencia()));
            Date fechaFinVigencia =
gov.sir.core.util.DateFormatUtil.parse(gov.sir.core.util.DateFormatUtil.format(tt.getFechaFinVig
encia()));
            if (tt.getIdTipoImpuesto().equals(t.getIdTipoImpuesto()) &&
tt.getIdCirculo().equals(this.circulo.getIdCirculo()) && (hoy.after(fechaInicioVigencia) ||
hoy.equals(fechaInicioVigencia))
                && (hoy.before(tt.getFechaFinVigencia()) || hoy.equals(fechaFinVigencia)))
{
                if((circulos.indexOf(circulo.getIdCirculo()) != -1 ) &&
"IM".equals(t.getIdTipoImpuesto()) &&
!"38".equals(request.getSession().getAttribute(CActo.TIPO_ACTO))){
                    continue;
                }
                elem.add(new ElementoLista(t.getIdTipoImpuesto(), t.getNombre()));
            }
        }
    }
}

```

#### **CLASE o ARCHIVO JSP:**

**C:\WORK\SIR-APP\src\gov\sir\hermod\HermodProperties.java**

**Se agregaron las siguiente linea al final de clase:**

```
public static final String HERMOD_CIRCULOS_IMPUESTO_META = "gov.sir.hermod.circulos.impuesto.meta";
```

#### **CLASE o ARCHIVO JSP:**

**C:\WORK\SIR-APP\src\properties\produccion\gov.sir.hermod.properties**

**Se agregaron las siguiente linea:**

**Código Línea 86:**

```
gov.sir.hermod.circulos.impuesto.meta=230,232,234
```

*Quedo listo el ajuste en la pantalla administrativa "Administración de Círculos Registrales" para que se puedan asignar círculos como cobradoras de impuestos.*

*Solución: Se creo un trigger "TGR\_ACT\_SIR\_NE\_CIRCULO" para que cuando se modifique la tabla "SIR\_NE\_CIRCULO" específicamente el campo CRCL\_COBRO\_IMPUESTO se cambiado a 1 se creen 3 registros en la tabla SIR\_OP\_TIPO\_IMPUESTO\_CIRCULO. con esto queda configurado completamente el circulo como generadora de impuestos.*

*Se adjunta el archivo TGR\_ACT\_SIR\_NE\_CIRCULO.sql con la creación del trigger.*

#### **206 Formato Nota Devolutiva**

#### **REQUERIMIENTO**

#### **NUEVO FORMATO NOTA DEVOLUTIVA**

Por Disposición del oficio SN2010IE012659, de la Superintendencia Delegada para el Registro la Doctora Zaida Barrero de Noguera solicita sea modificado el reporte de la Nota Devolutiva que opera en el sistema SIR.

Para dar cumplimiento a lo ordenado en el citado oficio, se requiere que el reporte de NOTA DEVOLUTIVA, generada por el rol de Calificador en el proceso de registro sea el siguiente:

**OFICINA REGISTRO INSTRUMENTOS PUBLICOS (NOMBRE CIRCULO  
REGISTRAL)  
NOTA DEVOLUTIVA**

Pagina (numero de pagina)

Impresa el (dia) de (mes) de (año) a las (hh:mm:ss am/pm)

El documento (tipo de documento) Nro (número de documento) de (oficina y ciudad origen del documento) fue presentado para su inscripción como solicitud de registro de documentos con Radicacion: (turno de radicación del documento) vinculado a la Matricula Inmobiliaria: (numero de matricula) y Certificado de Tradición (certificado asociado)

Conforme con el principio de Legalidad Registral previsto por el articulo 37 del Decreto Ley 1250 de 1970 (Estatuto del Registro de Instrumentos Publicos), se rechaza sin registrar el citado documento por las siguientes razones y fundamentos de derecho:

(causa uno)

...

(causa n)

UNA VEZ SUBSANADA(S) LA(S) CAUSAL(ES) QUE MOTIVO LA NEGATIVA DE INSCRIPCION, FAVOR RADICAR NUEVAMENTE EN ESTA OFICINA, EL DOCUMENTO PARA SU CORRESPONDIENTE TRAMITE, ADJUNTANDO

LA PRESENTE NOTA DEVOLUTIVA. CUANDO LA CAUSAL O LAS CAUSALES QUE RECHAZA (N) LA INSCRIPCION DEL DOCUMENTO NO SEA (N) SUBSANABLE (S), SE CONFIGURE EL PAGO DE LO NO DEBIDO, SE PRODUZCAN PAGOS EN EXCESO O SE DESISTA DEL TRAMITE, EL TERMINO PARA SOLICITAR LA DEVOLUCION DE DINERO PARA LOS DERECHOS DE REGISTRO ES DE CUATRO (4) MESES CALENDARIO, CONTADOS A PARTIR DE LA FECHA DE EJECUTORIA DEL ACTO ADMINISTRATIVO QUE NIEGA EL REGISTRO O SE DESANOTE EL DOCUMENTO INSCRITO.

LOS ACTOS O NEGOCIOS JURIDICOS A QUE SE REFIERE EL ARTICULO 2 DEL ESTATUTO DEL REGISTRO DE INSTRUMETNOS PUBLICOS, DEBERAN PRESENTARSE PARA SU INSCRIPCION, DENTRO DE LOS DOS (2) MESES CALENDARIO, SIGUIENTES A LA FECHA DE SU OTORGAMIENTO PARA ACTOS NOTARIALES O LA FECHA DE EJECUTORIA PARA PROVIDENCIAS JUDICIALES O ADMINISTRATIVAS; VENCIDOS LOS CUALES, SE COBRARAN INTERESES MORATORIOS POR IMPUESTO DE REGISTRO, PREVISTOS EN LA LEY 223 DE 1995 Y SU DECRETO REGLAMENTARIO 695 DE 1996 ARTICULO 14.

EXCEPTUESE DE LO ANTERIOR, LOS CASOS RELACIONADOS CON EL NEGOCIO JURIDICO DE HIPOTECA Y EL ACTO DE CONSTITUCION DE PATRIMONIO DE FAMILIA DE QUE TRATA EL ARTICULO 32 DEL DECRETO LEY 1250 DE 1970, LOS CUALES SE DEBEN REGISTRAR DENTRO DE LOS NOVENTA

(90) DIAS HABILES SIGUIENTES A SU AUTORIZACION. VENCIDO EL TERMINO REGISTRAL ANTES SENALADO, DEBERAN CONSTITUIRSE DE CONFORMIDAD CON EL PRECITADO ARTICULO.

CONTRA EL PRESENTE ACTO ADMINISTRATIVO, PROcede EL RECURSO DE REPOSICION ANTE EL REGISTRADOR DE INSTRUMENTOS PUBLICOS Y EN SUBSIDIO, EL DE APELACION ANTE EL DIRECTOR DE REGISTRO DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO DENTRO DE LOS CINCO (5) DIAS HABILES SIGUIENTES A SU NOTIFICACION, EN VIRTUD DE LO PREVISTO POR EL NUMERAL 2 DEL ARTICULO 23 DEL DECRETO 412 DE 2007, PREVIO EL CUMPLIMIENTO DE LOS REQUISITOS ESTABLECIDOS EN LOS ARTICULOS 51 Y 52 DEL DECRETO 01 DE 1984 (CODIGO CONTENCIOSO ADMINISTRATIVO).

---

#### FIN DE ESTE ACTO ADMINISTRATIVO

#### NOTIFICACION PERSONAL

CONFORME LO DISPUESTO EN EL ARTICULO 44 DEL CODIGO CONTENCIOSO ADMINISTRATIVO, EN LA FECHA \_\_\_\_\_ SE NOTIFICO PERSONALMENTE EL PRESENTE ACTO ADMINISTRATIVO A

SE IDENTIFICO CON \_\_\_\_\_ QUIEN  
No. \_\_\_\_\_

FUNCIONARIO NOTIFICADOR

EL NOTIFICADO “

NOTA: SE ANEXA REPORTE EXPEDIDO EN LA APLICACIÓN FOLIO..

#### DESARROLLO

<b>CLASE o ARCHIVO JSP:</b>
<b>gov.sir.core.negocio.modelo.imprimibles.ImprimibleNotaDevolutivaCalificacion.java</b>
<b>Modificacion de metodos</b>
<b>public void generate(PageFormat pageFormat)</b>

**Línea 103:**

```
/**  
 * @author Fernando Padilla Velez  
 * @change Mantis 5423: Acta - Requerimiento No 206 - FORMATO NOTA  
DEVOLUTIVA, Se modifica el marco legal.  
 */
```

*this.imprimirLinea(ImprimibleConstantes.TITULO2,ImprimibleConstantes.MARGEN\_IZ  
Q,"Conforme con el principio de Legalidad Registral previsto por el articulo 37 del Decreto Ley*

1250 de 1970 (Estatuto del Registro de Instrumentos Publicos), se rechaza sin registrar el citado documento por las siguientes razones y fundamentos de derecho:";135);

**Línea 147:**

```
/**  
 * @author Fernando Padilla Velez  
 * @change Mantis 5423: Acta - Requerimiento No 206 - FORMATO NOTA  
DEVOLUTIVA.  
 * Se modifican los parrafos segun acata de requerimiento y se agrega un nuevo  
parrafo para su impresion  
 * String descDevolutiva4 la cual se imprime tambien.  
 */  
String descDevolutiva = "UNA VEZ SUBSANADA(S) LA(S) CAUSAL(ES) QUE MOTIVO  
" +  
"LA NEGATIVA DE INSCRIPCION, FAVOR RADICAR NUEVAMENTE EN ESTA  
OFICINA, " +  
"EL DOCUMENTO PARA SU CORRESPONDIENTE TRAMITE, ADJUNTANDO LA  
PRESENTE NOTA DEVOLUTIVA.";  
  
String descDevolutiva1 = "CUANDO LA " +  
"CAUSAL O LAS CAUSALES QUE RECHAZA (N) LA INSCRIPCION DEL  
DOCUMENTO " +  
"NO SEA (N) SUBSANABLE (S), SE CONFIGURE EL PAGO DE LO NO DEBIDO,  
SE PRODUZCAN " +  
"PAGOS EN EXCESO O SE DESISTA DEL TRAMITE, EL TERMINO PARA  
SOLICITAR LA DEVOLUCION " +  
"DE DINERO PARA LOS DERECHOS DE REGISTRO ES DE CUATRO (4) MESES  
CALENDARIO, CONTADOS A " +  
"PARTIR DE LA FECHA DE EJECUTORIA DEL ACTO ADMINISTRATIVO QUE  
NIEGA EL REGISTRO O SE " +  
"DESANOTE EL DOCUMENTO INSCRITO.";  
  
String descDevolutiva2 = "LOS ACTOS O NEGOCIOS JURIDICOS A QUE SE REFIERE  
EL ARTICULO 2 DEL " +  
"ESTATUTO DEL REGISTRO DE INSTRUMENTOS PUBLICOS, DEBERAN  
PRESENTARSE PARA SU INSCRIPCION, " +  
"DENTRO DE LOS DOS (2) MESES CALENDARIO, SIGUIENTES A LA FECHA DE  
SU OTORGAMIENTO PARA " +  
"ACTOS NOTARIALES O LA FECHA DE EJECUTORIA PARA PROVIDENCIAS  
JUDICIALES O ADMINISTRATIVAS; " +  
"VENCIDOS LOS CUALES, SE COBRARAN INTERESES MORATORIOS POR  
IMPUESTO DE REGISTRO, PREVISTOS " +  
"EN LA LEY 223 DE 1995 Y SU DECRETO REGLAMENTARIO 695 DE 1996  
ARTICULO 14. " ;  
  
String descDevolutiva3 = "EXCEPTUESE DE LO ANTERIOR, LOS CASOS  
RELACIONADOS CON EL NEGOCIO JURIDICO " +  
"DE HIPOTECA Y EL ACTO DE CONSTITUCION DE PATRIMONIO DE FAMILIA  
DE QUE TRATA EL ARTICULO 32 " +  
"DEL DECRETO LEY 1250 DE 1970, LOS CUALES SE DEBEN REGISTRAR  
DENTRO DE LOS NOVENTA (90) " +  
"DIAS HABILES SIGUIENTES A SU AUTORIZACION. VENCIDO EL TERMINO  
REGISTRAL ANTES SENALADO, " +  
"DEBERAN CONSTITUIRSE DE CONFORMIDAD CON EL PRECITADO  
ARTICULO. " ;  
  
String descDevolutiva4 = "CONTRA EL PRESENTE ACTO ADMINISTRATIVO,  
PROcede EL RECURSO DE REPOSICION " +  
"ANTE EL REGISTRADOR DE INSTRUMENTOS PUBLICOS Y EN SUBSIDIO, EL  
DE APELACION ANTE EL DIRECTOR " +  
"DE REGISTRO DE LA SUPERINTENDENCIA DE NOTARIADO Y REGISTRO  
DENTRO DE LOS CINCO (5) DIAS HABILES " +  
"SIGUIENTES A SU NOTIFICACION, EN VIRTUD DE LO PREVISTO POR EL  
NUMERAL 2 DEL ARTICULO 23 DEL DECRETO " +  
"412 DE 2007, PREVIO EL CUMPLIMIENTO DE LOS REQUISITOS  
ESTABLECIDOS EN LOS ARTICULOS 51 Y 52 DEL " +  
"DECRETO 01 DE 1984 (CODIGO CONTENCIOSO ADMINISTRATIVO). " ;
```

**Línea 203:**

```
/**  
 * @author Fernando Padilla Velez  
 * @change Mantis 5423: Acta - Requerimiento No 206 - FORMATO NOTA
```

```

    DEVOLUTIVA, se agrega otra parrafo.
    */
    this.imprimirLinea(ImprimibleConstantes.PEQUE,descDevolutiva4);
    this.imprimirLinea(ImprimibleConstantes.PEQUE,"");

protected void makeNewPage()

Línea 261:
/*
     * @author Fernando Padilla Velez
     * @change Mantis 5423: Acta - Requerimiento No 206 - FORMATO NOTA
DEVOLUTIVA,
     *      Se modifica para que la fecha de impresion este centrada.
*/
String fechaImp = this.fechaImpresion;
fechaImp = StringFormat.getCentrada(fechaImp,
ImprimibleConstantes.MAX_NUM_CHAR, 15);
this.imprimirLinea(ImprimibleConstantes.PLANO, fechaImp);
protected void imprimirFirmaNotaDevolutiva()

Línea 458:
/*
     * @author Fernando Padilla Velez
     * @change Mantis 5423: Acta - Requerimiento No 206 - FORMATO NOTA
DEVOLUTIVA.
     *      Se modifica el fin del documento.
*/
String finDocumento = "          FIN DE ESTE ACTO ADMINISTRATIVO";
protected void imprimirEntregado()

Línea 559:
/*
     * @author Fernando Padilla Velez
     * @change Mantis 5423: Acta - Requerimiento No 206 - FORMATO NOTA
DEVOLUTIVA.
     * Se modifica el parrafo de notificacion personal y se elimina la variable String
linea4 asi como su impresion.
*/
String linea1 = "CONFORME LO DISPUESTO EN EL ARTICULO 44 DEL CODIGO
CONTENCIOSO ADMINISTRATIVO, EN LA FECHA_____";
String linea2 = "SE NOTIFICO PERSONALMENTE EL PRESENTE ACTO
ADMINISTRATIVO A _____";
String linea3 = "QUIEN SE IDENTIFICÓ CON_____ No.
_____.";
```

## 207 Exclusion de circulos registrales

### REQUERIMIENTO

La SNR requiere implementar funcionalidad para permitir la exclusión de círculos registrales (Parametrizable) de la lista de ciudades visualizadas en la expedición de certificados entre ORIPS en el SIR. La idea es que se permita habilitar y/o deshabilitar círculos registrales en esta funcionalidad.

### DESARROLLO

Se agrego desde pantallas administrativas (Administración Lookup Types) un nuevo Lookup Code asociado con Lookup Type = CERTIFICADOS con el nombre CIRCULOS\_NO\_EXPIDEN\_CERTIFICADOS y en la cual irán en su valor los círculos separados por comas (,) que deseen ser excluidos del proceso de certificados nacionales como origen.

<b>CLASE o ARCHIVO JSP:</b>
<b>gov.sir.core.negocio.modelo.constantes.COPLookupTypes.java</b>
<b>Línea 107:</b>
<code>/*      * @author Fernando Padilla Velez      * @change 5453: Acta - Requerimiento No 207 - Exclusion de circulos registrales      *      Se crea nueva constante.</code>

```

/*
public static final String CERTIFICADOS = "CERTIFICADOS";
gov.sir.core.negocio.modelo.constantes.COPLookupCodes.java
Línea 148:
/**
 * @author Fernando Padilla Velez
 * @change 5453: Acta - Requerimiento No 207 - Exclusion de circulos registrales
 * Se crea nueva constante.
 */
public static final String CIRCULOS_NO_EXPIDEN_CERTIFICADOS =
"CIRCULOS_NO_EXPIDEN_CERTIFICADOS";
gov.sir.core.web.helpers.comun.MatriculaHelper.java
Línea 169:
/**
 * @author Fernando Padilla Velez
 * @change 5453: Acta - Requerimiento No 207 - Exclusion de circulos registrales
 * Se consulta el valor del lookupcode para el tipo CERTIFICADOS y codigo
CIRCULOS_NO_EXPIDEN_CERTIFICADOS.
*/
JDOLookupDAO lookDAO = new JDOLookupDAO();
String circulosNoNacionales = lookDAO.getValor(COPLookupTypes.CERTIFICADOS,
COPLookupCodes.CIRCULOS_NO_EXPIDEN_CERTIFICADOS);

if(circulosNoNacionales == null){
    circulosNoNacionales = "";
}

Línea 183:
/**
 * @author Fernando Padilla Velez
 * @change 5453: Acta - Requerimiento No 207 - Exclusion de circulos
registrales
* Se incluye la validacion para que no sean cargados en la lista de
circulos.
*/
if (!circuloAux.getIdCirculo().equals(circ.getIdCirculo()) &&
!(circulosNoNacionales.indexOf(circuloAux.getIdCirculo()) != -1)) {

```

## **DESARROLLO**

<b>DESPLIEGUE – 2.6.1</b> <b>09/07/2010</b>
<b>023 Cambios de nombre o razón Social Correcciones</b> <b>(Informativo)</b>
<p><b>REQUERIMIENTO</b></p> <p>Cambios de nombre o razón Social Correcciones. SIR-199</p> <p>Se presentan inconvenientes cuando las personas jurídicas (Bancos Sociedades) o las personas naturales, modifican sus nombres porque al calificar turnos de documento o ingresar correcciones por NIT o cédula que ya tenían registrado con la razón social o nombre anterior, trae los datos grabados previamente y no permite editarlos con el mismo número para las nuevas anotaciones.</p> <p>Ejemplos:</p>

1.-Corporación Colombiana de Ahorro y Vivienda DAVIVIENDA, hoy Banco Davivienda;

2.- Las señoras que figuraban con el apellido de casada y actualmente con los dos Apellidos de soltera.

Se requiere una solución que permita mostrar la información actualizada, cuando se presentan esas modificaciones al nombre, sin que se borren los antecedentes ya registrados correctamente.

#### **SEGUIMIENTO ADA:**

Es necesario realizar el proceso de validación de la modificación solicitada, esto se debe a que cuando se digita el documento de identificación del tercero, el sistema de manera automática carga la información restante, no permitiendo modificarla, por la afectación del tercero y toda la información relacionada en las anotaciones.

El proceso que se sugiere debería ser realizado por Pantallas Administrativas.

#### **OBSERVACIONES SNR:**

**ESTE REQUERIMIENTO SE DEBE TENER ENCUENTA PARA EL PROCESO DE CORRECCIONES Y REGISTRO E N REGISTRO :**Cuando se ingrese un ciudadano en el proceso de calificación con número de identificación que ya tiene información en estado definitiva en la base de datos, pero que ha tenido una modificación en cuanto al nombre, apellidos o razón social el sistema solo debe permitir agregar la nueva información únicamente en la anotación que se está creando, sin que modifique la información de los nombres,apellidos o razón social definitiva que se encuentre inscrita en otras anotaciones del mismo folio o en otros folios, el sistema debe mantener el historial de los registros anteriores a la nueva información agregada. Si con posterioridad en otro registro se ingresa el numero de documento de la persona debe mostrar la información digitada en el último cambio.

**EN CORRECCIONES:** Cuando se modifique en una anotación de un folio de matrícula el sistema no debe mostrar otros folios donde se encuentre inscrito ese número de documento de identificación y persona sino que debe mostrar únicamente las anotaciones del folio asociado al turno de corrección donde está inscrito el documento y persona y además permitir seleccionar en cuales de esas anotaciones desea hacer modificación, una vez seleccionada la anotación o anotaciones el sistema solo debe hacer la modificación de la información en estas.

#### **DESARROLLO**

En el requerimiento se solicita que cuando se realicen modificaciones en el nombre que identifica a las personas jurídicas o naturales estas sean reflejadas en la aplicación desde ese tiempo en adelante y no modifiquen el identificador del nombre en las anotaciones definitivas.

Esta solicitud involucra un rediseño de un alto porcentaje del modelo de Bases de Datos y Aplicación, que involucre información de ciudadanos asociadas a una o muchas anotaciones. Debido a que la arquitectura no fue construida considerando esta nueva

apreciación se solicita que este requerimiento sea tenido en cuenta en la arquitectura del nuevo sistema.

Basado en lo anterior se procede a cerrar el caso.

### **028 Bloqueo turno cuando se registra pago con consignacion (Informativo)**

#### **REQUERIMIENTO**

En liquidación Registro cuando se registra un pago con consignación y por caída del sistema no se genera turno de registro, pero si se genera turno de liquidación, en el rol Cajero cuando se valida el turno de liquidación y muestra el pantallazo de la forma de pago el sistema no permite avanzar el turno y además toma el valor de la consignación como agotado el valor de estado.

#### **DESARROLLO**

Luego de replicar el problema se encuentra que el sistema genera la liquidación aún si el sistema se cae, no obstante, esto no implica un error del aplicativo si no un descuido por parte del usuario de no anotar el numero de la liquidación, dado que si el usuario intenta ingresar y hacer el proceso nuevamente el sistema le informará que se ha agotado el valor de estado a consignar.

En ese orden de ideas, lo que el usuario debe hacer es comunicarse con Mesa de Ayuda e informar del inconveniente, dado que allí muy amablemente se le comunicara el número de liquidación para que el usuario pueda seguir el proceso.

### **049 Eliminar rol Administrador Regional**

#### **REQUERIMIENTO**

Pantallas administrativas- opción administración de perfiles

Con el fin de establecer medidas de control y seguridad en el manejo de asignación de roles a usuarios de las orips. Se requiere que la pantalla administrativa "administración de perfiles" se asigne al rol registrador y se suprima al rol administrador regional, será de manejo exclusivo del registrador.

#### **DESARROLLO**

Para este requerimiento es necesario ejecutar los scripts en la base de datos de producción.

Scripts

```
--Eliminación de la pantalla "Administración de Perfiles" del Rol  
'SIR_ROL_ADMINISTRADOR_REGIONAL'  
delete  
from  
SIR_OP_ROL_PANTALLA a  
where  
a.id_pantalla_administrativa = '203' and  
a.id_rol = 'SIR_ROL_ADMINISTRADOR_REGIONAL'
```

```
--Asignación de la pantalla "Administración de Perfiles" al Rol  
'SIR_ROL_REGISTRADOR'  
INSERT INTO  
SIR_OP_ROL_PANTALLA(id_rol,id_pantalla_administrativa)VALUES('SIR_ROL_REGISTRADOR',203);
```

## **066 Certificados Masivos (Informativo)**

### **REQUERIMIENTO**

(Solicitud realizada por Javier Rocha)

Los turnos de certificados masivos, crean turnos individuales de certificados, estos quedan en la fase entrega de certificados para que se puedan entregar individualmente pero al realizar este proceso, se entregan todos los turnos individuales pero el turno padre o turno de certificados masivos queda en la fase entrega certificados masivos como finalizado pero sigue visualizándose en el listado. Se requiere que los turnos creados por turnos de certificados masivos solo se puedan entregar con la entrega o finalización del turno de certificados masivos, y si la entrega de los turnos se hace por la pantalla administrativa entrega masiva de certificados se entregue el turno de certificados masivos que dio origen a estos turnos individuales.

### **DESARROLLO**

Se hizo la respectiva prueba con la asesoría de Javier Rocha, para lo cual se creó el turno 2010-999-20-3 y se le agregaron las matrículas 999-1819 y 999-2010, dichas matrículas se entregaron desde el rol Cajero Certificado Masivos lo que originó los turnos de entrega 2010-999-1-94 y 2010-999-1-95. Luego se verificó desde el rol Entrega Certificados que dichos turnos no apareciesen y se hizo lo propio desde Entrega Masiva de Certificados en Administración.

A este caso le dio solución la respuesta dada al Acta - Requerimiento No 065.

## **068 Constancia De No Propiedad (Informativo)**

### **REQUERIMIENTO**

En las Constancias de No propiedad o Titular de Derechos inscritos en el campo donde se refleja el nombre del ciudadano sobre el cual se expide la constancia, se visualiza que el nombre se sobreponen sobre la palabra señor (a) mostrando una información poco legible, como se observa en el fax enviado por la Orip de Santa Marta, que corresponde al turno 2009-060-2-7646 adjunto al oficio 1517 de julio 8 de 2009 suscrito por el Registrador de los cuales se les envía copia a la línea de fax de mesa de ayuda, para que haga parte de este requerimiento.

### **DESARROLLO**

Debido a que semanalmente se borran los imprimibles, se hicieron pruebas en el ambiente de producción actual y no se encontró el turno '2009-060-2-7646' que reporta este requerimiento. No obstante, se hicieron varias pruebas para emular el problema reportado y no se encontró dicho error. Por ser un caso sui generis (Caso Único) se decide cerrar el caso con la respectiva aprobación de Javier Rocha.

## **070 Grabacion Folios (Informativo)**

### **REQUERIMIENTO**

En el proceso de grabación directa de folios por pantalla administrativa en la Oficina del Círculo de SIMITI se grabó la matrícula 14566 y se encuentran en estado temporal. Cuando se retoma nuevamente la matrícula el sistema no permite avanzar a la segunda pantalla para terminar la grabación del folio. En la fase de calificación se efectuó el proceso de segregación y el sistema asignó el número matrícula mencionado y no la secuencial del círculo que lleva el sistema en ese momento.

## **DESARROLLO**

El caso 073\_141\_Error\_en\_la\_creación\_directa\_de\_folio solucionado en el acta 2.2.0 trabajo el problema en las fases temporales.

De igual manera se reviso el problema en los ambientes de pruebas y producción, y el funcionamiento corresponde al esperado del sistema.

## 074 Copias De Anotaciones De Cancelación

### **REQUERIMIENTO**

Se requiere efectuar el siguiente arreglo en copia de anotaciones en el proceso de calificación, toda vez que cuando se registra ó inscribe un documento que contiene una cancelación para ser inscrita en más de una matrícula inmobiliaria, cuando se va a efectuar la copia de la anotación en otros folios, el sistema despliega la pantalla pero no muestra los folios destino que están vinculados al turno, por lo tanto no se puede efectuar la copia de la cancelación y el proceso respectivo como es la selección de la anotación en cada uno de los folios asociados. Por tal motivo “Se requiere que se haga el ajuste , toda vez que siempre que se cancela una anotación debe aparecer los folios destino para hacer la copia. Se aclara que la inconsistencia se presente solo con las copias de anotaciones de cancelación. La funcionalidad esta bien cuando se trata de copiar anotaciones estándar.”

NOTA: Según correo enviado este mismo día por M.A.G se solicito verificar la funcionalidad también en el proceso de correcciones.

## **DESARROLLO**

Se hicieron dos modificaciones puntuales. La primera fue comentarrear un if else debido a que se ejecutan las mismas sentencias tanto en el if como en el else, se realiza este cambio para optimizar el proceso ya que se elimina una pregunta.

La segunda fue comentarrear un if dado que inhabilita presentar los Folios Destino cuando se crea una anotación canceladora y posteriormente se desea Copiar Anotación.

Se probó el funcionamiento del proceso bajo la asesoría de Carlos Javier Rocha, por tal razón se da por solucionado el requerimiento SIR "074\_141\_Copias De Anotaciones De Cancelación"

- Script Bases de Datos:
  - NINGUNA

- Creación De Clases:
  - NINGUNA

CLASE o ARCHIVO JSP:
C:\WORK\SIR-APP\web\WEB-INF\jspSeguro\registro\COPIA_ANOTACION_SELECCION_FOLIOS.jsp
Método:
<b>Código Línea 42:</b>
<pre> if(folioTemp.isDefinitivo()){     labelMatricula= folioTemp.getIdMatricula();     idMatricula= folioTemp.getIdMatricula(); } else{     //labelMatricula= folioTemp.getNombreLote();     labelMatricula= folioTemp.getIdMatricula();     idMatricula= folioTemp.getIdMatricula(); } </pre>

```

gov.sir.core.web.helpers.comun.ElementoLista elista = new
gov.sir.core.web.helpers.comun.ElementoLista();
if(folioTemp.getEstado() != null && folioTemp.getEstado().getIdEstado() != null
&& !folioTemp.getEstado().getIdEstado().equals(CEstadoFolio.CERRADO))
// t.getIdProceso() == Long.parseLong(CProceso.PROCESO_CORRECCIONES)) {

elista.setId(idMatricula);
elista.setValor(labelMatricula!=null?labelMatricula:"");

if(folio==null){
    idmatriculas.add(elista);
} else if(folio!=null && !folio.getIdMatricula().equals(elista.getId())){
    idmatriculas.add(elista);
}
}

Por:
/*
 * @author : Henry Gómez Rocha
 * @change : Se comentariaron que se ejecutan las
 * mismas sentencias en ambos lados, se realiza este cambio
 * para optimizar el proceso ya que se elimina una pregunta.
 * Caso Mantis : 0004480
*/
//if(folioTemp.isDefinitivo()){
    labelMatricula= folioTemp.getIdMatricula();
    idMatricula= folioTemp.getIdMatricula();
//}else{
    //labelMatricula= folioTemp.getNombreLote();
    //labelMatricula= folioTemp.getIdMatricula();
    //idMatricula= folioTemp.getIdMatricula();
//}

/*
 * @author : Henry Gómez Rocha
 * @change : Se comentarió el if dado que inhabilita presentar los Folios
 * Destino cuando se crea una anotación canceladora y
 * posteriormente se desea Copiar Anotación.
 * Caso Mantis : 0004480
*/
gov.sir.core.web.helpers.comun.ElementoLista elista = new
gov.sir.core.web.helpers.comun.ElementoLista();
/*if(folioTemp.getEstado() != null && folioTemp.getEstado().getIdEstado() != null
&& !folioTemp.getEstado().getIdEstado().equals(CEstadoFolio.CERRADO))
// t.getIdProceso() == Long.parseLong(CProceso.PROCESO_CORRECCIONES)){*/
elista.setId(idMatricula);
elista.setValor(labelMatricula!=null?labelMatricula:"");

if(folio==null){
    idmatriculas.add(elista);
} else if(folio!=null && !folio.getIdMatricula().equals(elista.getId())){
    idmatriculas.add(elista);
}
}
}

```

## 075 Cambio de relación del turno en fase (Informativo)

### **REQUERIMIENTO**

Cambio de relación del turno en fase. Turno 2009-100-6-16973. Informa Rodrigo Florez que el turno 2009-100-6-16973 Desanotado, en Mesa de control tenia la relación 2009-100-2994 pero en la fase Firma Registro paso con otra relación 2009-100-3007.

Se presenta el mismo caso en la Orip de Montería, las siguientes relaciones tienen los turnos en la fase Certificados Asociados pero en la fase Firma Registro se genero la relación 2009-140-1985 la cual se encuentra vacía. 2009-140-1979, 2009-140-1975, 1984,

1981, 1953. Favor verificar, se anexa relaciones.

Se abre caso en Mantis No 1678. Id Glpi 11012.

### **DESARROLLO**

Luego de tratar de replicar el caso (Acta - Requerimiento No 075 - Cambio de relación del turno en fase ) en el ambiente de producción generando los turnos 2010-999-6-180, 2010-999-6-181, 2010-999-6-182 y 2010-999-6-183 y generar la respectiva relación 2010-999-135 que incluye los turnos citados anteriormente, no se pudo reproducir de manera satisfactoria este caso.

Se ha dialogado con Carlos Javier Rocha (Jefe Mesa de Ayuda) y ha sugerido cerrar el caso debido a que hace más de tres meses este no se presenta. Cabe anotar, que en caso de que se vuelva a presentar este requerimiento el caso será reabierto para su respectiva revisión y análisis.

### **110 Liquidacion Baldio con Cuantia para Impuesto (Informativo)**

#### **REQUERIMIENTO**

Se debe corregir la funcionalidad de Radicación de documentos, en asociar actos y tarifas, dado que al liquidar el acto Adjudicación baldío (con Cuantía para impuesto), identificado con el código 41, esta liquidando de manera incorrecta tanto por concepto de derechos de registro como impuesto de registro.

Actualmente los liquida tomando el valor ingresado en el campo cuantía como cantidad de acto y los liquida como acto sin cuantía.

Lo correcto es que liquide así:

Derechos de registro, debe liquidar con tarifa media es decir el 2.5x 1000 sobre el valor incluido el campo cuantía.

El impuesto de Registro debe liquidar la mitad es decir el 0.5% (tarifa media) sobre el valor incluido en el campo cuantía.

Es de aclarar que hay que diferenciar del acto Adjudicación baldío (sin cuantía para impuesto) código 24 que la funcionalidad esta correcta.

### **DESARROLLO**

Luego de emular el problema descrito junto con Javier Rocha se ha encontrado que se ha corregido por pantallas administrativas del rol administrador. Por tal razón se da por cerrado el caso.

### **116 requerimiento anotaciones definitivas sin personas (Informativo)**

#### **REQUERIMIENTO**

De manera atenta y preocupante la SNR ha detectado que a fecha 31 de agosto de 2009, existen 2905 anotaciones validas ingresadas en SIR información no migrada, que no tienen sus correspondientes ciudadanos, lo que ocasiona cierto nivel de incertidumbre e inseguridad tanto en el aplicativo como en la integridad de la base de datos, toda vez que dentro del proceso de inscripción registral (ingreso de anotaciones en cualquiera de las funcionalidades), el aplicativo debe validar que todas las anotaciones que se graben deben tener personas asociadas, de lo contrario debe mostrar alerta restrictiva que impida la grabación de la anotación.

Se efectuó prueba en ambiente de producción con el usuario pruebas y el sistema presenta

inconsistencia respecto a que solo valida o muestra alerta restrictiva para anotaciones que se agregan ciertos códigos de naturaleza jurídica, esta validación no debe tener excepciones es para todas las anotaciones.

### **DESARROLLO**

los asuntos que implican el uso de anotaciones en los roles Auxiliar de Correcciones, Buscador Antiguo Sistema, Calificador y Pantallas Administrativas (Creación Directa de Folios) son creadas justamente después de ingresar un ciudadano. En caso de que el usuario olvide ingresar los datos del ciudadano, el sistema arrojará el siguiente mensaje “Debe ingresar por lo menos un ciudadano”, obligándolo por consiguiente a ingresar dichos datos. Este problema ya se encontraba solucionado por la respuesta dada al Acta - Requerimiento No 187- Anotaciones Sin Ciudadanos.

### **123 Error al asignar consignacion (Informativo)**

#### **REQUERIMIENTO**

Error al asignar consignación en el aplicativo. En le momento de asignar la consignación el sistema arroja el siguiente error. Consignación 88081486. Por favor verificar la consignación ya que la usuaria esta realizando bien el procedimiento.

El problema se presenta debido a que con esa consignación (88081486) se pagaron los turnos 2009-340-6-7886, 2009-340-1-30040 los cuales se encuentran anulados.

Para este caso ya se encontró la solución puntual, actualizando los turnos a no anulados mientras se agota el saldo de la consignación.

Este caso general se debe reportar a desarrollo para su revisión y solución. Id Ref Glpi 12490.

### **DESARROLLO**

Este requerimiento ha sido corregido con la solución dada al Caso Mantis 1876 y correspondiente al requerimiento SNR "072-141 Liberar consignaciones turnos anulados", para tal fin queda como constancia los turnos 2010-999-6-179 y 2010-999-6-180, en el primero de ellos se ingresó la consignación nro. 895623, posteriormente se anuló dicho turno y se usó la misma consignación en el turno 2010-999-6-180 no presentando el sistema ningún tipo de inconveniente con el uso de dicha consignación.

### **131 Error al ingresar nueva entrada (Informativo)**

#### **REQUERIMIENTO**

El sistema está generando error al realizar nueva entrada. Cuando se ingresa el turno en la fase de entregado devuelto, aparece No fue posible realizar la liquidación: Index 0, sixe 0". Este error se replicó en las Orips de Pasto y Valledupar, se trabajo el caso puntual, pero se requiere una solución general, para evitar su reproducción en el aplicativo. Se anexa los casos de Glpi como referencia.

Error al realizar nueva entrada. Turno 2009-240-6-13835. Informa Miguel Galvis que al ingresar el turno 2009-240-6-13835 Entregado Devuelto, como nueva entrada aparece: " No fue posible realizar la liquidación: index 0, sixe 0". Se anexa error. A solicitud de Javier Rocha Se escala este caso a desarrollo para su solución ya que el error de este caso es atípico de los inconvenientes de nuevas entradas y su solución no es estándar. Se requiere solución por parte de desarrollo para ingresar este turno como nueva entrada.

Pasto. Id Ref 12198.

Turno 2009-190-6-10532. Informa Tania Brito que al ingresar el turno 2009-190-6-10532 Entregado Devuelto, como nueva entrada aparece: "No fue posible realizar la liquidación: index: 0. Sixe 0". Valledupar.. Id Ref 13042.

### **DESARROLLO**

Caso Glpi No 12198. Debido a que no se obtuvo respuesta positiva por parte de la mesa de ayuda respecto al turno 2009-240-6-13835 al cual se le debía dar nueva entrada lo que en efecto se realizó generando el turno 2009-240-6-14444 de nueva entrada, por error se liquidó mal y por lo tanto el turno 14444 se debió anular -lo que en efecto se hizo-; al tratar de generar una nueva entrada con la liquidación correcta y el 14444 anulado, el sistema no lo permitió generando error que fue enviado por fax y mail. Hasta el momento no se obtuvo solución por parte de la mencionada mesa de ayuda y el usuario (BANCO DE COLOMBIA) al insistir tanto ya que se le vencían términos de la hipoteca se optó por levantar el anulado del 2009-240-6-14444 el cual ya se encuentra registrado. Por lo expuesto fue la única solución a este problema y no requerimos por el momento más ayuda a este impasse; pero es bueno tenerlo en cuenta para próximos inconvenientes de esta clase. Miguel Galvis. Registro Pasto.

Caso resuelto con el cambio realizado en el código por el caso en el acta de requerimiento No 195 (Incluido en este despliegue 002\_6\_0).

## **134 Error Imprimir Folio**

### **REQUERIMIENTO**

Error al reimprimir. Turno 2009-368-2-1188. Informa Susana Rivera que al tratar de reimprimir por pantalla administrativa: Reimpresión de Consultas el turno 2009-368-2- 1188 Entregado, aparece: "Error al imprimir el folio: Index:0, Size:0" Purificación. Id Glpi 2803.

### **DESARROLLO**

Se estaba presentando un error al tratar de reimprimir una consulta, ya que, se intentaba acceder a una lista vacía.

se modifico la línea que intentaba acceder a la lista vacía y se construyo el objeto Ciudadano de la siguiente forma:

Clase: C:\WORK\SIR-  
APP\src\gov\sir\core\negocio\acciones\administracion\ANIImprimirFolio.java

Método: private EvnRespImprimirFolio reImprimirConsulta(EvnImprimirFolio evento)  
throws EventoException

Línea 163:

```
/**  
 * @author: Fernando Padilla Velez  
 * @change: Caso mantis 2803: Acta - Requerimiento No 134 - Error al reimprimir,  
 * Se comenta la linea 1678, ya que, la variable busqueda ya no será usada;  
 * Se crea el ciudadano con los valores que posee solConsulta.getCiudadano().  
 */  
//Busqueda busqueda = (Busqueda)solConsulta.getBusquedas().get(0);
```

```
Ciudadano ciudadano = new Ciudadano();
ciudadano.setTipoDoc(solConsulta.getCiudadano().getTipoDoc());
ciudadano.setDocumento(solConsulta.getCiudadano().getDocumento());
ciudadano.setNombre(solConsulta.getCiudadano().getNombre());
ciudadano.setApellido1(solConsulta.getCiudadano().getApellido1());
ciudadano.setApellido2(solConsulta.getCiudadano().getApellido2());
```

### 139 Error al Avanzar el Turno (Informativo)

#### **REQUERIMIENTO**

Error al avanzar turno. El Turno 2009-004-3-269. Al tratar de avanzar el turno de la fase de revisar y aprobar el sistema genera error. Al tratar de avanzar el turno de la fase de revisar y aprobar el sistema genera error. Fallo en el servicio para hacer definitivos los datos temporales, se anexa imagen. Andes. Id Glpi 13331.

#### **DESARROLLO**

Luego de un análisis del proceso, se encontró que regresar el turno desde la fase revisar y aprobar al role auxiliar de correcciones y continuar con el flujo del proceso, este no presentó ningún inconveniente en la fase de revisar y aprobar y se entregó satisfactoriamente.

### 153 Radicacion - Registro

#### **REQUERIMIENTO**

Para que al momento de seleccionar Oficina de Procedencia Internacional se despliegue únicamente el Campo Oficina Ubicación Internacional y su respectivo campo de texto, es decir que los campos de Departamento, Municipio, Tipo de Oficina, Número de Oficina no los debe desplegar, para evitar confusión al usuario.

#### **DESARROLLO**

Al momento de seleccionar Oficina de Procedencia Internacional se estaban desplegando junto con el Campo Oficina Ubicación Internacional y su respectivo campo de texto los campos de Departamento, Municipio, Tipo de Oficina y Número de Oficina que corresponden a Oficina de Procedencia Nacional.

Solución:

En el archivo “**HELPER\_OFICINAS\_NO\_POPUP.jsp**” existe una función denominada **showhide()**, la cual solamente estaba ocultando información parcial al momento de seleccionar Oficina de Procedencia Internacional, por tal motivo, se hicieron los ajustes correspondientes para que al momento de seleccionar Oficina de Procedencia Internacional toda la información correspondiente a Oficina de Procedencia Nacional sea ocultada.

- Script Bases de Datos:
  - NINGUNA
- Creación De Clases:
  - NINGUNA

Se modificaron los siguientes archivos:

**CLASE o ARCHIVO JSP:**

**C:\WORK\SIR-APP\web\WEB-INF\jspSeguro\registro\HELPER\_OFICINAS\_NO\_POPUP.jsp**

	<b>Método:</b> <b>showhide</b>	
	<pre>/**  * @author: Henry Gómez Rocha  * @change: Se valida que al momento de seleccionar Oficina de  *          Procedencia Internacional se despliegue únicamente el Campo Oficina  *          Ubicación Internacional.  */</pre>	
<b>Código Línea 84:</b> <pre>function showhide(){     var valor;      valor = document.getElementById("uno").checked;     if(valor){         document.getElementById("Nacional1").style.display = "block";         document.getElementById("Nacional2").style.display = "block";         document.getElementById("Internacional").style.display = "none";     }else{         document.getElementById("Nacional1").style.display = "none";         document.getElementById("Nacional2").style.display = "none";         document.getElementById("Internacional").style.display = "block";     } }</pre>		
<b>Por:</b> <pre>function showhide(){     var valor;      valor = document.getElementById("uno").checked;     if(valor){         document.getElementById("Nacional1").style.display = "block";         document.getElementById("Nacional2").style.display = "block";         document.getElementById("departamentoMunicipio").style.display = "block";         document.getElementById("oficinaOrigen").style.display = "block";         document.getElementById("Internacional").style.display = "none";     }else{         document.getElementById("Nacional1").style.display = "none";         document.getElementById("Nacional2").style.display = "none";         document.getElementById("departamentoMunicipio").style.display = "none";         document.getElementById("oficinaOrigen").style.display = "none";         document.getElementById("Internacional").style.display = "block";     } }</pre>		

## 182 Error en turno de devolucion

### **REQUERIMIENTO**

Turno: 2010-060-4-14

#### Motivo Inconsistencia:

el turno de devolución 2010-060-4-14 corresponde al turno masivo 2010-060-20-1144 que no se encuentra finalizado aún, pero el sistema validó el turno de devolución y en este momento se encuentra en la fase de resolución pero al tratar de avanzarlo el sistema genera error: ha ocurrido un error con descripción no disponible para ver más detalles haga clic en ver la excepción. Grupo Cartagena

### **DESARROLLO**

Se modificaron los siguientes archivos:

<b>CLASE:</b> <b>gov.sir.core.web.acciones.comun.ATurno.java</b>
<b>Método:</b>
<i>public void doEnd(HttpServletRequest request, EventoRespuesta evento)</i>
<b>Código Insertado Línea 921:</b> <i>/*JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno Certificado Masivos*/</i> <i>if(local_SolicitudAnterior instanceof SolicitudCertificadoMasivo){</i>

```

    LiquidacionTurnoCertificadoMasivo liqCon =
(LiquidacionTurnoCertificadoMasivo)local_SolicitudAnterior.getLiquidaciones().get(0);
    ValorTotalDevuelto += liqCon.getValor()>0?liqCon.getValor():0;
    if(liqCon.getPago()!=null && liqCon.getPago().getAplicacionPagos()!=null){
        List aPagos = liqCon.getPago().getAplicacionPagos();
        for(int j=0;j<aPagos.size();j++){
            if(((AplicacionPago)aPagos.get(j)).getDocumentoPago() instanceof DocPagoCheque){
                DocPagoCheque chPago = (DocPagoCheque)((AplicacionPago)aPagos.get(j)).getDocumentoPago();
                if(chPago.getSaldoDocumento()>0){
                    saldoFavor+=chPago.getSaldoDocumento();
                }
            }
            if(((AplicacionPago)aPagos.get(j)).getDocumentoPago() instanceof DocPagoConsignacion){
                DocPagoConsignacion chPago =
(DocPagoConsignacion)((AplicacionPago)aPagos.get(j)).getDocumentoPago();
                if(chPago.getSaldoDocumento()>0)
                    saldoFavor+=chPago.getSaldoDocumento();
            }
        }
    }
}

```

#### CLASE:

gov.sir.core.web.acciones.devolucion.AWDevolucion.java

#### Método:

**public Evento do\_PageProcessingProcess\_Resolucion\_OnAceptar(HttpServletRequest request)  
throws AccionWebException**

#### Código Insertado Línea 1415:

```

/*JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno Certificado Masivos*/
if(local_SolicitudAnterior instanceof SolicitudCertificadoMasivo){
    LiquidacionTurnoCertificadoMasivo liqFoto =
(LiquidacionTurnoCertificadoMasivo)local_SolicitudAnterior.getLiquidaciones().get(0);
    ValorTotalDevuelto += liqFoto.getValor()>0?liqFoto.getValor():0;
    if(liqFoto.getPago()!=null && liqFoto.getPago().getAplicacionPagos()!=null){
        List aPagos = liqFoto.getPago().getAplicacionPagos();
        for(int j=0;j<aPagos.size();j++){
            if(((AplicacionPago)aPagos.get(j)).getDocumentoPago() instanceof DocPagoCheque){
                DocPagoCheque chPago = (DocPagoCheque)((AplicacionPago)aPagos.get(j)).getDocumentoPago();
                if(chPago.getSaldoDocumento()>0){
                    saldoFavor+=chPago.getSaldoDocumento();
                }
            }
            if(((AplicacionPago)aPagos.get(j)).getDocumentoPago() instanceof DocPagoConsignacion){
                DocPagoConsignacion chPago =
(DocPagoConsignacion)((AplicacionPago)aPagos.get(j)).getDocumentoPago();
                if(chPago.getSaldoDocumento()>0){
                    saldoFavor+=chPago.getSaldoDocumento();
                }
            }
        }
    }
}

```

#### Código Línea 1728:

validateCondition = local\_SolicitudAnterior instanceof SolicitudCertificado;

#### POR 1751

```

/*JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno Certificado Masivos*/
validateCondition = (local_SolicitudAnterior instanceof SolicitudCertificado)?true:(local_SolicitudAnterior instanceof SolicitudCertificadoMasivo)?true:false;

```

#### Código Línea 1904:

```

local_Liquidacion = build_LocalLiquidacion(local_SolicitudAnterior, pageModel_ValorDerechos,
pageModel_ValorImpuestos, pageModel_ValorCertificados, pageModel_ValorTotal, texto);

```

#### POR 1928

```

/*JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno Certificado Masivos*/
local_Liquidacion = build_LocalLiquidacion(local_SolicitudAnterior, pageModel_ValorDerechos,
pageModel_ValorImpuestos, pageModel_ValorCertificados, pageModel_ValorTotal, pageModel_ValorSaldoFavor, texto);

```

#### Código Línea 1917:

```
Liquidacion local_Liquidacion2 = build_LocalLiquidacion(local_SolicitudAnterior, pageModel_ValorDerechosMayorValor,  
pageModel_ValorImpuestosMayorValor, pageModel_ValorCertificadosMayorValor, 0, texto);
```

#### POR 1942

```
/*JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno Certificado Masivos*/  
Liquidacion local_Liquidacion2 = build_LocalLiquidacion(local_SolicitudAnterior,  
pageModel_ValorDerechosMayorValor, pageModel_ValorImpuestosMayorValor, pageModel_ValorCertificadosMayorValor, 0,  
pageModel_ValorSaldoFavor ,texto);
```

#### Código Linea 1936:

```
private Liquidacion build_LocalLiquidacion(Solicitud solicitud, double valorDerechos, double valorImpuestos, double  
valorCertificados, double valorTotal, String justificacionTx) {
```

#### POR 1962

```
/*JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno Certificado Masivos*/  
private Liquidacion build_LocalLiquidacion(Solicitud solicitud, double valorDerechos, double valorImpuestos, double  
valorCertificados, double valorTotal, double valorSaldoFavor, String justificacionTx) {
```

#### Método:

```
public Evento do_PageProcessingProcess_Resolucion_OnAceptar(HttpServletRequest request)  
throws AccionWebException
```

#### Código Linea 1936:

```
private Liquidacion build_LocalLiquidacion(Solicitud solicitud, double valorDerechos, double valorImpuestos, double  
valorCertificados, double valorTotal, String justificacionTx) {
```

#### POR 1962

```
/*JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno Certificado Masivos*/  
private Liquidacion build_LocalLiquidacion(Solicitud solicitud, double valorDerechos, double valorImpuestos, double  
valorCertificados, double valorTotal, double valorSaldoFavor, String justificacionTx) {
```

#### Código Insertado Linea 2008:

```
/*JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno Certificado Masivos*/  
else if(solicitud instanceof SolicitudCertificadoMasivo){  
LiquidacionTurnoCertificadoMasivo local_Liquid = new LiquidacionTurnoCertificadoMasivo();  
valorSaldoFavor *= (-1);  
local_Liquid.setValor(valorSaldoFavor);  
result = local_Liquid;  
return result;  
}
```

#### CLASE:

```
gov.sir.core.negocio.acciones.devolucion.ANDevolucion.java
```

#### Método:

```
private List validarTurnoAnt(EvnDevolucion evento) throws EventoException {
```

#### Código Insertado Linea 957:

```
/*JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno Certificado Masivos*/  
//CASO TURNO DE CERTIFICADOS MASIVOS  
if(tAnt.getSolicitud() instanceof SolicitudCertificadoMasivo){  
    if(tAnt.getSolicitud().getLiquidaciones()==null || tAnt.getSolicitud().getLiquidaciones().size()==0 &&  
    !evento.getTipoEvento().equals(EvnDevolucion.AGREGAR_TURNO_ANT)){  
        throw new ValidacionParametrosException("El turno asociado no es válido.  
No existen liquidaciones ni pagos registrados para este turno");  
    }  
    List aPagos =  
(Liquidacion)tAnt.getSolicitud().getLiquidaciones().get(0).getPago().getAplicacionPagos();  
Iterator iter = aPagos.iterator();  
int efectivo = 0;  
int sinSaldoAFavor = 0;  
while(iter.hasNext()){  
    DocumentoPago docPago = ((AplicacionPago)iter.next()).getDocumentoPago();  
    if(docPago instanceof DocPagoCheque){  
        if(((DocPagoCheque)docPago).getSaldoDocumento()<=0){  
            sinSaldoAFavor++;  
        }  
    }else if(docPago instanceof DocPagoConsignacion){  
        if(((DocPagoConsignacion)docPago).getSaldoDocumento()<=0){  
            sinSaldoAFavor++;  
        }  
    }else{  
        efectivo++;  
    }  
}  
if(efectivo > 0 && !evento.getTipoEvento().equals(EvnDevolucion.AGREGAR_TURNO_ANT)){
```

```

        throw new ValidacionParametrosException("El turno asociado no es válido. Solo se permiten devolver turnos de
certificados masivos con pagos que no sean en efectivo");
    }
    if(sinSaldoAFavor > 0 && !evento.getTipoEvento().equals(EvnDevolucion.AGREGAR_TURNO_ANT)){
        throw new ValidacionParametrosException("El turno asociado no es válido. Solo se permiten devolver turnos de
certificados masivos con pagos que poseen saldo a favor");
    }
}

```

**Método:**

```

public EventoRespuesta procesar_ResolucionAceptar(EvnDevolucion evento) throws
EventoException {

```

**Código Insertado Linea 1169:**

```

/*JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno Certificado Masivos*/
else if (valorDevolver instanceof LiquidacionTurnoCertificadoMasivo){
    liquidacionDevolucion.setValor(evento.getValorSaldoFavor());
}

```

**ARCHIVO JSP:**

**WEB\_INF.jspSeguro.devolucion.ANALISIS\_DEVOLUCION.jsp**

**Código Insertado Linea 1237:**

```

<%-- JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno
Certificado Masivos
--%>
<%
// Solo cuando es de Certificados
displayCondition = local_SolicitudAnterior instanceof SolicitudCertificadoMasivo;
if( displayCondition ) {
// -----
%>


|                                                                                                     |             |    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |  |
|-----------------------------------------------------------------------------------------------------|-------------|----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|
|  | Valor total | \$ | <% try { textHelper.setId( AWDevolucion.PAGEITEM_LIQUIDVALORTOTAL ); textHelper.setNombre( AWDevolucion.PAGEITEM_LIQUIDVALORTOTAL ); textHelper.setFuncion("onBlur=\"reescibirValor(this.value,this.id)\""); textHelper.setCssClase("campositem"); textHelper.setReadonly(true); textHelper.render(request,out); } catch(HelperException re){ out.println("ERROR " + re.getMessage()); } %> </td> <td width="1%" style="text-align: right;">  |  |
|-----------------------------------------------------------------------------------------------------|-------------|----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|



|                                                                                                     |               |    |                                                                                                                                                                                                                                                                                                                                                                                             |
|-----------------------------------------------------------------------------------------------------|---------------|----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | Saldo a Favor | \$ | <% try { textHelper.setId( AWDevolucion.PAGEITEM_LIQUIDSALDOFAVOR ); textHelper.setNombre( AWDevolucion.PAGEITEM_LIQUIDSALDOFAVOR ); textHelper.setFuncion("onBlur=\"reescibirValor(this.value,this.id)\""); textHelper.setCssClase("campositem"); textHelper.setReadonly(true); textHelper.render(request,out); } catch(HelperException re){ out.println("ERROR " + re.getMessage()); } %> |
|-----------------------------------------------------------------------------------------------------|---------------|----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|


```

```

        }
    %>
    </td>
    <td width="1%"></td>
    <td> &nbsp; </td>
    </tr>
</table>
<%
} // :if
%>

```

**WEB\_INF.jspSeguro.devolucion.RECURSOS\_DEVOLUCION.jsp**

**Código Insertado Linea 1541:**

```

<%-- JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno
   Certificado Masivos
--%>
<%
// Solo cuando es de Certificados
displayCondition = local_SolicitudAnterior instanceof SolicitudCertificadoMasivo;
if( displayCondition ) {
// -----
%>
    <table width="100%" class="camposform">
        <tr>
            <td width="3%"></td>
            <td width="15%">Valor total</td>
            <td width="1%">$</td>
            <td width="15%">
                <%
                    try {
                        textHelper.setId( AWDevolucion.PAGEITEM_LIQUIDVALORTOTAL );
                        textHelper.setNombre( AWDevolucion.PAGEITEM_LIQUIDVALORTOTAL );
                        textHelper.setFuncion("onBlur=\"reescribirValor(this.value,this.id)\"");
                        if (tipoChecked == null || tipoChecked.equals(AWDevolucion.CHECKED_SI)){
                            textHelper.setCssClase("camposformtext");
                        }else{
                            textHelper.setCssClase("campositem");
                            textHelper.setReadonly(true);
                        }
                    }
                    textHelper.render(request,out);
                }
                catch(HelperException re){
                    out.println("ERROR " + re.getMessage());
                }
            %>
            </td>
            <td width="1%"></td>
            <td> &nbsp; </td>
        </tr>
    </table>
    <table width="100%" class="camposform">
        <tr>
            <td width="3%"></td>
            <td width="15%">Saldo a Favor</td>
            <td width="1%">$</td>
            <td width="15%">
                <%
                    try {
                        textHelper.setId( AWDevolucion.PAGEITEM_LIQUIDSALDOFAVOR );
                        textHelper.setNombre( AWDevolucion.PAGEITEM_LIQUIDSALDOFAVOR );

                        textHelper.setFuncion("onBlur=\"\\reescribirValor(this.value,this.id)\"");
                        if (tipoChecked == null || tipoChecked.equals(AWDevolucion.CHECKED_SI)){
                            textHelper.setCssClase("camposformtext");
                        }else{
                            textHelper.setCssClase("campositem");
                            textHelper.setReadonly(true);
                        }
                    }
                    textHelper.render(request,out);
                }
                catch(HelperException re){

```

```

        out.println("ERROR " + re.getMessage());
    }
%>
</td>
<td width="1%"></td>
<td &nbsp;</td>
</tr>
</table>
<%
} // :if
%>

```

## WEB\_INF.jspSeguro.devolucion.GENERAR\_RESOLUCION.jsp

### Código Insertado Linea 1185:

```

<%-- JAlcazar Caso Mantis 3905 Creacion Proceso Devolucion Turno
Certificado Masivos
--%>
<%
// Solo cuando es de Certificados
displayCondition = local_SolicitudAnterior instanceof SolicitudCertificadoMasivo;
if( displayCondition ) {
-----
%>
<table width="100%" class="camposform">
<tr>
<td width="3%"></td>
<td width="15%">Valor total</td>
<td width="1%">$</td>
<td width="15%">
<%
try {
textHelper.setId( AWDevolucion.PAGEITEM_LIQUIDVALORTOTAL );
textHelper.setNombre( AWDevolucion.PAGEITEM_LIQUIDVALORTOTAL );
textHelper.setFuncion("onBlur='reescribirValor(this.value,this.id)'''");
textHelper.setCssClase("campositem");
textHelper.setReadonly(true);
textHelper.render(request,out);
}
catch(HelperException re){
out.println("ERROR " + re.getMessage());
}
%>
</td>
<td width="1%"></td>
<td &nbsp;</td>
</tr>
</table>
<table width="100%" class="camposform">
<tr>
<td width="3%"></td>
<td width="15%">Saldo a Favor</td>
<td width="1%">$</td>
<td width="15%">
<%
try {
TextHelper textHelperAux = new TextHelper();
textHelperAux.setId( AWDevolucion.PAGEITEM_LIQUIDSALDOFAVOR );
textHelperAux.setNombre( AWDevolucion.PAGEITEM_LIQUIDSALDOFAVOR );

textHelperAux.setFuncion("onBlur='reescribirValor(this.value,this.id)'''");
textHelperAux.setCssClase("camposformtext");

textHelperAux.render(request,out);
}
catch(HelperException re){
out.println("ERROR " + re.getMessage());
}
%>
</td>
<td width="1%"></td>

```

```

        <td> &nbsp; </td>
        </tr>
    </table>
<%>
} // :if

```

## 187 Anotaciones Sin Ciudadanos

### **REQUERIMIENTO**

Haciendo pruebas, el sistema valida que se ingresen ciudadanos en las anotaciones, a excepción de las cancelaciones que se dejan crear sin ciudadanos. Se debería consultar si es un error del sistema o una regla de negocio.

Si estas anotaciones canceladoras deben llevar ciudadanos obligatoriamente se debe implementar la validación que se maneja para las otras anotaciones, para las que ya se encuentran registradas sin ciudadanos se debe notificar a la SNR para que realicen las correcciones ya que también es un error por parte del CALIFICADOR.

Se requiere realizar la validación por la interface para garantizar que todas las anotaciones tengan ciudadanos.

### **DESARROLLO**

- Script Bases de Datos:  
NINGUNA.
- Creación De Clases:  
NINGUNA (el mismo formato de modificación clases o jsp)
- Se modificaron los siguientes archivos:

CLASE
<b>SIR-APP\src\gov\sir\core\web\acciones\registro\AWCalificacion.java</b>
<b>Método:</b>
<b>private Evento eliminarCiudadano(HttpServletRequest request) throws AccionWebException</b>

Se adicionan las siguientes líneas al método.

```

/** @author: Guillermo Cabrera.
 * @change: se remueve de la sesión los ciudadanos ya que al ingresar los ciudadanos y posteriormente
 * eliminarlos, estos quedan en sesión y permite crear anotaciones sin ciudadanos asociados.
 * Mantis: 3397 Acta - Requerimiento No 187*/
//request.getSession().setAttribute(WebKeys.LISTA_CIUDADANOS_ANOTACION, ciudadanos);

request.getSession().removeAttribute(WebKeys.LISTA_CIUDADANOS_ANOTACION);

```

CLASE
<b>SIR-APP\src\gov\sir\core\web\acciones\registro\AWCalificacion.java</b>
<b>Método:</b>
<b>private Evento cancelarAnotacion(HttpServletRequest request) throws AccionWebException</b>

Se adicionan las siguientes líneas al método.

```

/*** @author: Guillermo Cabrera.
 * @change: Se realiza la validación para que sea obligatorio el ingreso de
 ciudadanos
 * cuando se va a crear una anotación canceladora.
 * Mantis: 3397 Acta - Requerimiento No 187*/
else
{
    exception.addError("Debe ingresar por lo menos un ciudadano");
    if (exception != null && exception instanceof ValidacionParametrosException &&
exception.getErrores().size() > 0)
    {
        throw exception;
    }
}

```

## 90 Reporte E01\_Estadística por Indicadores de Registro

### **REQUERIMIENTO**

Reporte E01: Estadística por Indicadores de Registro.

Al momento de generar el Reporte E01: Estadística por Indicadores de Registro el sistema notifica error.

EL usuario Josa Luis García informa que al momento de generar el Reporte E01: Estadística por Indicadores de Registro el sistema notifica error. Se adjunta imagen de error. Grupo Bucaramanga

### **DESARROLLO**

Se modifica procedimiento SP\_REPORTE\_E01, el cual se ejecutara en el próximo despliegue. Se adjunta actualización del procedimiento y plan Marcha Atrás del procedimiento SP\_REPORTE\_E01.

## 194 Reporte de consultas

### **REQUERIMIENTO**

*Gipi No 16728*

Motivo Inconsistencia:

El usuario José Luis García nos informa que al generar el reporte de INFORME RECAUDOS DE CONSULTAS DEL DIA POR USUARIO de fecha 23-03-2010 el sistema arroja un subtotal de 119.900, total consultas anuladas 5.500 y total en efectivo de 103.400. Grupo Bucaramanga

### **DESARROLLO**

Se modifican las expresiones para calcular el V\_RECAUDO\_EFECTIVO\_TOTAL, V\_RECAUDO\_CONSIG\_TOTAL, V\_RECAUDO\_CHEQUE\_TOTAL ya que se están calculando erradamente cuando hay turnos anulados.

1. Expresión Calcular V\_RECAUDO\_EFECTIVO\_TOTAL

*Antes:*

```

($F{EFECTIVO} != null ? $F{EFECTIVO} : new BigDecimal(0.0)).subtract(
    ($V{V_ANUL_EFECTIVO_TOTAL} != null ? $V{V_ANUL_EFECTIVO_TOTAL} : new BigDecimal(0.0))
)

```

*Código Modificado:*

```

($F{EFECTIVO} != null ? $F{EFECTIVO} : new
BigDecimal(0.0)).subtract($F{ANULADO}.equals("S") ? $F{EFECTIVO} : new

```

<p>BigDecimal(0.0))</p> <p>2. Expresión Calcular V_RECAUDO_CONSIG_TOTAL</p> <p><i>Antes:</i></p> <pre>(\$V{V_CONSIG_TOTAL} != null ? \$V{V_CONSIG_TOTAL} : new BigDecimal(0.0)).subtract( (\$V{V_ANUL_CONSIG_TOTAL} != null ? \$V{V_ANUL_CONSIG_TOTAL} : new BigDecimal(0.0)) )</pre> <p><i>Código Modificado:</i></p> <pre>(\$F{CONSIGNACION} != null ? \$F{CONSIGNACION} : new BigDecimal(0.0)).subtract(\$F{ANULADO}.equals("S") ? \$F{CONSIGNACION} : new BigDecimal(0.0))</pre> <p>3. Expresión Calcular V_RECAUDO_CHEQUE_TOTAL</p> <p><i>Antes:</i></p> <pre>(\$V{V_CHEQUE_TOTAL} != null ? \$V{V_CHEQUE_TOTAL} : new BigDecimal(0.0)).subtract( (\$V{V_ANUL_CHEQUE_TOTAL} != null ? \$V{V_ANUL_CHEQUE_TOTAL} : new BigDecimal(0.0)) )</pre> <p><i>Código Modificado:</i></p> <pre>(\$F{CHEQUE} != null ? \$F{CHEQUE} : new BigDecimal(0.0)).subtract(\$F{ANULADO}.equals("S") ? \$F{CHEQUE} : new BigDecimal(0.0))</pre>	<p><b>195 La nueva entrada no asocia el certificado</b></p> <p><b>REQUERIMIENTO</b></p> <p><b>Gipi No 16710</b></p> <p>Turno: 2010-442-6-785</p> <p>Motivo Inconsistencia: Se comunica Isabel Martínez e informa que al ingresar el turno 2010-442-6-785 como nueva entrada el sistema no le está asociando el turno de certificado.</p> <p>En la oficina van a ingresar el turno 2010-442-6-785 como nueva entrada, al realizar el proceso el sistema no le asocia el certificado.</p> <p>Por favor tener en cuenta que en la oficina la funcionaria anulo dos turnos al tratarlos de ingresar como nueva entrada, el 2010-442-6-1004 y el 2010-442-6-1005. Grupo Puerto Así</p> <p><b>DESARROLLO</b></p> <p>Al realizar nuevas entradas de registros con certificados asociados y teniendo en cuenta que ya se había utilizado anteriormente pero cancelando esos turnos, se observa que al tomar nuevamente el turno como nueva entrada no asocia el certificado.</p> <p>Solución: Este problema se da ya que al realizar una validación de si existe un turno de devolución para este certificado asociado toma todos los turnos siguientes a él, teniendo</p>
---	--

en cuenta los turnos anulados. se modifica la consulta de turnos siguientes de devolución para que no tenga en cuenta los turnos anulados.

clase: gov\sir\hermod\dao\impl\jdogenie\JDOTurnosDAO.java

método: protected TurnoEnhanced getTurnoSiguienteDevolucion(String idWork, String idCirculo,PersistenceManager pm) throws DAOException

linea: 5839

/\*\*

\* @author: Fernando Padilla Velez

\* @change: Caso Mantis 4275: error al ingresar nueva entrada, Se modifica la consulta para que no obtenga los turnos anulados.

\*\*/

```
query.setFilter("this.solicitud.turnoAnterior.idWorkflow==idWork &&\n" +\n"this.solicitud.turnoAnterior.idCirculo==idCir && this.anulado == 'N'");
```

## DESPLIEGUE – 2.6.0

16/06/2010

### Impuesto Puerto Lopez (Informativo)

#### **REQUERIMIENTO**

Motivo Inconsistencia: Se comunica la sra Margarita Ortiz informando que en el momento de generar la liquidación del documento no esta cobrando el impuesto.

Grupo Puerto Lopez

#### **DESARROLLO**

Se suben cambios del ear en producción de las clases

- \* C:\WORK\SIR-APP\src\gov\sir\core\web\helpers\registro\AgregarActosHelper.class
- \* C:\WORK\SIR-APP\src\gov\sir\hermod\HermodProperties.class

Se actualiza el archivo de mapas.jar

C:\WORK\SIR-APP\src\properties\produccion\gov.sir.hermod.properties

Se realiza prueba en ambiente de producción con Javier Rocha

USUARIO: LUZ.DOMINGUEZ

CIRCULO: PUERTO LOPEZ (234)

Y se observa que en el proceso de registro ya se despliegan los tipos de impuesto

### 185 IntegracionCodigoBotonPago

#### **REQUERIMIENTO**

Integración de las piezas de código requeridas para la comunicación entre el aplicativo Botón de Pago y SIR.

#### **DESARROLLO**

1. Creación del Círculo PORTAL en SIR

Para consistencia en reportes y una mejor identificación para los certificados entre ORIP es

necesario crear un Círculo para la generación de turnos del módulo de pago. Se deben seguir los siguientes pasos:

- Ingresar al SIR como Administrador Nacional
- En pantallas administrativas, en "Administración de Círculos Registrales", crear el circulo PORTAL con id 000
- Para el nuevo círculo, crear el usuario "PORTAL.SNR"
- En base de datos, ejecutar los siguientes scripts:

--habilitar el circulo para el particionamiento

```
alter table SIR_OP_TURNO modify partition TRNO_PART_300 add values('000');
alter table SIR_OP_TURNO_HISTORIA modify partition TRHS_PART_300 add values('000');
alter table SIR_NE_CIUDADANO modify subpartition CDDN_PART_0_300 add values('000');
alter table SIR_NE_CIUDADANO modify subpartition CDDN_PART_1_300 add values('000');
```

--crear el nivel para ese circulo

```
insert into AURIGA.CMN_NIVEL
values('NPTL','N0','PORTAL', '000',null,null,null,null);
```

--agregar la estacion a ese nivel

```
insert into AURIGA.CMN_REL_ESTACION_NIVEL
values ('X-PORTAL.SNR', NPTL);
```

--Para agregar las estaciones compartidas

```
insert into AURIGA.CMN_ESTACION
values('X-ENTREGA.CER.000', 'Estacion ENTREGA.CER 000', 'Estacion ENTREGA.CER 000')
```

--agregar la estacion compartida a ese nivel

```
insert into AURIGA.CMN_REL_ESTACION_NIVEL
values ('X-ENTREGA.CER.000', 'NPTL');
```

- Otra vez en el sistema SIR, ir a pantalla administrativa "Administración de Perfiles" y asignar al usuario PORTAL.SNR las estaciones X- PORTAL.SNR para el rol de cajero de certificados y

X-ENTREGA.CER.000 para la entrega de certificados.

- Ir a la pantalla administrativa "Administración de Tipos de Pago por Círculo", agregar el tipo de pago "Consignación"
- Ir a la pantalla administrativa "Administración de Recibos en Estaciones" y asignar las secuencias de recibos para certificados al usuario PORTAL.SNR en la estación X- PORTAL.SNR
- Realizar el siguiente query para obtener el id del usuario de portal el cual entrará en la configuración del aplicativo PORTAL

```
select id_usuario from sir_op_usuario
where srio_nombre_de_usuario = 'PORTAL.SNR';
```

2. Preparación de la comunicación con el sistema de información registral SIR  
PORTAL se comunica con SIR a través de acceso a servicios remotos por medio de tres puertos: 1098, 1099 y 4444. Por lo tanto el tráfico debe ser permitido a través de estos puertos. (Verificación de puertos abiertos en el servidor de pruebas 192.168.202.9)

Modificación en Clases:

1. C:\WORK\SIR-APP\src\gov\sir\core\negocio\modelo\imprimibles\ImprimibleRecibo.java

**Linea 90**

```
/**  
 * Author: Ingeniero Diego Hernandez  
 * Modificado en 2010/02/23 by jvenegas  
 */  
private static final long serialVersionUID = -2287868441483777106L;  
private boolean imprimirDatosCertificadosMasivos =true;  
private List listaPines;
```

**Linea 163**

```
/*Adiciona Funcionalidad Boton de Pago  
 * Author: Ingeniero Diego Hernandez  
 * Modificacion en: 2010/02/23 by jvenegas  
 */  
public List getListaPines() {  
    return listaPines;  
}  
  
public void setListaPines(List listaPines) {  
    this.listaPines = listaPines;  
}
```

**Linea 303**

```
/*Adiciona Funcionalidad Boton de Pago  
 * Author: Ingeniero Diego Hernandez  
 * Modificacion en: 2010/02/23 by jvenegas  
 */  
public ImprimibleRecibo(String fechaImpresion,String tipoImprimible) {  
    super(tipoImprimible);  
    setTransferObject(pago);  
  
    // prevenir, cuando se edite y llame a otro constructor  
    AppletLogger loggerImpl = AppletLoggerImpl1.getAppletLogger();  
    if (null == this.logger) {  
        this.logger = loggerImpl;  
    }  
    logger.debug("Start Executing Super Method");  
    //this.pago = pago;  
    this.fechaImpresion = fechaImpresion;  
    //this.solicitud = pago.getLiquidacion().getSolicitud().getTurno()  
    // .getSolicitud();  
    /*if (this.solicitud == null) {  
        this.solicitud = pago.getLiquidacion().getSolicitud();  
    }*/  
    //logger.debug("pago=" + pago);  
}
```

**Linea 750**

```
/**  
 * Author: Ingeniero Diego Hernandez  
 * Modificado en 2010/02/23 by jvenegas  
 */  
if(listaPines !=null && listaPines.size()>0){  
    // Modificado 2010/02/23 by jvenegas para Boton de Pago  
    this.imprimirLinea(ImprimibleConstantes.PLANO, "");  
    Iterator iter = listaPines.iterator();  
    while(iter.hasNext()){  
        String[] linea = ((String)iter.next()).split(":");  
        this.imprimirLinea(ImprimibleConstantes.TITULO2, linea[0],false);  
  
        this.imprimirLinea(ImprimibleConstantes.TITULO2,(ImprimibleConstantes.MARGEN_IZQ)*6,linea[1],true);  
    }  
    this.imprimirLinea(ImprimibleConstantes.PLANO, "");  
    this.imprimirLinea(ImprimibleConstantes.PLANO, "");  
}
```

**Linea 3426**

```

* Modificado 2010/02/23 by jvenegas para Boton de Pago
* @param imprimirDatosCertificadosMasivos the imprimirDatosCertificadosMasivos to set
*/
public void setImprimirDatosCertificadosMasivos(
    boolean imprimirDatosCertificadosMasivos) {
    this.imprimirDatosCertificadosMasivos = imprimirDatosCertificadosMasivos;
}
/*

```

2. C:\WORK\SIR-APP\src\gov\sir\core\negocio\modelo\imprimibles\ImprimibleCertificado.java

### Línea 53

```

/*Adiciona Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 20/04/2010 by jvenegas */
private String pinCertificado = "";

```

### Línea 118

```

if(complementacion.indexOf(AMPLIACION_DE_TRADICION)!=-1){
    String[] complementaciones = complementacion.split(AMPLIACION_DE_TRADICION);
    for(int i=0; i< complementaciones.length; i++){
        if(complementaciones[i]!=null && !complementaciones[i].trim().equals("")){
            if(i>0){
                complementacion = "<b>" + AMPLIACION_DE_TRADICION + "<b>";
                this.imprimirLinea(ImprimibleConstantes.PLANO, complementacion);
            }
            complementacion = complementaciones[i].trim();
            this.imprimirLinea(ImprimibleConstantes.PLANO, complementacion);
        }
    }
} else{
    complementacion = complementacion.trim();
    this.imprimirLinea(ImprimibleConstantes.PLANO, complementacion);
}

```

### Línea 165

```
this.imprimirSalvedadesFolio(this.folio);
```

### Línea 171

```
int tamFinDoc = getTamanoFinDocumento();
```

### Línea 466

```

/*Adiciona Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 20/04/2010 by jvenegas */
if(pinCertificado!=null && !pinCertificado.trim().equals("")){
    this.imprimirLinea(ImprimibleConstantes.TITULO1,
        //ImprimibleConstantes.MARGEN_IZQ - 2)* 4,
        "Certificado Generado con el Pin No: " + pinCertificado, false);
}

```

### Línea 487

```
this.imprimirLinea(ImprimibleConstantes.PLANO, "");
```

### Línea 492

```
this.imprimirLinea(ImprimibleConstantes.TITULO1, "");
```

### Línea 497

```
this.imprimirLinea(ImprimibleConstantes.PLANO, "");
```

### Línea 647

```

/*Adiciona Funcionalidad Boton de Pago
 * Author: Ingeniero Diego Hernandez
 * Modificacion en: 2010/02/23 by jvenegas
 */

```

```
public String getPinCertificado() {
```

```

        return pinCertificado;
    }
    public void setPinCertificado(String pinCertificado) {
        this.pinCertificado = pinCertificado;
    }
}

```

3. C:\WORK\SIR-APP\src\gov\sir\core\negocio\modelo\imprimibles\base\HojaImprimible.java

### Línea 181

```
boolean textoEmpezo = false;
```

### Línea 198

```
int tipoFuente;
int tamFuente;
```

### Línea 331

```
/*Adiciona Funcionalidad Boton de Pago
 * Author: Ingeniero Diego Hernandez
 * Modificacion en: 20/04/2010 by jvenegas
 */
```

### Línea 373

```
/*Adiciona Funcionalidad Boton de Pago
 * Author: Ingeniero Diego Hernandez
 * Modificacion en: 20/04/2010 by jvenegas
 */
```

4. C:\WORK\SIR-APP\src\gov\sir\forseti\ejb\ForsetiServiceEJBBean.java

### Línea 2200

```
public boolean deshacerCambiosFolioByTurno(TurnoPk turnoID, Usuario usuario, boolean desbloquear) throws ForsetiException {
    return forseti.deshacerCambiosFolioByTurno(turnoID, usuario, desbloquear);
}
/*Adiciona Funcionalidad Boton de Pago
 * Author: Ingeniero Diego Hernandez
 * Modificacion en: 2010/02/23 by jvenegas
 *
 * @ejb:interface-method view-type="both"
 */
```

5. C:\WORK\SIR-APP\src\gov\sir\forseti\interfaz\ForsetiServiceInterface.java

### Línea 2279

```
/*Adiciona Funcionalidad Boton de Pago
 * Author: Ingeniero Diego Hernandez
 * Modificacion en: 2010/02/23 by jvenegas
 *
 * Retorna una lista de matrículas
 * @param idMatricula
 * @return
 * @throws Throwable
 */
public List getAlertasMatriculas(String idMatricula) throws Throwable;
```

6. C:\WORK\SIR-APP\src\gov\sir\forseti\dao\impl\jdogenie\JDOGenieFolioDAO.java

```
/*Adiciona Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 23/02/2010 by jvenegas */
public List getAlertasMatriculas(String idMatricula) throws DAOException{
    List alertas = new ArrayList();
    try{
        String[] consultas = consultaBD(idMatricula);
        if(consultas!=null && consultas.length == 8){
            if(!consultas[0].equals ""){
                alertas.add(armarMensaje(consultas[0], "DEMANDA VIGENTE"));
            }
            if(!consultas[1].equals ""){
                alertas.add(armarMensaje(consultas[1], "EMBARGO VIGENTE"));
            }
        }
    }
```

```

        if(!consultas[2].equals("")){           alertas.add(armarMensaje(consultas[2], "PATRIMONIO VIGENTE"));
        }
        if(!consultas[3].equals("")){           alertas.add(armarMensaje(consultas[3], "AFECTACION VIGENTE"));
        }
        if(!consultas[4].equals("")){           alertas.add(armarMensaje(consultas[4], "VALORIZACION VIGENTE"));
        }
        if(!consultas[5].equals("")){           alertas.add(armarMensaje(consultas[5], "HIPOTECA VIGENTE"));
        }
        if(!consultas[6].equals("")){           alertas.add(armarMensaje(consultas[6], "OFERTA VIGENTE"));
        }
        if(!consultas[7].equals("")){           alertas.add(armarMensaje(consultas[7], "FALSA TRADICION"));
        }

        String anotacionesInvalidas = consultaAnotacionesInvalidas(idMatricula);

        if(anotacionesInvalidas!=null && !anotacionesInvalidas.trim().equals(""))
            alertas.add(anotacionesInvalidas);
    }
}catch(DAOException e){
    Log.getInstance().error(JDOGenieFolioDAO.class, e.getMessage());
    throw e;
}catch(Exception e){
    Log.getInstance().error(JDOGenieFolioDAO.class, e.getMessage());
    throw new DAOException(e.getMessage(),e);
}
return alertas;
}

/*Adiciona Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 23/02/2010 by jvenegas */
private String armarMensaje(String datos, String complemento){
    String mensaje = null;
    if(datos!=null && !datos.trim().equals ""){
        if((datos.split(",")).length > 2){
            int posicion = datos.lastIndexOf(',');
            mensaje = "LAS ANOTACIONES " + datos.substring(1,posicion) +
                " y" + datos.substring(posicion+1) + " TIENEN " + complemento;
        }else{
            mensaje = "LA ANOTACION " + datos.substring(1) + " TIENE " + complemento;
        }
    }
    return mensaje;
}

/*Adiciona Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 23/02/2010 by jvenegas */
private String[] consultaBD(String idMatricula)throws DAOException{
    PersistenceManager pm = AdministradorPM.getPM();
    pm.currentTransaction().begin();
    String[] lstConsulta = null;
    VersantPersistenceManager jdoPM = (VersantPersistenceManager) pm;
    Connection connection = null;
    connection = jdoPM.getJdbcConnection(null);
    //SELECT realizado a las tablas SIR_NE_ANOTACION y SIR_NE_NATURALEZA_JURIDICA para cargar las consultas
    //de alertas para una matricula.
    String sql = "SELECT N.ID_GRUPO_NAT_JURIDICA,A.NTCN_ORDEN,N.ID_DOMINIO_NAT_JUR " +
        "FROM SIR_NE_ANOTACION A,SIR_NE_NATURALEZA_JURIDICA N " +
        "WHERE A.ID_MATRICULA = ? " +
        "AND A.ID_NATURALEZA_JURIDICA = N.ID_NATURALEZA_JURIDICA " +
        "AND (N.ID_NATURALEZA_JURIDICA IN (SELECT ID_NATURALEZA_JURIDICA " +
        "FROM SIR_NE_NATURALEZA_JURIDICA WHERE ID_DOMINIO_NAT_JUR IN " +
        "('H','A','P','D','E','V','O')) " +
        "OR N.ID_NATURALEZA_JURIDICA IN ('600','601','610','611','612') " +
        "OR N.ID_NATURALEZA_JURIDICA LIKE '06%' ) " +
        "AND A.ID_ANOTACION NOT IN (SELECT ID_ANOTACION1 " +
        "FROM SIR_NE_REL_CANCELACION " +
        "WHERE ID_MATRICULA = ?) " +
        "AND A.ID_ESTADO_AN='V' " +
        "ORDER BY 2";

    try{

```

```

List demanda = new ArrayList();
List embargo = new ArrayList();
List patrimonio = new ArrayList();
List afectacion = new ArrayList();
List valorizacion = new ArrayList();
List hipoteca = new ArrayList();
List oferta = new ArrayList();
List falsaTradicion = new ArrayList();
PreparedStatement pst = connection.prepareStatement(sql);
pst.setString(1,idMatricula);
pst.setString(2,idMatricula);
ResultSet rs = pst.executeQuery();
//List en el que se cargaran los resultados de la consulta.
//Este será un List<String[]> en el que el String[] tendrá
//4 posiciones [0] = id_naturaleza_juridica, [1] = id_anotacion
//[2] = ntcn_orden, [3] = id_dominio_nat_jur
while (rs.next()) {
    String idGrupoNatJuridica = rs.getString(1);
    String ntcnOrden = rs.getString(2);
    String idDominioNatJur = rs.getString(3);
    while(ntcnOrden.startsWith("0"))ntcnOrden = ntcnOrden.substring(1);
    if(idDominioNatJur.trim().equals("D")) demanda.add(new Integer(ntcnOrden)); else
    if(idDominioNatJur.trim().equals("E")) embargo.add(new Integer(ntcnOrden));
    else if(idDominioNatJur.trim().equals("P")) patrimonio.add(new Integer(ntcnOrden));
    else if(idDominioNatJur.trim().equals("A")) afectacion.add(new Integer(ntcnOrden));
    else if(idDominioNatJur.trim().equals("V")) valorizacion.add(new Integer(ntcnOrden));
    else if(idDominioNatJur.trim().equals("H")) hipoteca.add(new Integer(ntcnOrden));
    else if(idDominioNatJur.trim().equals("O")) oferta.add(new Integer(ntcnOrden));
    else
}
if(idGrupoNatJuridica.equals(CGrupoNaturalezaJuridica.FALSA_TRADICION))falsaTradicion.add(new Integer(ntcnOrden));
}
String[] aux = {list2Str(demanda),list2Str(embargo),list2Str(patrimonio),
               list2Str(afectacion),list2Str(valorizacion),list2Str(hipoteca),
               list2Str(oferta),list2Str(falsaTradicion)};
lstConsulta = aux;
rs.close();
pst.close();
pm.currentTransaction().commit();
} catch (SQLException e) {
    if (pm.currentTransaction().isActive()) {
        pm.currentTransaction().rollback();
    }
    Log.getInstance().error(JDOGenieFolioDAO.class, e.getMessage());
    throw new DAOException("Error SQL: " + e.getMessage(), e);
} catch (Exception e) {
    if (pm.currentTransaction().isActive()) {
        pm.currentTransaction().rollback();
    }
    Log.getInstance().error(JDOGenieFolioDAO.class, e.getMessage());
    throw new DAOException("Error SQL: " + e.getMessage(), e);
} finally{
    if(connection!=null){
        try {
            connection.close();
        } catch (SQLException e1) {
            Log.getInstance().error(JDOGenieFolioDAO.class, e1.getMessage());
        }
    }
    pm.close();
}
return lstConsulta;
}

/*Adiciona Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 23/02/2010 by jvenegas */
private String list2Str(List list){
    if(list != null && list.size()>0){
        Collections.sort(list);
        return "," + list.toString().substring(1,list.toString().length()-1);
    }
    return "";
}

/*Adiciona Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 23/02/2010 by jvenegas */
private String consultaAnotacionesInvalidas(String idMatricula)throws DAOException{
    PersistenceManager pm = AdministradorPM.getPM();
}

```

```

String rta = null;
List<res> = new ArrayList();
AnotacionEnhanced anota;

try{
    //Se debe buscar por algún criterio las anotaciones
    pm.currentTransaction().begin();
    VersantQuery query = (VersantQuery) pm.newQuery(AnotacionEnhanced.class);
    query.declareParameters("String idMatricula");
    query.setRandomAccess(true);
    query.setOrdering("ordenLPAD ascending");
    query.setFilter("this.idMatricula == idMatricula && this.estado.idEstadoAn == 'A'");
    res = (List<res>) query.execute(idMatricula);
    if(res!=null && res.size()>0){
        String aux = "";
        int tamano = res.size();
        for(int i=0 ;i<tamano; i++){
            anota = (AnotacionEnhanced)res.get(i);
            String orden = anota.getOrden();
            while(orden.startsWith("0"))orden = orden.substring(1);
            if(i == 0){
                aux += orden;
            }else if(i == tamano-1){
                aux += " y " + orden;
            }else{
                aux += ", " + orden;
            }
        }
        if(tamano > 1){
            rta = "LAS ANOTACIONES " + aux + " SON INVALIDAS";
        }else{
            rta = "LA ANOTACION " + aux + " ES INVALIDA";
        }
    }
} catch (JDOException e) {
    if (pm.currentTransaction().isActive()) {
        pm.currentTransaction().rollback();
    }
    Log.getInstance().error(JDOGenieFolioDAO.class, e.getMessage());
    throw new DAOException(e.getMessage(), e);
} catch (Throwable e) {
    if (pm.currentTransaction().isActive()) {
        pm.currentTransaction().rollback();
    }
    Log.getInstance().error(JDOGenieFolioDAO.class, e.getMessage());
    throw new DAOException(e.getMessage(), e);
} finally {
    if (pm.currentTransaction().isActive()) {
        pm.currentTransaction().commit();
    }
    pm.close();
}
return rta;
}

7. C:\WORK\SIR-APP\src\gov\sir\forseti\dao\FolioDAO.java
/*Adiciona Funcionalidad Boton de Pago
 * Author: Ingeniero Diego Hernandez
 * Modificacion en: 2010/02/23 by jvenegas
 */
public List getAlertasMatriculas(String idMatricula) throws DAOException;

8. C:\WORK\SIR-APP\src\gov\sir\forseti\ForsetiService.java
/*Adiciona Funcionalidad Boton de Pago
 * Author: Ingeniero Diego Hernandez
 * Modificacion en: 2010/02/23 by jvenegas
 */
public List getAlertasMatriculas(String idMatricula) throws ForsetiException{
try {
    return forseti.getFolioDAO().getAlertasMatriculas(idMatricula);
} catch (DAOException e) {
    ForsetiException fe = new ForsetiException("Fallo en la consulta de alertas del folio", e);
    fe.setHashErrores(e.getHashErrores());
    Log.getInstance().error(ForsetiService.class,fe.getMessage(), e);
}
}

```

```
        throw fe;
    }
```

```
9. C:\WORK\SIR-APP\src\gov\sir\print\server\dao\impl\jogenie\JDOGenieImpresionDAO.java
```

## Línea 59

```
import java.sql.Blob;
```

## Línea 661

```
/* Modificacion 2010-03-16 por jvnegas para Boton de Pago */
```

```
public Imprimible getImprimible(ImprimiblePk id) throws DAOException{
    Log.getInstance().error(this.getClass(),"Ingresa imprimible id =" + id.idImprimible);
    PersistenceManager pm = AdministradorPM.getPM();
    Imprimible rta = null;
    ImprimibleEnhanced impresionVal = null;
    ImprimibleEnhancedPk idEnhanced = new ImprimibleEnhancedPk();
    idEnhanced.idImprimible = id.idImprimible;
    byte[] arreglo = null;

    try{
        impresionVal = this.getBundle(idEnhanced, pm);
        if(impresionVal!=null){
            Log.getInstance().error(this.getClass(),"ImpresionVal no nulo =" + impresionVal.getIdImprimible());
            pm.makeTransient(impresionVal);
            arreglo = impresionVal.getDatosImprimible();
        }
    } catch (JDOException e) {
        Log.getInstance().error(this.getClass(),e.getMessage());
        throw new DAOException(e.getMessage(), e);
    } catch (Throwable e) {
        if (pm.currentTransaction().isActive()) {
            pm.currentTransaction().rollback();
        }
        throw new DAOException(e.getMessage(), e);
    }finally{
        pm.close();
    }
    if(impresionVal!=null){
        rta = (Imprimible)impresionVal.toTransferObject();
        rta.setDatosImprimible(arreglo);
        Log.getInstance().error(this.getClass(),"valor datosImprimible =" + rta.getDatosImprimible());
    }
    Log.getInstance().error(this.getClass(),"valor rta =" +(rta==null?"nulo":rta.toString()));
    return rta;
}

/*
 * public Imprimible getImprimible(ImprimiblePk id) throws DAOException{
```

## Línea 1264

```
/* Modificado 2010-03-15 po jvenegas para Boton de Pago */
```

```
protected ImprimibleEnhanced getBundle(
    ImprimibleEnhancedPk oid, PersistenceManager pm) throws DAOException {
    ImprimibleEnhanced rta = null;
    Query query = null;
    Collection col = null;
    try {
        query = pm.newQuery(ImprimibleEnhanced.class);
        Log.getInstance().error(this.getClass(),"getBundle query " + query.toString());
        query.setFilter("idImprimible==" + oid.idImprimible);
        Log.getInstance().error(this.getClass(),"getBundle idImprimible " + oid.idImprimible);
        col = (Collection)query.execute();
        Iterator iter = col.iterator();
        while(iter.hasNext()){
            rta = (ImprimibleEnhanced)iter.next();
        }
    } catch (JDOObjectNotFoundException e) {
        rta = null;
    } catch (JDOException e) {
        Log.getInstance().error(this.getClass(),e.getMessage());
        throw new DAOException(e.getMessage(), e);
    }finally{
```

```

        query.closeAll();
        col = null;
    }
Log.getInstance().error(this.getClass(),"getBundle valor rta "+(rta==null?"nulo":rta.toString()));
return rta;
}
/*     protected ImprimibleEnhanced getBundle(

```

### Línea 1691

```

    public byte[] getImprimibleBytes(ImprimiblePk id) throws DAOException {
        byte[] rta = null;
        VersantPersistenceManager pm =null;
        String      sql = null;
        Connection   con = null;
        PreparedStatement pst = null;
        ResultSet    rs = null;

        try {
            pm = (VersantPersistenceManager) AdministradorPM.getPM();
            pm.currentTransaction().begin();
            sql = "select MPRM_DATOS from SIR_OP_IMPRIMIBLE where id_imprimible = ?";
            con = pm.getJdbcConnection(null);
            pst = con.prepareStatement(sql);
            pst.setInt(1, id.idImprimible);
            rs = pst.executeQuery();
            Blob blob = null;
            if (rs != null && rs.next())
            {
                blob = rs.getBlob("MPRM_DATOS");
            }
            if(blob != null){
                rta = blob.getBytes(1, (int) blob.length());
            }
            pm.currentTransaction().commit();
        } catch (SQLException e) {
            Log.getInstance().error(this.getClass(),"error cargando SIR_OP_IMPRIMIBLE ",e);
        } finally{
            try {
                if(rs != null){
                    rs.close();
                }
                if(pst != null){
                    pst.close();
                }
                if (con != null && !con.isClosed()){
                    con.close();
                }
            } catch (SQLException e) {
                Log.getInstance().warn(this.getClass(),"error cerrando la conexion");
            }
            if (pm.currentTransaction().isActive()){
                pm.currentTransaction().rollback();
            }
            pm.close();
        }
        return rta;
    }

```

10. C:\WORK\SIR-APP\src\gov\sir\print\server\dao\ImpresionDAO.java

### Línea 44

```

    /**
     * Author: Ingeniero Diego Hernandez
     * Modificado en 2010/02/23 by jvenegas
     */
    public byte[] getImprimibleBytes(ImprimiblePk id) throws DAOException;

```

11. C:\WORK\SIR-APP\src\gov\sir\hermod\HermodService.java

### Línea 184

```

    /* Adiciona Funcionalidad Boton de Pago
     * Author: Ingeniero Diego Henandez
     * Modificacion en: 2010/02/23 by jvenegas
     */

```

```
private static HermodService instancia;  
private HermodDAOFactory factory;
```

### Línea 201

```
/* Adiciona Funcionalidad Boton de Pago  
 * Author: Ingeniero Diego Hernandez  
 * Modificacion en: 2010/02/23 by jvenegas  
 */  
factory = HermodDAOFactory.getFactory();  
forseti = ForsetiFactory.getFactory();  
fenrir = FenrirFactory.getInstance();  
impresion = PrintFactory.getFactory();
```

### Línea 9449

```
/*Adiciona Funcionalidad Boton de Pago  
 * Author: Ingeniero Diego Hernandez  
 * Modificacion en: 2010/03/12 by jvenegas  
public Hashtable crearTurnoTransaccion(Transaccion transaccion, Ciudadano solicitante, String idUsuario, String idBanco) throws  
HermodException{  
this.getPathFirmasRegistradores(), idBanco);
```

### Línea 9577

```
/*Adiciona Funcionalidad Boton de Pago  
 * Author: Ingeniero Diego Hernandez  
 * Modificacion en: 2010/02/23 by jvenegas  
 */  
/* Obtiene un turno dado el identificador de su solicitud, si no tiene turno  
 * asociado retorna null  
 * @param solid  
 * @return  
 * @throws HermodException  
 */  
  
public Turno getTurnoBySolicitudPortal(SolicitudPk solID) throws HermodException{  
try {  
    return factory.getTurnosDAO().getTurnoBySolicitudPortal(solID);  
} catch (DAOException e) {  
    HermodException he = new  
HermodException(HermodException.TURNO_NO_OBTENIDO, e);  
    ExceptionPrinter ep = new ExceptionPrinter(he.fillInStackTrace());  
    Log.getInstance().error(this.getClass().getName(), e);  
    throw he;  
}  
}  
  
/*Adiciona Funcionalidad Boton de Pago  
 * Author: Ingeniero Diego Hernandez  
 * Modificacion en: 2010/03/12 by jvenegas  
 * Obtiene datos de impresion  
 * retorna arreglo caracter  
 * @param solid  
 * @return  
 * @throws HermodException  
 */  
  
/* Modificado 2010-03-16 by jvenegas para Boton de Pago */  
public byte[] getDatosImprimible(Integer idImprimible) throws HermodException {  
byte[] imprimible= null;  
try{  
Log.getInstance().debug(this.getClass(),"valor idImprimible =" +idImprimible);  
ImprimiblePk id = new ImprimiblePk();  
id.idImprimible = idImprimible.intValue();  
imprimible = impresion.getImpresionDAO().getImprimibleBytes(id);  
}catch(Exception e){  
throw new HermodException(e.getMessage(), e);  
}  
return imprimible;  
}
```

## Línea 3692

```
/*Adiciona Funcionalidad Boton de Pago
 * Author: Ingeniero Diego Hernandez
 * Modificacion en: 2010/02/23 by jvenegas

 /**
 * @ejb:interface-method view-type="both"
 */
 public Hashtable crearTurnoTransaccion(Transaccion transaccion, Ciudadano solicitante, String idUsuario, String idBanco) throws
HermodException{
    return hermod.crearTurnoTransaccion(transaccion, solicitante, idUsuario, idBanco);

/** Modificacion en: 2010/02/23 by jvenegas
 * @ejb:interface-method view-type="both"
 */
 public Hashtable procesarPagoVUR(Transaccion transaccion, Ciudadano solicitante, String idBanco, String idUsuarioSIR, String
cedula) throws HermodException{
    return hermod.procesarPagoVUR(transaccion, solicitante, idBanco, idUsuarioSIR, cedula);
}

/** Modificacion en: 2010/02/23 by jvenegas
 * @ejb:interface-method view-type="both"
 */
public ImprimibleCertificado getImprimible(Integer idImprimible)throws HermodException{
    return hermod.getImprimible(idImprimible);
}

/** Modificacion en: 2010/02/23 by jvenegas
 * @ejb:interface-method view-type="both"
 */
public ImprimibleRecibo getRecibo(Integer idImprimible) throws HermodException{
    return hermod.getRecibo(idImprimible);
}

/** Modificacion en: 2010/02/23 by jvenegas
 * @ejb:interface-method view-type="both"
 */
public List getTurnosByFechaAndCirculoMinusMasivos(Proceso proceso, Fase fase, Date fecha, Circulo circulo) throws
HermodException {
    return hermod.getTurnosByFechaAndCirculoMinusMasivos(proceso, fase, fecha, circulo);
}

/** Modificacion en: 2010/02/23 by jvenegas
 * @see
gov.sir.hermod.interfaz.HermodServiceInterface#getTurnoBySolicitudPortal(gov.sir.core.negocio.modelo.SolicitudPk)
 * @ejb:interface-method view-type="both"
 */
public Turno getTurnoBySolicitudPortal(gov.sir.core.negocio.modelo.SolicitudPk solID) throws HermodException {
    return hermod.getTurnoBySolicitudPortal(solID);
}

/** Modificacion en: 2010/02/23 by jvenegas
 * @ejb:interface-method view-type="both"
 */
public byte[] getDatosImprimible(Integer idImprimible) throws HermodException {
    return hermod.getDatosImprimible(idImprimible);
}
/*Final Adiciona Funcionalidad Boton de Pago*/
```

13. C:\WORK\SIR-APP\src\gov\sir\hermod\interfaz\HermodServiceInterface.java

## Línea 4543

```
/*Adiciona Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 23/02/2010 by jvenegas */
    public ImprimibleCertificado getImprimible(Integer idImprimible) throws Throwable;
    public ImprimibleRecibo getRecibo(Integer idImprimible) throws Throwable;
/*Adiciona Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 23/02/2010 by jvenegas */
    /**
     * Metodo encargado de la creacion de turno de certificado masivo con certificados individuales asociados de orden nacional
     *
     * @param transaccion
     * @param solicitante
     * @return Tabla con los turnos, el recibo de pago, y los identificadores de los certificados
     * @throws Throwable
     */
```

```

    public Hashtable crearTurnoTransaccion(Transaccion transaccion, Ciudadano solicitante, String idUsuario, String idBanco)
throws Throwable;

/*Adiciona Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 23/02/2010 by jvenegas */
 /**
 * Metodo encargado de procesar el pago por ventanilla unica de registro
 *
 * @param transaccion
 * @param solicitante
 * @param idBanco
 * @return Tabla con el turno creado y el imprimible recibo
 * @throws DAOException
 */
 public Hashtable procesarPagoVUR(Transaccion transaccion, Ciudadano solicitante, String idBanco, String idUsuarioSIR,
String cedula) throws Throwable;

/*Adiciona Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 23/02/2010 by jvenegas */
 /**
 * Obtiene un turno dado el identificador de su solicitud, si no tiene turno
 * asociado retorna null
 * @param solID
 * @return
 * @throws Throwable
 */
 public Turno getTurnoBySolicitudPortal(SolicitudPk solID) throws Throwable;
 /**
 * Trae Datos de impresion
 * retorna arreglo caracter
 * @param solID
 * @return
 * @throws Throwable,java.rmi.RemoteException
 */
 public byte[] getDatosImprimible(Integer idImprimible) throws Throwable;
}

```

14. C:\WORK\SIR-APP\src\gov\sir\hermod\dao\TurnosPortalDAO.java

### Línea 3

```

/*Nueva Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 23/02/2010 by jvenegas */

String pathFirmasRegistradores, String idBanco) throws DAOException;
```

15. C:\WORK\SIR-APP\src\gov\sir\hermod\dao\TurnosDAO.java

### Línea 771

```

 /**
 * Author: Ingeniero Diego Hernandez
 * Modificado en 2010/02/23 by jvenegas
 */
 public Turno getTurnoBySolicitudPortal(SolicitudPk solID) throws DAOException;
```

16. C:\WORK\SIR-APP\src\gov\sir\hermod\dao\impl\jdomenie\JDOTurnosPortalDAO.java

```

/*Nueva Funcionalidad Boton de Pago Author: Ingeniero Diego Hernandez Modificacion en: 23/02/2010 by jvenegas */
import gov.sir.core.negocio.modelo.DocPagoEfectivo;
import gov.sir.core.negocio.modelo.jdomenie.DocPagoEfectivoEnhanced;
```

### Línea 160

```
String pathFirmasRegistradores, String idBanco) throws DAOException{
```

### Línea 181

```

/***
 * Author: Ingeniero Diego Hernandez
 * Modificado en 2010/02/23 by jvenegas
 */
if(banco==null)
    banco = this.getBancoByID(new BancoEnhancedPk("19"),pm);
DocPagoConsignacionEnhanced cons = new DocPagoConsignacionEnhanced(banco, "", "", 
Long.toString(transaccion.getNumTransaccion()), transaccion.getValor());
cons.setFecha(this.getFecha());
```

### Línea 351

```
/**
```

```

 * Author: Ingeniero Diego Hernandez
 * Modificado en 2010/02/23 by jvenegas
 */
protected String getFecha() {
    Calendar c = Calendar.getInstance();
    int dia;
    int ano;
    int hora;
    String min;
    String seg;
    String mes;
    dia = c.get(Calendar.DAY_OF_MONTH);
    mes = ImprimibleConstantes.MESES[c.get(Calendar.MONTH)]; //0-Based
    ano = c.get(Calendar.YEAR);
    hora = c.get(Calendar.HOUR_OF_DAY);
    if (hora > 12) {
        hora -= 12;
    }
    min = formato(c.get(Calendar.MINUTE));
    seg = formato(c.get(Calendar.SECOND));
    String fechaImp = "" + dia + " de " + mes + " de " + ano;
    return fechaImp;
}
/**/

```

### Línea 415

```

 /** * Author: Ingeniero Diego Hernandez
 * Modificado en 2010/02/23 by jvenegas
 */
DocumentoPagoEnhanced cons = null;
if(banco == null){
    cons = new DocPagoEfectivoEnhanced(transaccion.getValor());
} else{
cons = new DocPagoConsignacionEnhanced(banco, "", "", Long.toString(transaccion.getNumTransaccion()), transaccion.getValor());
}

//DocPagoConsignacionEnhanced cons = new DocPagoConsignacionEnhanced(banco, "", "", 
Long.toString(transaccion.getNumTransaccion()), transaccion.getValor());

```

17. C:\WORK\SIR-APP\src\gov\sir\hermod\dao\HermodDAOFactory.java

### Línea 99

```

 /*Adiciona Funcionalidad Boton de Pago
 * Author: Ingeniero Diego Hernandez
 * Modificacion en: 2010/02/23 by jvenegas
 */

```

public abstract TurnosPortalDAO getTurnosPortalDAO();

## 064 Modificación adición del código fuente en la aplicación SIR

### **REQUERIMIENTO**

Buenas tardes Ingeniero.(Correo 23/06/2009)

Para la modificación realizada en el aplicativo (forma de pago en efectivo y en consignación) falto la modificación de 4 clases del aplicativo SIR por lo que no se está reflejando el pago en efectivo. Podría por favor realizar la solicitud al personal de ETB para que realicen la modificación de estas.

Buenas tardes, Ing. Venegas: (Correo Martes, 23 de Junio de 2009)

De manera atenta, solicito su colaboración para realizar los ajustes solicitados para implementar la nueva versión de Supercades.

### **DESARROLLO**

Este caso esta relacionado con el requerimiento 185, por tal motivo se cierra acta.

## 147 Reporte Estadísticas Generales

**REQUERIMIENTO**

Se requiere crear reporte estadístico de los diferentes procesos, turnos generados, matriculas generadas que se llevan en cada circulo registral, asi:

El aplicativo debe permitir seleccionar en un rango de fecha del cual se requiere la información y que en el imprimible se refleje lo siguiente :

Oficina que genera el reporte – (Encabezado general para todos los reporte)

Titulo “Reporte estadísticas Generales”

Fecha de la cual se genero el reporte

Fecha de impresión

Página

También se solicita que el Reporte Estadísticas Generales se asocie al Rol CONSULTA\_ESTADISTICA\_SNR y al Administrador Regional.

Luego se listara la información asi:

**MATRICULAS:** (total de matriculas generadas en el rango de fecha). Ultima Numero de matricula Generado. Total de matriculas del circulo hasta la fecha final del rango. Esta información de matriculas debe ir en un solo bloque

**CORRECCIONES:** total turnos radicados en el rango de fecha seleccionado. Ultimo turno de correcciones en el rango de fecha. Total de correcciones desanotadas tomadas del total de turnos radicados en el rango de fecha. Total de correcciones desanotadas en el rango de fecha (total anterior mas las correcciones generadas en fechas anteriores al rango de fecha seleccionado y que fueron desanotadas en el rango seleccionado).

Luego para los procesos de Certificados, documentos, Consultas y Fotocopias deberan listarse en bloques separados para cada proceso y cada bloque tendrá las siguientes columnas y especificaciones:

Las columnas con los siguientes títulos:

Proceso	Cantidad de Turnos	Ultimo Turno Radicado	Valor de los derechos (\$)
---------	--------------------	-----------------------	----------------------------

**CERTIFICADOS**

Certificados Inmediatos locales

Certificados Inmediatos Nacionales

Certificados Antiguo Sistema

Certificados de Pertenencia

Certificados de Ampliación de la Tradición

TOTAL CERTIFICADOS ( Se totaliza la columna cantidad de turnos y valor de derechos).

Certificados Tarifa Exenta

Certificados Tarifa Normal

Certificados Tarifa Especial

Certificados Tarifa Media

Certificados Anulados

Certificados de Correcciones

Certificados Devueltos al publico

El numero del ultimo turno radicado del proceso de certificados debe mostrarse frente al titulo Certificados. (no en cada item de los certificados).

**DOCUMENTOS:**

Documentos radicados

Documentos anulados

TOTAL DOCUMENTOS (( Se totaliza la columna cantidad de turnos y valor de derechos restando los anulados)).

Documentos Inscritos

Documentos Devueltos al publico  
Documentos Inscritos parcialmente  
Documentos con Mayor valor  
El número del último turno radicado del proceso de documentos debe mostrarse frente al título Documentos.

**CONSULTAS:**

Consultas inmediatas y múltiples  
Consultas Exentas  
Consultas Anuladas  
Consulta de No propiedad  
Consulta de No propiedad Anuladas

TOTAL CONSULTAS RADICADAS ( Se totaliza la columna cantidad de turnos y valor de derechos restando las anuladas).  
El numero del ultimo turno radicado del proceso de Consultas debe mostrarse frente al título Consultas.

**FOTOCOPIAS:**

Fotocopias pagadas  
Fotocopias anuladas

TOTAL FOTOCOPIAS RADICADAS ( Se totaliza la columna cantidad de turnos y valor de derechos restando las anuladas).  
El numero del ultimo turno radicado del proceso de Fotocopias debe mostrarse frente al título Fotocopias.

**DEVOLUCIONES DE DINERO**

Devoluciones de dinero radicadas  
Devoluciones de dinero pagadas  
Este bloque no lleva totales.

Devoluciones de dinero por Documentos  
Devoluciones de dinero por certificados

El número del último turno radicado del proceso de devoluciones de dinero debe mostrarse frente al título Devoluciones de dinero.

**DESARROLLO**

- Configuración de la pantalla “Reporte-E02 - Estadística Generales”:

```
INSERT INTO SIR_OP_PANTALLA_ADMINISTRATIVA(id_pantalla_administrativa, id_tipo_pantalla, ptdm_nombre,  
ptdm_vista, ptdm_pagina)  
VALUES(1114, 13, 'Reporte E02: Estadísticas Generales', 'admin.reportes.report-E02.param-form.view', 'PG_REPORTES');
```

- Se asigna la pantalla " Reporte-E02 - Estadística Generales " al Rol

SIR\_ROL\_CONSULTA\_ESTADISTICA\_SNR con el siguiente script:

```
INSERT INTO  
SIR_OP_ROL_PANTALLA(id_rol,id_pantalla_administrativa)VALUES('SIR_ROL_CONSULTA_ESTADISTICA_SNR',1  
114);
```

- Se asigna la pantalla " Reporte-E02 - Estadística Generales " al Rol

SIR\_ROL\_ADMINISTRADOR\_REGIONAL con el siguiente script:

```
INSERT INTO  
SIR_OP_ROL_PANTALLA(id_rol,id_pantalla_administrativa)VALUES('SIR_ROL_ADMINISTRADOR_REGIONAL',11  
14);
```

- Se creo el procedimiento SP\_Reporte\_E02

Se adjunta script.

- Se creo la secuencia SEQ\_Reporte\_E02

```
CREATE SEQUENCE "SIR"."SEQ_Reporte_E02" MINVALUE 1 MAXVALUE  
999999999999999999999999999999 INCREMENT BY 1 START WITH 117 CACHE 2 NOORDER NOCYCLE ;
```

- Creación Tabla SIR\_OP\_Reporte\_E01

```
CREATE TABLE "SIR"."SIR_OP_Reporte_E02"  
(  
"DESCRIPCION" VARCHAR2(100 BYTE),  
"TOTAL" NUMBER,  
"ULT_TURNO" VARCHAR2(100 BYTE),  
"VALOR" NUMBER,  
"GRUPO" VARCHAR2(100 BYTE),  
"SEC_PROCESO" NUMBER  
) PCTFREE 10 PCTUSED 40 INITTRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING  
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645  
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)  
TABLESPACE "TBS_OWB_RUNTIME" ;
```

**Configuración Reporte E02 – Estadística Generales**

SIR-APP\web\WEB-INF\clases\org.auriga.smart.vistas.xml

<!-- AHERRENO 26/04/2010

report-E02 -->

<vista>

```

<autenticacion>true</autenticacion>
<excepcion>gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector</excepcion>
<excepcion>gov.sir.core.web.acciones.excepciones.ValidacionParametrosException</excepcion>
<nombre>admin.reportes.report-E02.param-form.view</nombre>
<parametro><llave>error</llave><valor>/WEB-INF/jspSeguro/comun/ERROR_MULTIPLE.jsp</valor></parametro>
<parametro><llave>head</llave><valor>/WEB-INF/jspSeguro/comun/HEADER_LOGOUT.jsp</valor></parametro>
<parametro><llave>body</llave><valor>/WEB-
INF/jspSeguro/administracion/ADMINREPORTES_REPORTE02_PARAMFORM.jsp</valor></parametro>
<parametro><llave>bottom</llave><valor>/WEB-INF/jspSeguro/comun/FOOTER_DEFAULT.jsp</valor></parametro>
</vista>
SIR-APP\web\WEB-INF\classes\org.auriga.smart.web.xml
<!-- AHERRENO 26/04/2010
report-E02 -->
<navegacion>
<resultado>REPORTE_E02_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK</resultado>
<url-redireccion>admin.reportes.report-E02.param-form.view</url-redireccion>
</navegacion>
SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES_REPORTE02_PARAMFORM
Archivo Nuevo.
SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java
// AHERRENO 26/04/2010
// Reporte E02
public static final String REPORTE_E02_SEND_ACTION = "REPORTE_E02_SEND_ACTION";
public static final String REPORTE_E02_BACK_ACTION = "REPORTE_E02_BACK_ACTION";

// AHERRENO 26/04/2010
// Reporte E02
public static final String REPORTE_E02_LOADCIRCULOSBYUSUARIO =
"REPORTE_E02_LOADCIRCULOSBYUSUARIO";

// AHERRENO 26/04/2010
// Declaración de las Variables para el reporte E02
public static final String REPORTE_E02_PARAM_CMDKEY = "reporte-E02.jasper";
public static final String REPORTE_E02_NOMBRE = "";
public static final String REPORTE_E02_PARAM_PFECHAINI = "V_FECHAINI";
public static final String REPORTE_E02_PARAM_PFECHAFIN = "V_FECHAFIN";
public static final String REPORTE_E02_PARAM_PCIRCULONOMBRE = "V_CIRCULO";

//AHERRENO 26/04/2010
//Reporte E02
if (REPORTE_E02_LOADCIRCULOSBYUSUARIO.equals(accion)) {
    return do_ReporteXX_LoadCirculosByUsuario_Action(request);
}

// AHERRENO 26/04/2010
// Reporte E02
if (REPORTE_E02_SEND_ACTION.equals(accion)) {
    do_ReporteE02_Send_Action(request);
}
if (REPORTE_E02_BACK_ACTION.equals(accion)) {
    do_ReporteE02_Back_Action(request);
}

//AHERRENO 26/04/2010
//Reporte E02
private void do_ReporteE02_Send_Action(HttpServletRequest request)
    throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_E02_PARAM_PFECHAINI);
    String param_02 = request.getParameter(REPORTE_E02_PARAM_PFECHAFIN);
    String param_03 = request.getParameter(REPORTE_E02_PARAM_PCIRCULONOMBRE);

    // apply validators
    if ((null == param_01) || ("").equalsIgnoreCase(param_01)) {
        exception.addError("parametro: Fecha Inicio Radicación: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
        }
    }
}

```

```

        param_01 = send1;
    } catch (Exception e) {
        exception.addError("parametro: Fecha Inicio Radicación: La fecha es inválida" + e.getMessage());
    }
}

// apply validators
if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
    exception.addError("parametro: Fecha Fin Radicación: Debe digitar un valor");
} else {
    String send1 = null;
    Date fecha = null;
    try {
        fecha = DateFormatUtil.parse(param_02);
        send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
        param_02 = send1;
    } catch (Exception e) {
        exception.addError("parametro: Fecha Fin Radicación: La fecha es inválida" + e.getMessage());
    }
}

// apply validators
if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {
    exception.addError("parametro: Circulo: Debe digitar un valor");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    }
}

do_ReporteE02_SaveState(request);

// raise -----
if (exception.getErrores().size() > 0) {
    throw exception;
}
// -----
String[] params = new String[]{
    REPORTE_XX_PARAM_CMDKEY, REPORTE_E02_PARAM_PFECHAINI,
    REPORTE_E02_PARAM_PFECHAFIN, REPORTE_E02_PARAM_PCIRCULONOMBRE
};
String[] values = new String[]{
    REPORTE_E02_PARAM_CMDKEY, param_01, param_02, param_03
};

String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_ReporteE02_RemoveState(request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}

private void do_ReporteE02_Back_Action(HttpServletRequest request) {

    do_ReporteE02_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);

} // :do_ReporteE02_Back_Action

private void do_ReporteE02_SaveState(HttpServletRequest request) {
    HttpSession session;
    String[] itemIds = new String[]{
        REPORTE_E02_PARAM_PFECHAINI, REPORTE_E02_PARAM_PFECHAFIN,
        REPORTE_E02_PARAM_PCIRCULONOMBRE
    };

    session = request.getSession();
    save_PagelItemsState(itemIds, request, session);

} // :do_ReporteE02_SaveState

private void do_ReporteE02_RemoveState(HttpServletRequest request) {
    HttpSession session;

```

```

String[] itemIds = new String[]{  

    REPORTE_E02_PARAM_PFECHAINI, REPORTE_E02_PARAM_PFECHAFIN,  

    REPORTE_E02_PARAM_PCIRCULONOMBRE  

};

```

```

session = request.getSession();  

delete_PageItemsState(itemIds, request, session);

```

```
} // :do_ReporteE02_RemoveState
```

#### JASPER – reporte-E02.jasper

Archivo Nuevo.

#### SIR-APP\src\gov\sir\core\web\navegacion\administracion\ControlNavegacionReportes.java

```

//AHERRENO 26/04/2010
//Reporte E02
public static final String REPORTE_E02_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK =
    "REPORTE_E02_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK";

```

```
//AHERRENO 26/04/2010
```

```
//Reporte E02
```

```
listaReportesJasper.add(AWReportes.REPORTE_E02_SEND_ACTION);
```

```
//AHERRENO 26/04/2010
```

```
// -- navegacion sobre pagina
```

```
if( AWReportes.REPORTE_E02_LOADCIRCULOSBYUSUARIO.equals( accion ) ){
    return REPORTE_E02_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK;
}
```

```
//AHERRENO 26/04/2010
```

```
//Reporte E02
```

```
if( AWReportes.REPORTE_E02_BACK_ACTION.equals( accion ) ) {
    return REPORTE_XX_BACK_EVENTRESP_OK;
}
```

```
//AHERRENO 26/04/2010
```

```
//Reporte E02
```

```
if( AWReportes.REPORTE_E02_BACK_ACTION.equals( accion ) ) {
    return REPORTE_XX_BACK_EVENTRESP_OK;
}
```

#### SIR-APP\src\gov\sir\hermod\dao\impl\jdomenie\JDORreportesJasperDAO.java

```
//AHERRENO 26/04/2010
```

```
//Reporte E02
```

```
//Se obtienen los parámetros para enviar al procedimiento
```

```
listaParametros = new ArrayList();
```

```
listaParametros.add(AWReportes.REPORTE_E02_PARAM_PFECHAINI);
```

```
listaParametros.add(AWReportes.REPORTE_E02_PARAM_PFECHAFIN);
```

```
listaParametros.add(AWReportes.REPORTE_E02_PARAM_PCIRCULONOMBRE);
```

```
mapaReportes.put(AWReportes.REPORTE_E02_PARAM_CMDKEY, new Object[]{
```

```
    "SELECT seq_reporte_E02.NEXTVAL FROM dual",

```

```
    "sp_reporte_E02(?, ?, ?, ?)", listaParametros,

```

```
    "DELETE FROM sir_op_reporte_E02 WHERE sec_proceso = ")");

```

#### SIR-APP\src\org.auriga.smart.vistas.xml

```
<!-- AHERRENO 26/04/2010
```

```
report-E02 -->
```

```
<vista>
```

```
<autenticacion>true</autenticacion>
```

```
<excepcion>gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector</excepcion>
```

```
<excepcion>gov.sir.core.web.acciones.excepciones.ValidacionParametrosException</excepcion>
```

```
<nombre>admin.reportes.report-E02.param-form.view</nombre>
```

```
<parametro><llave>error</llave><valor>/WEB-INF/jspSeguro/comun/ERROR_MULTIPLE.jsp</valor></parametro>
```

```
<parametro><llave>head</llave><valor>/WEB-INF/jspSeguro/comun/HEADER_LOGOUT.jsp</valor></parametro>
```

```
<parametro><llave>body</llave><valor>/WEB-
```

```
INF/jspSeguro/administracion/ADMINREPORTES_REPORT02_PARAMFORM.jsp</valor></parametro>
```

```
<parametro><llave>bottom</llave><valor>/WEB-INF/jspSeguro/comun/FOOTER_DEFAULT.jsp</valor></parametro>
```

```
</vista>
```

#### SIR-APP\src\org.auriga.smart.web.xml

```
<!-- AHERRENO 26/04/2010
```

```
report-E02 -->
```

```
<navegacion>
```

```
    <resultado>REPORTE_E02_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK</resultado>
```

```
    <url-redirect>admin.reportes.report-E02.param-form.view</url-redirect>
```

```
</navegacion>
```

#### SIR-APP\web\WEB-INF\jspseguro\administracion\old.REPORT\_LIST

```
<%-- + + + + + + + + --%>
```

```

<%-- AHERRENO 26/04/2010 --%
<%-- report_E02 --%
<tr>
<td>&nbsp;</td>
<td onMouseOver="this.style.backgroundColor="#F7F9FD" onMouseOut="this.style.backgroundColor=""><a href="admin.reportes.report-E02.param-form.view" class="links">Reporte E02: Estad&iacute;sticas Generales</a>
</td>
<td>
<%
legendHelper.setMessage( "" );
legendHelper.setRegionId( "reportE02" );
legendHelper.setRegionName( ": Estad&iacute;sticas Generales" );
try {
    legendHelper.render( out );
}
catch( java.io.IOException e ){
    out.write( e.toString() );
}
%>
</td>
<td>&nbsp;</td>
<td onMouseOver="this.style.backgroundColor="#F7F9FD" onMouseOut="this.style.backgroundColor="">&nbsp;</td>
</tr>

```

#### mapas.jar

Se actualizan los archivos:

- org.auriga.smart.vistas.xml
- org.auriga.smart.web.xml

### 091 Estadísticas RUPTA

#### **REQUERIMIENTO**

Se requiere modificar el reporte RUPTA01 Informe estadístico RUPTA, así:

El criterio de búsqueda debe ser por rango de fecha, DE- HASTA y la fecha de validación corresponderá a la **fecha de radicación de los turnos** que están con anotaciones definitivas que correspondan a los códigos, 0470, 0474, y 0927 que son los códigos implementados para rupta según resolución 4906 de 30-06-2009 turnos con resolución inscrito o devuelto al público.

Al generar el reporte el título debe mostrar el rango de fechas y el círculo (nombre de oficina y código).

El reporte debe tener tres partes: la primera parte debe relacionar la cantidad de turnos radicados en el rango de fechas, validando que el **acto asociado** en la radicación de documentos sea el **código 474**; De los documentos radicados con ese código, debe indicar la cantidad de turnos que fueron inscritos con los códigos de naturaleza jurídica 0470, 0474 y 0927; De los documentos radicados con ese código de acto, debe reportar la cantidad de turnos que fueron devueltos al público, en el rango de fecha seleccionado.

La segunda parte (consolidado del Departamento) debe mostrar la misma información que tiene actualmente para el rango de fechas seleccionado al generar el reporte, que correspondan a los códigos 0470, 0474 y 0927.

La tercera parte del reporte hace relación a las cancelaciones de RUPTA.

En el rango de fecha seleccionado debe mostrar la cantidad de turnos que correspondan a la **cancelación de las anotaciones registradas con los códigos de naturaleza jurídica 0470 y 0474**.

#### **DESARROLLO**

- Se creo el procedimiento SP\_REPORTE\_RUPTA01  
Se adjunta script.

- Se creo la secuencia SEQ\_Reporte\_RUPTA01  

```
CREATE SEQUENCE "SIR"."SEQ_Reporte_RUPTA01" MINVALUE 1 MAXVALUE
999999999999999999999999999999 INCREMENT BY 1 START WITH 117 CACHE 2 NOORDER NOCYCLE
```
- Creación Tabla SIR\_OP\_Reporte\_RUPTA01

```
CREATE TABLE "SIR"."SIR_OP_Reporte_RUPTA01"
("ID_DEPARTAMENTO" VARCHAR2(100 BYTE),
 "DEPARTAMENTO" VARCHAR2(100 BYTE),
 "ID_MUNICIPIO" VARCHAR2(100 BYTE),
 "MUNICIPIO" VARCHAR2(100 BYTE),
 "ID_VEREDA" VARCHAR2(100 BYTE),
 "VEREDA" VARCHAR2(100 BYTE),
 "SEC_PROCESO" NUMBER
) PCTFREE 10 PCTUSED 40 INITTRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "TBS_OWB_RUNTIME" ;
```

### Configuración Reporte Rupta01 – Informe Estadístico RUPTA01

SIR-APP\web\WEB-INF\classes\org.auriga.smart.vistas.xml

Línea 4459

```
<!-- AHERRENO 03/05/2010
report-RUPTA01 -->

<vista>
    <autenticacion>true</autenticacion>

    <excepcion>gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector</e
xcepcion>
        <nombre>admin.reportes.report-RUPTA01.param-form.view</nombre>
        <parametro><llave>error</llave><valor>WEB-
INF/jspSeguro/comun/ERROR_MULTIPLE.jsp</valor></parametro>
        <parametro><llave>head</llave><valor>WEB-
INF/jspSeguro/comun/HEADER_LOGOUT.jsp</valor></parametro>
        <parametro><llave>body</llave><valor>WEB-
INF/jspSeguro/administracion/ADMINREPORTES_REPORTRUPTA01_PARAMFORM.jsp</valor></parametro>
        <parametro><llave>bottom</llave><valor>WEB-
INF/jspSeguro/comun/FOOTER_DEFAULT.jsp</valor></parametro>
    </vista>
```

### SIR-APP\web\WEB-INF\jspSeguro\administracion\ADMINREPORTES\_REPORTE02\_PARAMFORM

```
<!-- AHERRENO 03/05/2010
Se agrega Condición de búsqueda Fecha Fin de Radicación
-->
<tr>
    <td width="20"></td>
    <td align="left" width="150"><div align="right" >Fecha Final Radicaci&on:</div></td>
    <td width="200" align="left">
        <% try {
            if( null == session.getAttribute( AWReportes.REPORTE_RUPTA01_PARAM_PFECHAFIN ) ) {
                session.setAttribute( AWReportes.REPORTE_RUPTA01_PARAM_PFECHAFIN, fechaAct );
            }
            TextHelper textHelper = new TextHelper();
            textHelper.setNombre(AWReportes.REPORTE_RUPTA01_PARAM_PFECHAFIN);
            textHelper.setCssClase("camposformtext");
            textHelper.setId(AWReportes.REPORTE_RUPTA01_PARAM_PFECHAFIN);
            textHelper.render(request,out);
        }
        catch(HelperException re){
            out.println("ERROR " + re.getMessage());
        }
    %>
    
```

```
        <a href="javascript:NewCal('<%=AWReportes.REPORTE_RUPTA01_PARAM_PFECHAFIN%>','ddmmmyyyy',true,24)"><
img src="<%request.getContextPath()%>/jsp/images/ico_calendario.gif" alt="Fecha" width="15" height="21"
border="0" onClick="javascript:Valores('<%=request.getContextPath()%>')"/></a>
    </td>
    <td>&nbsp;</td>
```

</tr>

SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java

Línea 906

```

// AHERRENO 03/05/2010
// Declaración de las Variables para el reporte RUPTA
// Se agrega variable REPORTE_RUPTA01_PARAM_PFECHAFIN
public static final String REPORTE_RUPTA01_PARAM_PFECHAFIN = "P_FECHA_FIN";

1. private void do_ReporteRUPTA01_Send_Action(HttpServletRequest request) throws
   ValidacionParametrosException

```

**Línea 9168**

```

// AHERRENO 03/05/2010
// Declaración de los parametros para el reporte RUPTA
// Se agrega parametro REPORTE_RUPTA01_PARAM_PFECHAFIN
String param_02 = request.getParameter(REPORTE_RUPTA01_PARAM_PFECHAFIN);

```

**Línea 9189**

```

// AHERRENO 03/05/2010
// Se valida parametro REPORTE_RUPTA01_PARAM_PFECHAFIN
if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
    exception.addError("parametro: Fecha Fin Radicación: Debe digitar un valor");
} else {
    String send1 = null;
    Date fecha = null;
    try {
        fecha = DateFormatUtil.parse(param_02);
        send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
        param_02 = send1;
    } catch (Exception e) {
        exception.addError("parametro: Fecha Fin Radicación: La fecha es inválida" + e.getMessage());
    }
}

```

**Línea 9224**

```

// AHERRENO 03/05/2010
// Se agrega parametro REPORTE_RUPTA01_PARAM_PFECHAFIN
REPORTE_XX_PARAM_CMDKEY, REPORTE_RUPTA01_PARAM_PFECHA,
REPORTE_RUPTA01_PARAM_PFECHAFIN, REPORTE_RUPTA01_PARAM_PCIRULONOMBRE

```

**Línea 9230**

```

// AHERRENO 03/05/2010
// Se agrega parametro REPORTE_RUPTA01_PARAM_PFECHAFIN "param_02"
REPORTE_RUPTA01_PARAM_CMDKEY, param_01, param_02, param_03

2. private void do_ReporteRUPTA01_SaveState(HttpServletRequest request) {

```

**Línea 9254**

```

// AHERRENO 03/05/2010
// Se agrega parametro REPORTE_RUPTA01_PARAM_PFECHAFIN
REPORTE_RUPTA01_PARAM_PFECHA, REPORTE_RUPTA01_PARAM_PFECHAFIN,
REPORTE_RUPTA01_PARAM_PCIRULONOMBRE

```

**Línea 9267**

```

// AHERRENO 03/05/2010
// Se agrega parametro REPORTE_RUPTA01_PARAM_PFECHAFIN
REPORTE_RUPTA01_PARAM_PFECHA, REPORTE_RUPTA01_PARAM_PFECHAFIN,
REPORTE_RUPTA01_PARAM_PCIRULONOMBRE

```

**JASPER – Archivos Nuevos**

1. reporte-RUPTA01.jasper
2. SUBREPORTE\_reporte-RUPTA01\_Cancelaciones.jasper

**SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDORreportesJasperDAO.java**

```

//AHERRENO 03/05/2010
//Reporte RUPTA 01
//Se obtienen los parámetros para enviar al procedimiento
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_RUPTA01_PARAM_PFECHA);
listaParametros.add(AWReportes.REPORTE_RUPTA01_PARAM_PFECHAFIN);
listaParametros.add(AWReportes.REPORTE_RUPTA01_PARAM_PCIRULONOMBRE);
mapaReportes.put(AWReportes.REPORTE_RUPTA01_PARAM_CMDKEY, new Object[]{
    "SELECT seq_reporte_rupta01.NEXTVAL FROM dual",
    "sp_reporte_rupta01(?, ?, ?, ?)", listaParametros,
    "DELETE FROM sir_op_reporte_rupta01 WHERE sec_proceso = ?");
}

```

20/04/2010

## 051 Validación de turnos siguientes al radicar nuevas entradas de registro

### REQUERIMIENTO

Validación de turnos siguientes al radicar nuevas entradas de registro.

Cuando un turno se ingresa como nueva entrada se genera un turno siguiente, si se trata de ingresar otra vez este turno el sistema valida que ya tiene un turno de nueva entrada, si este turno siguiente es anulado el sistema libera el turno anterior y permite ingresarlo para radicar una nueva entrada, sin embargo el sistema ya no valida que tiene turnos siguientes y permite ingresar este turno anterior como nueva entrada teniendo ya un turno siguiente activo creado. Se requiere escalar este caso a desarrollo para corregir este problema de forma general ya que el sistema siempre debe hacer estas validaciones para poder dar trámite correcto a los turnos. Como ejemplo se tiene el turno 2009-350-6-8148 de Ibagué el cual se ingresó como nueva entrada y se generó el turno 2009-350-6-8949 que fue anulado, se volvió a ingresar el turno 2009-350-6-8148 y generó el turno 2009-350-6-8965 que se encuentra en la fase confrontación, luego el mismo turno 2009-350-6-8148 se volvió a ingresar y generó el turno 2009-350-6-8967 que también se encuentra en la fase confrontación. ID GLPI 9162.

El sistema en este momento no valida correctamente las anulaciones en los turnos que fueron radicados como nueva entrada y esto permite que el turno anterior pueda radicar varios turnos siguientes cuando solo puede radicar un turno siguiente, si el sistema hace la validación correcta evitara que se radiquen turnos con un solo turno anterior y evitará fraudes que puede causar cualquier usuario que pueda conocer este bug del sistema.

### DESARROLLO

- Script Bases de Datos:  
NINGUNA.
- Creación De Clases:  
NINGUNA (el mismo formato de modificación clases o jsp)
- Se modificaron los siguientes archivos:

CLASE
SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\ JDOturnosDAO.java
Método:
protected TurnoEnhanced getTurnoSiguiente(String idWork, String idCírculo, PersistenceManager pm) throws DAOException

#### Código Línea 5790:

Se adicionan las siguientes líneas al método.

```
/*
 * @author: GUILLERMO CABRERA 25-02-2010
 * @change: Se adiciona la línea 5795 para definir el ordenamiento del query en forma descendente "de mayor a menor",
 * para que en la consulta traiga el turno siguiente mayor.
 * @request: No 051, caso mantis3980.
 */
query.setOrdering("anio descending, idCírculo descending, idProceso descending, idTurno descending");
```

## 071 Interés Mora Impuesto

### REQUERIMIENTO

Se requiere modificar el aplicativo SIR en la radicación de documentos con nueva entrada, en los casos en que normativamente se cambia la configuración de COBRO DE IMPUESTO DE REGISTRO, esto es que la ORIP a partir de un momento dado NO

debe recaudar el mencionado impuesto.

Lo anterior porque al ingresar el documento el aplicativo debe generar el pago de interés de mora sobre el impuesto de registro (esta generalidad esta bien para las oficinas que recaudan impuesto).

Se debe tener en cuenta lo siguiente: Una radicación de nueva entrada a un documento radicado en SIR, en una oficina que recaudaba impuesto de registro, valida los pagos automáticamente y les genera, si es el caso, el cobro de interés de mora. No obstante el aplicativo debería validar y tener en cuenta que en esa oficina actualmente NO se recauda impuesto de registro (caso de Guamo e Ibagué Tolima), por lo tanto se requiere que el SIR valide si la oficina esta parametrizada para que recaude dicho impuesto y si no lo está, no debe generar interés de mora.

### **DESARROLLO**

Solución: Se creó la validación para que no calcule los valores de recargos cuando no este habilitado el impuesto para el círculo correspondiente.

Clase: C:\WORK\SIR-APP\src\gov\sir\hermod\pagos\LiquidadorRegistro.java

Línea 724:

```
if(turnoAnt.getSolicitud().getCirculo().isCobroImpuesto()){
    obtenerValorMultasActos(actos, s ,true, turnoAnt.getFechaInicio());
}
```

## 127 Corrección cancelaciones turnos de correcciones

### **REQUERIMIENTO**

Cuando se realiza una corrección de anotaciones canceladoras, al avanzar el turno de correcciones de la fase revisar y aprobar el sistema arroja error indicando "Fallo en hacer definitivos los datos temporales, la cancelación no tiene objeto cancelada para buscarla". Se requiere analizar y revisar este caso ya que se presenta con frecuencia en las diferentes ORIPs y lo que se ha podido detectar es que las correcciones realizadas intervienen con anotaciones canceladoras. Por favor crear acta de requerimiento a desarrollo para la solución general de este caso. Id Glpi 12630.

### **DESARROLLO**

En el método JDOGenieFolioDAO.getCancelacion() solo se estaba realizando la validación con getCanceladaDef() para obtener el id de la anotación definitiva que se está cancelando, pero para muchos turnos esta anotación se encuentra temporal y se debe utilizar el método getCanceladaTmp().

Se modifica el método JDOGenieFolioDAO.getCancelacion() para que se realice la validación con el objeto correspondiente, es decir el objeto que sea diferente a null.

Clase Modificada: JDOGenieFolioDAO.java

Ubicación: WORK\SIR-APP\src\gov\sir\forseti\dao\impl\jdogenie

Código:

```
/* @author: Guillermo Cabrera.
 * @change: Esta generando la excepción del caso Mantis 2558, ya que se esta validando
```

```

 * solamente la cancelación definitiva.
 * se comentarea y se modifica por el código líneas: 21270 a 21277.

//Se busca si la cancelación temporal que se busca,
//cancela a una anotación temporal
if (can.getCanceladaDef() == null)
{
    throw new DAOException( "La cancelación no tiene objeto cancelada para buscarla");
}
String idAnota=can.getCanceladaDef().getIdAnotacion();
/*
*/
/***
 *@author: Guillermo Cabrera.
 * @change: se evalúa las anotaciones canceladas definitivas y temporales en caso
 * de que el objeto sea diferente a null se obtiene el id de la anotación cancelada.
 * en caso de que los dos objetos sean nulos se envía la excepción.
 * CASO MANTIS 2558 Requerimiento No 127
**/


String idAnota;
if (can.getCanceladaDef() != null)
    idAnota= can.getCanceladaDef().getIdAnotacion();
else if (can.getCanceladaTmp() != null)
    idAnota= can.getCanceladaTmp().getIdAnotacionTmp();
else
    throw new DAOException("La cancelación no tiene objeto cancelada para buscarla");

```

## 135 Impuesto Meta

### **REQUERIMIENTO**

Correo Electrónico Luis Emilio Romero. De manera atenta y respetuosa me permito solicitar a usted se realicen los ajustes a que haya lugar a fin de que el aplicativo SIR, permita realizar el cobro de impuesto de registro y anotación a un valor correspondiente al 0.8%, para aquellos negocios jurídicos de vivienda de interés social en el departamento del META, correspondiente a los estratos 1 y 2.

Es decir que para las ORIPS del departamento del META, deben quedar configurados dos cobros de impuesto, uno normal (1 por mil) y el de 0.8 por mil, el cajero dependiendo del documento aportado por el ciudadano debe elegir que valor cobra de impuesto.

Mañana enviaré la ordenanza mediante la cual se reglamenta esta decisión. Dicha tarifa se aplicará hasta el 31 de diciembre de 2011.

### **DESARROLLO**

Análisis: Se requiere asignar un nuevo impuesto a los actos de registro del departamento del META.

Solución: Se creara el impuesto, se llamará **IMPUESTO META**. Para realizar esta modificación fue necesario realizar los siguientes cambios en la aplicación y en base de datos:

#### a) Aplicación:

##### 1. Se modiflico el código fuente .java a:

Nombre: C:\WORK\SIR-APP\src\gov\sir\core\eventos\comun\EvnRespSistema.java

##### Linea 84:

```

 /**
 * @author: Fernando Padilla
 * @change: Requerimiento externo No. 135, Esta constante identifica los tipos de impuestos disponibles por
circulo
 */
public static final String LISTA_TIPOS_IMPUESTO_CIRCULO = "LISTA_TIPOS_IMPUESTO_CIRCULO";

```

##### 2. Se modiflico el código fuente .java a:

Nombre: C:\WORK\SIR-

APP\src\gov\sir\core\negocio\acciones\comun\ANSistema.java

Método: private EventoRespuesta initParametros(EvnSistema evento)  
throws EventoException

**Línea 375:**

```
/*
 * @author: Fernando Padilla
 * @change: Requerimiento externo No. 135, Se carga en el contexto de la
 * aplicación la lista de tipos de impuesto por círculo.
 */
try {
    this.getTiposImpuestoCirculo(evento);
} catch (Throwable e) {
    ExceptionPrinter printer = new ExceptionPrinter(e);
    Log.getInstance().error(ANSistema.class,"Ha ocurrido una excepcion " +
        "inesperada al tratar de cargar los tipos de impuestos por círculo" +
        printer.toString());
    throw new EventoException(e.getMessage(), e);
}
```

### 3. Se modifica el código fuente .java a:

Nombre: C:\WORK\SIR-

APP\src\gov\sir\core\negocio\modelo\transferutil\MakeTransferEnhaced.java

Método: public static Object makeTransfer(Object source)

**Línea 238:**

```
/*
 * @author Fernando Padilla Velez
 * @change Modificado para el caso MANTIS 135_141_Impuesto Meta. en caso de que el objeto
 * sea de la instancia gov.sir.core.negocio.modelo.jdogenie.TipoImpuestoCirculoEnhanced
 * realizara ca copia del objeto.
 */

else if (source instanceof gov.sir.core.negocio.modelo.jdogenie.TipoImpuestoCirculoEnhanced )
{
    gov.sir.core.negocio.modelo.jdogenie.TipoImpuestoCirculoEnhanced obj =
(gov.sir.core.negocio.modelo.jdogenie.TipoImpuestoCirculoEnhanced)source;
    rta = TipoImpuestoCirculoTransferCopier.deepCopy(obj,cache);
}
```

### 4. Se modifica el código fuente .java a:

Nombre: C:\WORK\SIR-

APP\src\gov\sir\core\web\helpers\registro\AgregarActosHelper.java

**Línea 6:**  
import gov.sir.core.negocio.modelo.TipoImpuestoCirculo;

**Línea 14:**  
import java.text.ParseException;  
**Línea 18 y 19:**  
import java.util.logging.Level;  
import java.util.logging.Logger;  
**Línea 26:**  
import java.util.Date;

Método: public void drawGUI(HttpServletRequest request, JspWriter out)  
throws HelperException, IOException

**Línea 72:**

```
/*
 * @author: Fernando Padilla Velez
 * @change: Modificado para el caso MANTIS 135_141_Impuesto Meta, se realiza
 * la validacion que exista el TipoImpuesto en la lista de TipoImpuestoCirculo.
 */
try {
    List<TipoImpuesto> tipoImp = (List<TipoImpuesto>)
request.getSession().getServletContext().getAttribute(WebKeys.LISTA_TIPOS_IMPUESTO);
    List<TipoImpuestoCirculo> tipoImpCir = (List<TipoImpuestoCirculo>)
request.getSession().getServletContext().getAttribute(WebKeys.LISTA_TIPOS_IMPUESTO_CIRCULO);
    List elem = new ArrayList();
    Date hoy = gov.sir.core.util.DateFormatUtil.parse(gov.sir.core.util.DateFormatUtil.format(new Date()));
    for (TipoImpuesto t : tipoImp) {
        for (TipoImpuestoCirculo tt : tipoImpCir) {
            Date fechaInicioVigencia =
gov.sir.core.util.DateFormatUtil.parse(gov.sir.core.util.DateFormatUtil.format(tt.getFechaInicioVigencia()));
            Date fechaFinVigencia =
gov.sir.core.util.DateFormatUtil.parse(gov.sir.core.util.DateFormatUtil.format(tt.getFechaFinVigencia()));
            if (tt.getIdTipoImpuesto().equals(t.getIdTipoImpuesto()) &&
```

```

tt.getIdCirculo().equals(this.circulo.getIdCirculo()) && (hoy.after(fechaInicioVigencia) ||
hoy.equals(fechaInicioVigencia))
    && (hoy.before(tt.getFechaFinVigencia()) || hoy.equals(fechaFinVigencia) ))
if(("230".equals(circulo.getIdCirculo()) || "232".equals(circulo.getIdCirculo())) &&
"IM".equals(t.getIdTipoImpuesto()) &&
!"38".equals(request.getSession().getAttribute(CActo.TIPO_ACTO))){  

    continue;  

}  

elem.add(new ElementoLista(t.getIdTipoImpuesto(), t.getNombre()));  

}
}
}
Línea 258:  

} catch (ParseException ex) {  

    Logger.getLogger(AgregarActosHelper.class.getName()).log(Level.SEVERE, null, ex);
}

```

## 5. Se modifica el código fuente .java a:

Nombre: C:\WORK\SIR-APP\src\gov\sir\core\web\InitParametrosServlet.java  
**Método:** public static void cargarListasEnContexto(Hashtable mapa, ServletContext context )

**Línea 826:**

```

/*
 * @author: Fernando Padilla Velez
 * @change: Modificado para el caso MANTIS 135_141_Impuesto Meta, se capturan
 *          la lista de TipoImpuestoCirculo y se agrega en el contexto de la
 *          aplicacion.
 */
List tipoImpuestosCirculo =
(List)mapa.get(EvnRespSistema.LISTA_TIPOS_IMPUESTO_CIRCULO);

if (tipoImpuestosCirculo == null) {
    if (context.getAttribute(WebKeys.LISTA_TIPOS_IMPUESTO_CIRCULO)
== null)
    {
        logger.warn("La lista de los tipos de impuesto por circulo no pudo
ser cargada. Se colocara una lista vacia");
        context.setAttribute(WebKeys.LISTA_TIPOS_IMPUESTO,
Collections.unmodifiableList(new ArrayList()));
        logger.info("La lista de los tipos de impuesto por circulo ha sido
cargada al contexto.");
    }
    else {
        context.setAttribute(WebKeys.LISTA_TIPOS_IMPUESTO_CIRCULO,
Collections.unmodifiableList(tipoImpuestosCirculo));
        logger.info("La lista de los tipos de impuesto por circulo ha sido cargada al
contexto.");
    }
}

```

## 6. Se modifica el código fuente .java a:

Nombre: C:\WORK\SIR-APP\src\gov\sir\core\web\WebKeys.java

**Línea 439:**

```

/*
 * @author: Fernando Padilla Velez
 * @change: Esta contante identifica los tipos de impuestos por circulo disponibles.
 */
public static final String LISTA_TIPOS_IMPUESTO_CIRCULO = "LISTA_TIPOS_IMPUESTO_CIRCULO";

```

## 7. Se modifica el código fuente .jdo de mapas.jar a:

Nombre: C:\WORK\SIR-APP\src\gov.sir.impl.jdo

**Línea 3909:**

```

<class name="TipoImpuestoCirculoEnhanced" identity-type="application" objectid-
class="TipoImpuestoCirculoEnhancedPk">
    <extension vendor-name="versant" key="jdbc-table-name"
value="SIR_OP_TIPO_IMPUESTO_CIRCULO" />
    <field name="idTipoImpuestoCirculo" primary-key="true">
        <extension vendor-name="versant" key="jdbc-column">
            <extension vendor-name="versant" key="jdbc-column-name"
value="ID_TIPO_IMPUESTO_CIRCULO" />
            <extension vendor-name="versant" key="jdbc-type" value="INTEGER" />
            <extension vendor-name="versant" key="jdbc-sql-type" value="NUMBER" />
            <extension vendor-name="versant" key="jdbc-length" value="20" />

```

```

        </extension>
    </field>
    <field name="idTipoImpuesto">
        <extension vendor-name="versant" key="jdbc-column">
            <extension vendor-name="versant" key="jdbc-type" value="VARCHAR" />
            <extension vendor-name="versant" key="jdbc-column-name" value="ID_TIPO_IMPUESTO" />
            <extension vendor-name="versant" key="jdbc-length" value="20" />
        </extension>
    </field>
    <field name="idCirculo">
        <extension vendor-name="versant" key="jdbc-column">
            <extension vendor-name="versant" key="jdbc-type" value="VARCHAR" />
            <extension vendor-name="versant" key="jdbc-column-name" value="ID_CIRCULO" />
            <extension vendor-name="versant" key="jdbc-length" value="20" />
        </extension>
    </field>
    <field name="fechaInicioVigencia">
        <extension vendor-name="versant" key="jdbc-column">
            <extension vendor-name="versant" key="jdbc-type" value="DATE" />
            <extension vendor-name="versant" key="jdbc-column-name" value="FECHA_INICIO_VIGENCIA" />
        </extension>
    </field>
    <field name="fechaFinVigencia">
        <extension vendor-name="versant" key="jdbc-column">
            <extension vendor-name="versant" key="jdbc-type" value="DATE" />
            <extension vendor-name="versant" key="jdbc-column-name" value="FECHA_FIN_VIGENCIA" />
        </extension>
    </field>
    <field name="soporte">
        <extension vendor-name="versant" key="jdbc-column">
            <extension vendor-name="versant" key="jdbc-type" value="VARCHAR" />
            <extension vendor-name="versant" key="jdbc-column-name" value="SOPORTE" />
            <extension vendor-name="versant" key="jdbc-length" value="255" />
        </extension>
    </field>
    <extension vendor-name="versant" key="jdbc-optimistic-locking" value="none" />
</class>

```

## 8. Se modifco el código fuente .java a:

Nombre: C:\WORK\SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\  
JDOVariablesOperativasDAO.java

Línea 154:

```
import gov.sir.core.negocio.modelo.jdogenie.TipoImpuestoCirculoEnhanced;
```

Método nuevo: public List getTiposImpuestoCirculo() throws DAOException

Línea 1544:

```
/***
 * Obtiene una la lista de objetos <code>TipoImpuestoCirculo</code> existentes en
 * la configuración del sistema.
 * <p>
 * El ordenamiento de los resultados se realiza alfabéticamente, utilizando
 * como criterio de ordenamiento el nombre del <code>TipoImpuestoCirculo</code>
 * @author Fernando Padilla Velez
 * @change Modificado para el caso MANTIS 135_141_Impuesto Meta
 * @return una lista de objetos <code>TipoImpuestoCirculo</code>
 * @throws DAOException
 * @see gov.sir.core.negocio.modelo.TipoImpuestoCirculo
 */
```

```
public List getTiposImpuestoCirculo() throws DAOException {
    List lista = new ArrayList();
    PersistenceManager pm = AdministradorPM.getPM();

    try {
        pm.currentTransaction().begin();
        Query q = pm.newQuery(TipoImpuestoCirculoEnhanced.class);
        Collection results = (Collection) q.execute();
        Iterator it = results.iterator();

        //Se hacen transientes los elementos de la lista.
        while (it.hasNext()) {
            TipoImpuestoCirculoEnhanced obj = (TipoImpuestoCirculoEnhanced)
            it.next();
            pm.makeTransient(obj);
            lista.add(obj);
        }
    }
```

```

        pm.currentTransaction().commit();
        q.close(results);

    } catch (Throwable e) {
        if (pm.currentTransaction().isActive()) {
            pm.currentTransaction().rollback();
            Log.getInstance().error(JDOVariablesOperativasDAO.class,e.getMessage());
            throw new DAOException(e.getMessage(), e);
        } finally {
            pm.close();
        }

        lista = TransferUtils.makeTransferAll(lista);
        return lista;
    }

```

**9. Se modifica el código fuente .java a:**

Nombre: C:\WORK\SIR-

APP\src\gov\sir\hermod\dao\VariablesOperativasDAO.java

Línea 562, se agrega en la interface lo siguiente:

```

    /**
     * @author Fernando Padilla Velez
     * @change Modificado para el caso MANTIS 135_141_Impuesto Meta, Obtiene
     * una la lista de objetos <code>TipoImpuestoCirculo</code> existentes en
     * la configuración del sistema.
     * @return una lista de objetos <code>TipoImpuestoCirculo</code>
     * @throws DAOException
     * @see gov.sir.core.negocio.modelo.TipoImpuestoCirculo
    */
    public List getTiposImpuestoCirculo() throws DAOException;

```

**10. Se modifica el código fuente .java a:**

Nombre: C:\WORK\SIR-APP\src\gov\sir\hermod\ejb\HermodServiceEJBBean.java

Línea 562, se agrega el método siguiente:

```

    /**
     * @author Fernando Padilla Velez
     * @change Modificado para el caso MANTIS 135_141_Impuesto Meta.
     * @see gov.sir.hermode.interfaz.HermodeServiceInterface#getTiposImpuestoCirculo()
     * @ejb:interface-method view-type="both"
    */
    public List getTiposImpuestoCirculo() throws HermodException {
        return hermod.getTiposImpuestoCirculo();
    }

```

**11. Se modifica el código fuente .java a:**

Nombre: C:\WORK\SIR-

APP\src\gov\sir\hermod\interfaz\HermodeServiceInterface.java

Línea 979, se agrega el método siguiente:

```

    /**
     * @author Fernando Padilla Velez
     * @change Modificado para el caso MANTIS 135_141_Impuesto Meta, Obtiene la
     * lista de tipos de Impuestos por Circulo.
     * @return una lista de objetos <code>TipoImpuestoCirculo</code> con todos sus atributos.
     * @see gov.sir.core.negocio.modelo.TipoImpuestoCirculo
     * @throws <code>Throwable</code>
    */
    List getTiposImpuestoCirculo() throws Throwable;

```

**12. Se modifica el código fuente .java a:**

Nombre: C:\WORK\SIR-APP\src\gov\sir\hermod\HermodService.java

Línea 1102, se agrega el método siguiente:

```

    /**
     * @author Fernando Padilla Velez
     * @change Modificado para el caso MANTIS 135_141_Impuesto Meta,
     * Obtiene la lista de tipos de Impuestos por circulos.
     * @return una lista de objetos <code>TipoImpuestoCirculo</code> con todos sus atributos.
     * @see gov.sir.core.negocio.modelo.TipoImpuestoCirculo
     * @throws <code>HermodException</code>
    */
    public List getTiposImpuestoCirculo() throws HermodException {
        try {
            return factory.getVariablesOperativasDAO().getTiposImpuestoCirculo();
        } catch (DAOException e) {
            HermodException he =
                new

```

```

        HermodException(HermodException.LISTA_TIPOS_IMPUESTOS_NO_OBTENIDA, e);
        Log.getInstance().error(this.getClass(),he.getMessage(), e);
        throw he;
    }
}

```

**13. Se crea la clase:**

C:\WORK\SIR-  
APP\src\gov\sir\core\negocio\modelo\transferutil\TipoImpuestoCirculoTransferCopier.java

**14. Se crea la clase:**

C:\WORK\SIR-APP\src\gov\sir\core\negocio\modelo\TipoImpuestoCirculo.java

**15. Se crea la clase:**

C:\WORK\SIR-APP\src\gov\sir\core\negocio\modelo\TipoImpuestoCirculoPk.java

**16. Se crea la clase:**

C:\WORK\SIR-  
APP\src\gov\sir\core\negocio\modelo\jdogenie\TipoImpuestoCirculoEnhanced.java

**17. Se crea la clase:**

C:\WORK\SIR-  
APP\src\gov\sir\core\negocio\modelo\jdogenie\TipoImpuestoCirculoEnhancedPk.java

**b) Base de Datos:**

1. Se crea la secuencia SEC\_SIR\_OP\_TIPO\_IMPUESTO\_CIRC en el esquema SIR, se adjunta script de creación de la secuencia en C:\WORK\Documentos\DESARROLLO\REQUERIMIENTOS SNR\DOCUMENTOS ADJUNTOS\REQUERIMIENTO\_135\SEC\_SIR\_OP\_TIPO\_IMPUESTO\_CIRC.C.sql.
2. Se crea la tabla SIR\_OP\_TIPO\_IMPUESTO\_CIRCULO en el esquema SIR, se adjunta script de creación de la tabla en C:\WORK\Documentos\DESARROLLO\REQUERIMIENTOS SNR\DOCUMENTOS ADJUNTOS\REQUERIMIENTO\_135\SIR\_OP\_TIPO\_IMPUESTO\_CIRCULO.sql.
3. Se crea un nuevo registro en la tabla SIR\_OP\_TIPO\_IMPUESTO en el esquema SIR, se adjunta script de ejecución de este insert en C:\WORK\Documentos\DESARROLLO\REQUERIMIENTOS SNR\DOCUMENTOS ADJUNTOS\REQUERIMIENTO\_135\SIR\_OP\_TIPO\_IMPUESTO.sql.
4. Se crea un nuevo registro en la tabla SIR\_OP\_TARIFA en el esquema SIR, se adjunta script de ejecución de este insert en C:\WORK\Documentos\DESARROLLO\REQUERIMIENTOS SNR\DOCUMENTOS ADJUNTOS\REQUERIMIENTO\_135\SIR\_OP\_TARIFA.sql.
5. Se crearan 159 en la tabla SIR\_OP\_TIPO\_IMPUESTO\_CIRCULO en el esquema SIR, como carga inicial de esta tabla, se adjunta script de ejecución de este insert en C:\WORK\Doecumentos\DESARROLLO\REQUERIMIENTOS SNR\DOCUMENTOS ADJUNTOS\REQUERIMIENTO\_135\SIR\_OP\_TIPO\_IMPUESTO\_CIRCULO\_DATOS.sql.

Una vez realizados estos cambios se debe recomilar la aplicación y generar el nuevo .ear, además de hacer el reemplazo del archivo C:\WORK\SIR-APP\src\gov.sir.impl.jdo

en el mapas.jar del jboss.

```
\gov\sir\core\web\helpers\registro\AgregarActosHelper.java
```

```
if(("230".equals(circulo.getIdCirculo()) || "232".equals(circulo.getIdCirculo())) &&
    "IM".equals(t.getIdTipoImpuesto()) &&
    !"38".equals(request.getSession().getAttribute(CActo.TIPO_ACTO))){  
    continue;  
}  
elem.add(new ElementoLista(t.getIdTipoImpuesto(), t.getNombre()));  
}
```

```
\gov\sir\hermod\dao\impl\jdogenie\JDOLiquidacionesDAO.java
```

```
/**  
 * @author: Fernando Padilla Velez  
 * @change: Modificado para el caso MANTIS 135_141_Impuesto Meta, Se modifica la formula  
 *          para el calculo del valor del impuesto cuando el acto sea el 38-VIVIENDA DEINTERES  
SOCIAL.  
 */  
if(acto.getTipoActo() != null &&  
    acto.getTipoActo().getIdTipoActo() != null &&  
    "38".equals(acto.getTipoActo().getIdTipoActo()) &&  
    "IM".equals(acto.getTipoImpuesto().getIdTipoImpuesto())){  
    TarifaEnhanced porcImpuesto = tarifasDAO.getTarifa (CTipoTarifa.PORCENTAJE_IMPUESTO,  
acto.getTipoImpuesto().getIdTipoImpuesto(),pm);  
    double porcentajeImpuesto = porcImpuesto.getValor();  
    valorImpuesto = cuantia * porcentajeImpuesto;  
}else{  
    //SE OBTIENE EL VALOR DEL IMPUESTO BASADO EN CUANTIA  
    TarifaEnhanced tarifaImpuesto = tarifasDAO.getTarifa(CTipoTarifa.TIPO_IMPUESTO_ACTO,  
acto.getTipoActo().getNombre()+"_"+acto.getLiquidacion().getSolicitud().getCirculo().getNombre(), pm);  
    if (tarifaImpuesto == null){  
        throw new DAOException ("No se encontró valor de impuestos para la llave:  
"+CTipoTarifa.TIPO_IMPUESTO_ACTO+" -  
"+acto.getTipoActo().getNombre()+"_"+acto.getLiquidacion().getSolicitud().getCirculo().getNombre());  
    }  
    double impuesto = tarifaImpuesto.getValor();  
    //Se obtiene el tipo de impuesto y su factor de multiplicación (exento, medio, normal)  
    TarifaEnhanced porcImpuesto = tarifasDAO.getTarifa (CTipoTarifa.PORCENTAJE_IMPUESTO,  
acto.getTipoImpuesto().getIdTipoImpuesto(),pm);  
    double porcentajeImpuesto = porcImpuesto.getValor();  
  
    valorImpuesto = (impuesto * cuantia) * porcentajeImpuesto;  
}
```

## 167 Turno Manual Sin Matrícula

### REQUERIMIENTO

Cuando se crea un turno de registro manualmente por la pantalla administrativa manual turno registro, si el turno debe ser creado sin matricula el sistema exige datos de antiguo sistema, se digitán los datos de antiguo sistema y el sistema sigue exigiendo esos datos que ya están digitados y no es posible crear el turno sin matrícula.

### DESARROLLO

- Script Bases de Datos:
  - NINGUNA
- Creación De Clases:
  - NINGUNA
- Se modificaron el siguiente archivo:

CLASE o ARCHIVO JSP:
C:\WORK\SIR-
APP\src\gov\sir\core\web\acciones\administracion\AWTurnoManualRegistro.java
Método:
liquidarRegistro
<pre>/**  * @author: Henry Gómez Rocha  * @change: Se valida que la variable que almacena la información de la caja de texto  "Comentario (Datos Antiguo Sistema)" contenga información.  **/</pre> <p><b>Código Línea 1427:</b></p> <pre>String documento_comentario_as =     request.getParameter(CDatosAntiguoSistema.DOCUMENTO_COMENTARIO_AS);     documento_comentario_as = (documento_comentario_as == null) ? ""         : documento_comentario_as;</pre> <p><b>Por:</b></p> <pre>String documento_comentario_as = request.getParameter(CSolicitudRegistro.COMENTARIO);     documento_comentario_as = (documento_comentario_as == null) ? ""         : documento_comentario_as;</pre> <p><b>Código Línea 1672:</b></p> <pre>if(documento_comentario_as == null    documento_comentario_as.equals(""))     exception.addError("Debe digitar un comentario para Antiguo Sistema"); <b>Por:</b></pre> <pre>if(comentario_as == null    comentario_as.equals(""))     exception.addError("Debe digitar un comentario para Antiguo Sistema");</pre>

### 175 Error en el reparto (Informativo)

#### **REQUERIMIENTO GLPI 15670 – MANTIS 3832**

Reparto:2010-070-145- 2010-070-146

Motivo Inconsistencia:

Se comunica Nubia Reyes Zipa e indica que cuando repartió los turnos le manifiesto error: No se pudo obtener el tipo de relación: Row not found.

Se verifica en el sistema en el Reporte Rel 01

Reporte genérico de relaciones se puede ver claramente que los turnos fueron repartidos doble vez a los calificadores encargados, en la relación generada en el sistema en el pdf.

Por favor revisar las relaciones de reparto y corregir el pdf para que la oficina continúe con el trámite, se adjunta imagen, gracias.

Grupo Tunja

**DESARROLLO**

Descripción técnica:

El reparto se ejecuto dos veces por diferente usuario.

Descripción solución:

Se borra el reparto duplicado y se pueden generar las relaciones 2010-070-144, 2010-070-145 y 2010-070-146. Informar al usuario para que genere las relaciones.

Scripts:

```
DELETE FROM "SIR"."SIR_OP_REPARTO" WHERE ANIO = '2010' and  
ID_CIRCULO = '070' and ID_PROCESO = 6 and ID_TURNO in ('1384', '1385', '1386',  
'1387', '1392', '1395')
```

```
DELETE FROM "SIR"."SIR_OP_REPARTO" WHERE id_proceso_reparto = '6702037'
```

### 176 Reporte Rel 01- el listado de turnos no se esta mostrando de acuerdo al calificador

#### REQUERIMIENTO

Se genera el reporte Rel 01 – Reporte Genérico de Relaciones de la ORIP Tunja (070) para las relaciones 2010-070-145 y 2010-070-146 y el listado de turnos no se esta mostrando de acuerdo al calificador.

#### DESARROLLO

Reporte Rel -01 (Reporte Genérico de Relaciones)

- Script Bases de Datos:

NINGUNO

- Creación De Clases:

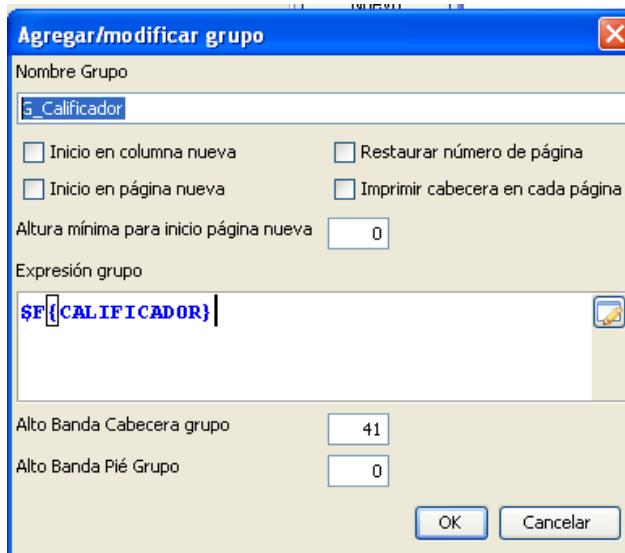
NINGUNO

- Modificación de archivos:

Se realizaron las siguientes modificaciones en el reporte Rel -01 (Reporte Genérico de Relaciones) en la cual se configuro un grupo:

Archivo reporte-41.jasper

- G\_Calificador



En la banda G\_CalificadorHEADER se ubico la sección que se debe imprimir por

calificador:

Calificador: \${CALIFICADOR}					
Turno	Fecha Radicación	Turno Anterior	Documento	Num.	Matriculas

Se realizaron pruebas con la ORIP de Tunja relación 2010-07-145

## ANTES

2010-07-06-1419 05-02-2010 11:40:16 AM ESCRITURA 2102 070-110452  
2010-07-06-1423 05-02-2010 12:00:21 PM ESCRITURA 16 070-179613  
2010-07-06-1431 05-02-2010 02:25:19 PM ESCRITURA 63 070-82879  
2010-07-06-1433 05-02-2010 02:33:35 PM ESCRITURA 2299 070-85096  
2010-07-06-1436 05-02-2010 02:42:23 PM OFICIO 18 070-179613  
05-02-2010 02:42:23 PM OFICIO 1792 070-108423  
05-02-2010 02:42:23 PM OFICIO 1792 070-116478  
05-02-2010 02:42:23 PM OFICIO 1792 070-1849  
2010-07-06-1439 05-02-2010 03:06:43 PM OFICIO 12 070-172060  
2010-07-06-1442 05-02-2010 03:15:53 PM ESCRITURA 123 070-99303  
2010-07-06-1391 05-02-2010 09:43:35 AM ESCRITURA 003 070-6310  
05-02-2010 09:43:35 AM ESCRITURA 003 070-76691  
05-02-2010 09:43:35 AM ESCRITURA 003 070-95003  
2010-07-06-1394 05-02-2010 10:01:23 AM ESCRITURA 143 070-92555  
2010-07-06-1398 05-02-2010 10:14:43 AM ESCRITURA 2010 070-88073  
2010-07-06-1401 05-02-2010 10:24:57 AM ESCRITURA 147 070-168405  
05-02-2010 10:24:57 AM ESCRITURA 147 070-168406  
2010-07-06-1404 05-02-2010 10:35:25 AM 2009-070-6-16590 ESCRITURA 2709 070-62320  
2010-07-06-1405 05-02-2010 10:38:50 AM ESCRITURA 200 070-121077

Turnos Reportados a otro usuario

## AHORA

Turno	Fecha Radicación	Turno Anterior	Documento	Num.	Matriculas
2010-070-6-1431	05-02-2010 02:25.19 PM		ESCRITURA	2299	070-85096
2010-070-6-1433	05-02-2010 02:33.35 PM		ESCRITURA	18	070-179613
2010-070-6-1436	05-02-2010 02:42.23 PM		OFICIO	1792	070-108423
	05-02-2010 02:42.23 PM		OFICIO	1792	070-1849
	05-02-2010 02:42.23 PM		OFICIO	1792	070-19397
	05-02-2010 02:42.23 PM		OFICIO	1792	070-116478
2010-070-6-1439	05-02-2010 03:06.43 PM		OFICIO	12	070-172060
2010-070-6-1442	05-02-2010 03:15.53 PM		ESCRITURA	123	070-99303
<b>Calificador: 1044 FLOR.VARGAS</b>					
Turno	Fecha Radicación	Turno Anterior	Documento	Num.	Matriculas
2010-070-6-1391	05-02-2010 09:43.35 AM		ESCRITURA	003	070-6310
	05-02-2010 09:43.35 AM		ESCRITURA	003	070-95003
	05-02-2010 09:43.35 AM		ESCRITURA	003	070-76691
2010-070-6-1394	05-02-2010 10:01.23 AM		ESCRITURA	143	070-92555
2010-070-6-1398	05-02-2010 10:14.43 AM		ESCRITURA	2010	070-88073
2010-070-6-1401	05-02-2010 10:24.57 AM		ESCRITURA	147	070-168405
	05-02-2010 10:24.57 AM		ESCRITURA	147	070-168406
2010-070-6-1404	05-02-2010 10:35.25 AM	2009-070-6-16590	ESCRITURA	2709	070-62320

1

## 177 Error al segregar con área 100%

### REQUERIMIENTO GLPI 15746 – MANTIS 3877

Turno: Turno 2008-999-6-500

Matricula(s):999-4788

Motivo Inconsistencia: Al segregar el folio y colocar área, estas sumadas dan como resultado 100% pero el sistema indica que no puede superar esa cantidad.

Al segregar un folio y cada segregado se coloca área, estas sumadas dan como resultado 100% pero el sistema arroja error indicando que el porcentaje no puede superar el 100%. Este inconveniente no es muy frecuente pero presenta en algunos casos en el sistema, se hizo una prueba con el turno 2008-999-6-500 donde se puede evidenciar el error. Anexo imágenes.

Por favor escalar este caso a desarrollo para su solución general, por favor crear acta de requerimiento a desarrollo.

Grupo Snr

### DESARROLLO

- Script Bases de Datos:
  - NINGUNA
- Creación De Clases:
  - NINGUNA

CLASE o ARCHIVO JSP:
C:\WORK\SIR-APP\src\gov\sir\core\web\acciones\registro
Método:
AWSegregacion.java
<b>Código Línea 1396:</b>
<pre>double porcentajeTotal = 0; while (itFoliosD.hasNext()) {     try {</pre>

```

        double numPorFolio = 0;
        try {
            numPorFolio = Double.valueOf(((WebFolioDerivado)
itFoliosD.next()).getPorcentaje().replaceAll("%", "").trim()).doubleValue();
        } catch (Exception ee) {
            numPorFolio = 0;
        }
        porcentajeTotal += numPorFolio;
    } catch (NumberFormatException e) {
        porcentajeTotal += 0;
    }
}

String lote = request.getParameter(CFolioDerivado.NOMBRE_LOTE);
String area = request.getParameter(CFolioDerivado.AREA);
String porcen = request.getParameter(CFolioDerivado.PORCENTAJE);
//String unidad = request.getParameter(CFolioDerivado.UNIDAD_MEDIDA);

if(area == null || area.length() <= 0) {
    exception.addError("Debe ingresar un área para el folio derivado");
}

if(lote == null || lote.length() <= 0) {
    exception.addError("Debe ingresar el nombre del inmueble para el folio
derivado");
}

porcen = (porcen.replaceAll("%", "")).trim();
double porcentaje = 0;

if(porcen != null) {
    if(porcen.length() > 0 || !porcen.equals("")) {
        try {
            porcentaje = Double.parseDouble(porcen);
            if(porcentaje <= 0) {
                exception.addError("El porcentaje no puede
ser negativo o cero");
            }
        } catch (NumberFormatException e) {
            exception.addError("El porcentaje es inválido");
        }
        porcen += "%";
    }
}

if(porcentajeTotal + porcentaje > 100) {
    exception.addError("El porcentaje no puede superar 100%");
}

```

**Por:**

```

/*
 * @author: Henry Gómez Rocha
 * @change: Modificación del tipo de dato a las variables porcentajeTotal y porcentaje de tipo double
a tipo BigDecimal, para
 * controlar inexactitud ocasionada por usar tipo double al almacenar la división de los lotes al
momento de separar.
 ** MANTIS: 0003877
 */
BigDecimal porcentajeTotal = new BigDecimal("0.0");
while (itFoliosD.hasNext()) {
    try {
        BigDecimal numPorFolio = new BigDecimal("0.0");
        try {
            numPorFolio = new BigDecimal(((WebFolioDerivado)
itFoliosD.next()).getPorcentaje().replaceAll("%", "").trim());
        } catch (Exception ee) {
            numPorFolio = new BigDecimal("0.0");
        }
        porcentajeTotal = porcentajeTotal.add(numPorFolio);
    } catch (NumberFormatException e) {
        porcentajeTotal = porcentajeTotal.add(new BigDecimal("0.0"));
    }
}


```

```

String lote = request.getParameter(CFolioDerivado.NOMBRE_LOTE);
String area = request.getParameter(CFolioDerivado.AREA);
String porcen = request.getParameter(CFolioDerivado.PORCENTAJE);
//String unidad = request.getParameter(CFolioDerivado.UNIDAD_MEDIDA);

if(area == null || area.length() <= 0) {
    exception.addError("Debe ingresar un área para el folio derivado");
}

if(lote == null || lote.length() <= 0) {
    exception.addError("Debe ingresar el nombre del inmueble para el folio
derivado");
}

porcen = (porcen.replaceAll("%", "")).trim();
BigDecimal porcentaje = new BigDecimal("0.0");

if(porcen != null) {
    if(porcen.length() > 0 || !porcen.equals("")) {
        try {
            porcentaje = new BigDecimal(porcen);
            if(porcentaje.doubleValue() <= 0) {
                exception.addError("El porcentaje no puede
ser negativo o cero");
            }
        } catch (NumberFormatException e) {
            exception.addError("El porcentaje es inválido");
        }
        porcen += "%";
    }
}

porcentajeTotal = porcentajeTotal.add(porcentaje);
if(porcentajeTotal.doubleValue() > 100) {
    exception.addError("El porcentaje no puede superar 100%");
}

```

#### CLASE o ARCHIVO JSP:

C:\WORK\SIR-APP\src\gov\sir\core\web\acciones\registro

Método:

AWCalificacion.java

#### Código Línea 1396:

```

double porcentajeTotal = 0;
while(itFoliosD.hasNext()){
    try {
        porcentajeTotal += Double.valueOf(((FolioDerivado)itFoliosD.next()).getPorcentaje()).doubleValue();
    }
    catch (NumberFormatException e) {
        porcentajeTotal += 0;
    }
}

String descripcion = request.getParameter(CFolioDerivado.DESCRIPCION);
String area = request.getParameter(CFolioDerivado.AREA);
String porcen = request.getParameter(CFolioDerivado.PORCENTAJE);
String unidad = request.getParameter(CFolioDerivado.UNIDAD_MEDIDA);

if(area == null || area.length()<=0){
    exception.addError("Debe ingresar una area para el folio derivado");
}

if(descripcion == null || descripcion.length()<=0){
    exception.addError("Debe ingresar una descripción para el folio derivado");
}

if(unidad.equals(WebKeys.SIN_SELECCIONAR)){
    exception.addError("Debe seleccionar una unidad de medida válida para el folio
derivado");
}

Iterator itUnidades =
((List)session.getServletContext().getAttribute(WebKeys.LISTA_UNIDADES_MEDIDA)).iterator();

```

```

        while(itUnidades.hasNext()){
            OPLookupCodes code = (OPLookupCodes)itUnidades.next();
            if(code.getCodigo().equals(unidad)){
                unidad = code.getValor();
            }
        }

        porcen = (porcen.replaceAll("%","")).trim();
        double porcentaje = 0;

        if ( porcen != null ) {
            if ( porcen.length()>0 || !porcen.equals("") ){
                try {
                    porcentaje = Double.parseDouble(porcen);
                    if (porcentaje <= 0) {
                        exception.addError("El porcentaje no puede ser
negativo o cero");
                    }
                } catch (NumberFormatException e) {
                    exception.addError("El porcentaje es inválido");
                }
            }
        }

        if(porcentajeTotal + porcentaje > 100){
            exception.addError("El porcentaje no puede superar 100%");
        }
    }

Por:
/*
 * @author: Henry Gómez Rocha
 * @change: Modificación del tipo de dato a las variables porcentajeTotal y porcentaje de tipo double
a tipo BigDecimal, para
 * controlar inexactitud ocasionada por usar tipo double al almacenar la división de los lotes al
momento de separar.
** MANTIS: 0003877
*/
BigDecimal porcentajeTotal = new BigDecimal("0.0");
while(itFoliosD.hasNext()){
try {
    //porcentajeTotal +=
    Double.valueOf(((FolioDerivado)itFoliosD.next()).getPorcentaje()).doubleValue();
    porcentajeTotal = porcentajeTotal.add(new
    BigDecimal(((FolioDerivado)itFoliosD.next()).getPorcentaje().toString()));
}
catch (NumberFormatException e) {
    //porcentajeTotal += 0;
    porcentajeTotal = porcentajeTotal.add(new BigDecimal("0.0"));
}
}

String descripcion = request.getParameter(CFolioDerivado.DESCRIPCION);
String area = request.getParameter(CFolioDerivado.AREA);
String porcen = request.getParameter(CFolioDerivado.PORCENTAJE);
String unidad = request.getParameter(CFolioDerivado.UNIDAD_MEDIDA);

if(area == null || area.length()<=0){
    exception.addError("Debe ingresar una área para el folio derivado");
}

if(descripcion == null || descripcion.length()<=0){
    exception.addError("Debe ingresar una descripción para el folio derivado");
}

if(unidad.equals(WebKeys.SIN_SELECCIONAR)){
    exception.addError("Debe seleccionar una unidad de medida válida para el
folio derivado");
}

Iterator itUnidades =
((List)session.getServletContext().getAttribute(WebKeys.LISTA_UNIDADES_MEDIDA)).iterator();

while(itUnidades.hasNext()){
    OPLookupCodes code = (OPLookupCodes)itUnidades.next();
```

```

        if(code.getCodigo().equals(unidad)){
            unidad = code.getValor();
        }

        porcen = (porcen.replaceAll("%","")).trim();
        //double porcentaje = 0;
        BigDecimal porcentaje = new BigDecimal("0.0");

        if( porcen != null ) {
            if( porcen.length()>0 || !porcen.equals("") ){
                try {
                    //porcentaje = Double.parseDouble(porcen);
                    porcentaje = new BigDecimal(porcen);
                    if(porcentaje.doubleValue() <= 0) {
                        exception.addError("El porcentaje no puede
ser negativo o cero");
                    }
                } catch (NumberFormatException e) {
                    exception.addError("El porcentaje es inválido");
                }
            }
        }

        porcentajeTotal = porcentajeTotal.add(porcentaje);
        //if (Redondear(porcentajeTotal + porcentaje, 5) > 100) {
        if(porcentajeTotal.doubleValue() > 100) {
            exception.addError("El porcentaje no puede superar 100%");
        }
    }
}

```

## 181 Diferencia de dinero en el reporte 51 (Informativo)

### **REQUERIMIENTO**

Motivo Inconsistencia: Se comunica la Sra. Esperanza Ibarra informando que en el reporte 51 se nota una diferencia de \$-17395.

Por favor colaborarme en revisar el reporte.

Grupo Popayán

### **DESARROLLO**

El reporte se estaba generando con valores incorrectos ya que el valor de la consignación número 61191695 tenía registrado el valor del documento en 0.

Se corrige esta información y se genera el reporte correctamente.

Por favor informar al usuario que ya puede generar el reporte por la aplicación.

## 184 Turnos de correcciones enviados a mayor valor

### **REQUERIMIENTO GLPI 15912**

Turnos de correcciones enviados a mayor valor. Motivo Inconsistencia: Cuando un turno de correcciones es enviado a mayor valor, este avanza a la fase custodia mayor valor, al tratar avanzar el turno de esta fase el sistema deja la pantalla en blanco y no permite realizar el trámite del turno.

Por favor escalar este caso a desarrollo para su solución general ya que el sistema no está permitiendo el avance normal del turno para continuar con el proceso de mayor valor. Por favor crear acta de requerimiento formal a desarrollo.

## **DESARROLLO**

Análisis:

Al momento de dar un mayor valor desde el turno correcciones el aplicativo estaba tratando de asignar valores al turno, razón por la cual el sistema no presentaba ninguna información ya que desde correcciones no es posible adicionar valor.

Se creó una variable de tipo Liquidación la cual verifica desde donde fue generado el Turno de Registro, esto se hace con el fin de no adicionar valores cuando el turno es generado desde el Rol Cajero Correcciones.

➤ Script Bases de Datos:

- NINGUNA (en caso de script de modificación colocar en este lugar el nombre del archivo que se adjunta a la aplicación mantis)

➤ Creación De Clases:

- NINGUNA (el mismo formato de modificación clases o jsp)

➤ Se modificaron los siguientes archivos:

CLASE o ARCHIVO JSP:
<b>SIR-APP\web\WEB-INF\jspSeguro\pagomayorvalor \CON_CUSTODIA_PAGOS_MAYORES.JSP</b>
Método:
Nombre Metodo
<b>Código Línea 33:</b> <i>LiquidacionTurnoRegistro objLiquidacionTurnoRegistro = LiquidacionTurnoRegistro)iteraSolicitudLiquidacion.next(); if(objLiquidacionTurnoRegistro instanceof LiquidacionTurnoRegistro){</i>
<b>Por:</b> <i>Liquidacion obj = (Liquidacion)iteraSolicitudLiquidacion.next(); if(obj instanceof LiquidacionTurnoRegistro){ LiquidacionTurnoRegistro objLiquidacionTurnoRegistro = (LiquidacionTurnoRegistro)obj;</i>

## **186 Fechas de radicación y documento turnos de correcciones**

### **REQUERIMIENTO GLPI 16059**

En un turno de correcciones cuando se va a realizar alguna corrección sobre anotaciones en cuanto a la fecha de radicación y documento y esta anotación es temporal, el sistema no guarda los cambios ni notifica error.

Por favor escalar este caso a desarrollo para su solución general. Por favor crear acta de requerimiento formal a desarrollo.

## **DESARROLLO**

➤ Script Bases de Datos:

- NINGUNA

➤ Creación De Clases:

- NINGUNA

➤ Se modificaron los siguientes archivos:

CLASE:
<b>gov.sir.forseti.dao.impl.jdogenie.JDOGenieFolioDAO.java</b>
Método:
<i>protected boolean actualizarAnotacionTMP(FolioEnhanced folio, AnotacionTMP toUpdate, AnotacionTMP datos, UsuarioEnhanced usuario,</i>

```
boolean soloEliminarContraparteTemporal, TurnoEnhanced turno, PersistenceManager pm) throws DAOException
```

**Código Insertado Linea 19919:**

```
/* JAlcaza caso Mantis 03996 Acta - Requerimiento No 186 - Fechas de radicación y documento turnos de correcciones
 * No se permitía la actualización de fechas en anotaciones temporales
 */
if (datos.getFechaRadicacion() != null) {
    toUpdate.setFechaRadicacion(datos.getFechaRadicacion());
}
```

## 188 No aparece la nota Devolutiva

### REQUERIMIENTO No 16086

Turno:2009-350-6-23230 - Matricula(s):350-81665

Motivo Inconsistencia: Informa la funcionaria Venovit Mejía que al verificar el turno 2009-350-6-23230 no aparece la nota devolutiva.

El turno 2009-350-6-23230 se trabajó en la oficina y quedó como devuelto, al verificar la nota devolutiva no aparece y el turno ya está como desanotado correspondencia.  
Por favor verificar por que la nota devolutiva no aparece al verificar el turno ya que los datos se encuentran como definitivos.

grupo Ibagué

### DESARROLLO

- Script Bases de Datos:
  - NINGUNA
- Creación De Clases:
  - NINGUNA
- Se modificaron los siguientes archivos:

#### CLASE:

```
gov.sir.core.web.helpers.comun.ListarNotasDevolutivasHelper.java
```

#### Método:

```
public void setProperties(HttpServletRequest request) throws HelperException
```

**Código Linea 93:**

```
if(turno.getIdFase().equals("REG_CERTIFICADOS_ASOCIADOS") ||
turno.getIdFase().equals("REG_ENTREGA") || turno.getIdFase().equals("REG_ENTREGA_EXTERNO") ||
turno.getIdFase().equals("FINALIZADO")|| turno.getIdFase().equals("FINALIZADO-DEVOL")){
temporales = 0;
listaAuxNotas.add(notas);
}
```

**POR**

/\*JAlcazar caso Mantis 03992 Requerimiento No 188 - No aparece la nota devolutiva  
\* se modifco la cadena REG\_ENTREGA\_EXTERNOS la cual no correspondía a la fase en la BD  
\* por REG\_ENTREGA\_EXTERNO y no permitía visualizar las notas devolutivas de los turnos en esta fase.  
\*/

```
if(turno.getIdFase().equals("REG_CERTIFICADOS_ASOCIADOS") ||
turno.getIdFase().equals("REG_ENTREGA") || turno.getIdFase().equals("REG_ENTREGA_EXTERNO") ||
turno.getIdFase().equals("FINALIZADO")|| turno.getIdFase().equals("FINALIZADO-DEVOL")){
temporales = 0;
listaAuxNotas.add(notas);
}
```

## 189 Reporte 9 (Informativo)

### REQUERIMIENTO Glpi 16098

Turno:2010-280-6-3196 - Matricula(s):280-160955

Motivo Inconsistencia: Informa Judith Naranjo que al generar el Reporte 9: Diario radicador de documentos del día y el Reporte 99: Ingresos diarios por documentos del día 22/02/2010, pero hay una diferencia de 14,280 cuyo valor corresponde al turno 2010-280-6-3196, el cual fue ingresado como una nueva entrada.

Se genera el Reporte 9 - Diario radicador de documentos del día con los siguientes parámetros:

Fecha Radicación: 22/02/2010

No. de Página: 1

Círculo Registral : Armenia

En Total Derechos se genera un valor de \$ 9.800.480.

Se genera el Reporte 99; Ingresos diarios por documentos con los siguientes parámetros:

Fecha Inicial: 22/02/2010

Fecha Final: 22/02/2010

Círculo Registral : Armenia

En total recaudo se genera un valor de \$ 9.786.200

Al consultarlos actos asociados del turno 2010-280-6-3196 vemos que tiene asociados los actos:

CANCELACION PATRIMONIO DE FAMILIA Valor: 14,280

VENTA Valor: 14,280

CANCELACION AFECTACION FAMILIAR Valor: 14,280

El turno anterior solo tiene asociados actos y cuando radicaron la nueva entrada radicaron un nuevo acto por valor de 14.280. Favor verificar los datos asociados al turno de nueva entrada.

Grupo Armenia

### ***DESARROLLO***

**ANÁLISIS:** La inconsistencia se presenta en el reporte-99, se compara con los reportes 9, 25 y 51 y es el único que presenta un valor diferente en el recaudo de documentos de registro.

Esta inconsistencia se está presentando al realizar operaciones con la variable v\_valor cuando esta tiene un valor null.

Se inicializa en 0 la variable v\_valor de la Función fnc\_valores\_acto.

Código:

```
v_valor NUMBER(20,3):=0;
```

Adjunto el pdf del REPORTE-99 donde se observa el valor 9,800,480, que coincide con el valor del reporte-9.

INCIDENCIAS RESUELTAS
<b>025 Matrices y Segregados (Informativo)</b>
<b>REQUERIMIENTO</b>
Matrices y Segregados. SIR-194
Se han presentado casos en los cuales al corregir o incluir datos en los campos de matrices y segregados, los cambios se realizan y se visualizan en el proceso del turno de corrección, pero al finalizarlo y consultar los folios corregidos, no se ven los cambios efectuados. (DR E - 1004).
<b>SEGUIMIENTO ADA:</b>
Favor tener unos folios ejemplo o turnos donde podamos apreciar el problema.. no se ha podido reproducir en desarrollo
<b>OBSERVACIONES SNR:</b>
Verificar en GLPI si esta inconsistencia persiste con base a los casos reportados o si han sido casos puntuales.
<b>DESARROLLO</b>
<ul style="list-style-type: none"> <li>➤ Script Bases de Datos: NINGUNO</li> <li>➤ Creación De Clases: NINGUNO</li> <li>➤ Se modificaron los siguientes archivos: NINGUNO</li> </ul>
En el formato del Requerimiento 025_141_Matrices y Segregados.doc se encuentra la siguiente observación.
<b>OBSERVACIONES SNR:</b>
<b>Verificar en GLPI si esta inconsistencia persiste con base a los casos reportados o si han sido casos puntuales.</b>
El problema al que hace referencia el requerimiento, no se está produciendo en la actualidad en el Sistema. Esta aseveración es producto de la revisión del proceso y de la revisión de los casos reportados en GLPI, el cual reflejo que las modificaciones temporales en el proceso de correcciones una vez aprobadas pasan a estado definitivo y pueden ser observadas en el folio.
Basado en lo anterior se procede a cerrar el caso.
054 Creacion de un esquema de mensajeria
<b>REQUERIMIENTO</b>
En el sistema SIR se debe crear un esquema de mensajería sobre el cual se soportara la integración con el SGD, para lo cual SIR deberá publicar un mensaje con la

información que se requiere para el SGD con los siguientes datos:

5- Para los turnos radicados, los siguientes campos:

- NumeroTurno Varchar(20); Número de Turno
- FechaRadicacion Date; Fecha de radicación (Fecha y hora)
- OficinaOrigen Varchar(50); Oficina origen
- Instrumento Varchar(2); Instrumento (Tipo de documento a registrar)
- NumeroDocumento Varchar(10); Número del documento a registrar
- FechaDocumento Date; Fecha del documento a registrar (Fecha y hora)
- Ciudad Varchar(50); Ciudad de origen del documento
- NumeroRecibo Numeric(10); Número del recibo asignado al pago
- Usuario Varchar(15); Usuario (Identificador del sistema que realiza la radicación)
- Solicitante Varchar(60); Solicitante (Nombre de la persona natural o jurídica que solicita el registro)
- RadicadoAnterior Varchar(20); Radicado anterior (Número de radicación de documento al cual se le ha generado una nueva entrada)
- TurnoAsociado Varchar(20); Turno Asociado (Número del turno del cual proviene el nuevo turno)
- TipoMensaje Integer; Tipo de mensaje (Constante con valor 1)

Consideraciones

Se debe publicar un mensaje con la estructura definida, cada vez que se cree un nuevo turno en SIR, para los procesos correspondientes a Registro, Correcciones, Certificados de Antiguo Sistema, Ampliación de Tradición y Pertenencia, Consultas Masivas y Reparto Notarial.

En caso de presentarse modificaciones en la información de los turnos creados en SIR con respecto a los campos definidos en el mensaje, se deberá publicar un mensaje con la estructura definida.

Para aquellos nuevos turnos que son generados como resultado de un sub- proceso de otro turno, por ejemplo las correcciones internas, en el campo “Turno Asociado” se deberá indicar el número de turno del cual proviene el nuevo turno.

Es de tener en cuenta que para aquellos turnos que se les modifique información del encabezado tales como tipo de documento, numero de documento, fecha del documento, oficina origen y ciudad, se debe enviar nuevamente la información actualizada del turno

6- Para matriculas asociadas:

- NumeroTurno Varchar(20); Número de Turno
- NumeroMatricula Numeric(10); Número de matrícula (Asociada al número de radicación)
- Accion Integer; Acción (1. Adicionar, 2. Modificar y 3. Eliminación)

- |  |  |
|--|--|
| <ul style="list-style-type: none"> <li>• FechaEvento Date; Fecha (Fecha y hora en la que se ejecuta la acción)</li> <li>• TipoMensaje Integer; Tipo de mensaje (Constante con valor 2)</li> <li>• NUmeroCirculo numeric(3)</li> </ul> <p>Consideraciones</p> <p>En éste esquema SIR debe publicar el mensaje con la información de todos los números de matriculas que están asociados a un numero de turno en particular, indicando la acción que se esta llevando a cabo (Adición, Modificación ó Eliminación)</p> <p>3.Turnos Anulados.</p> <ul style="list-style-type: none"> <li>• NumeroTurno Varchar(20); Número de Turno</li> <li>• FechaAnulacion Date; Fecha y hora en que se efectúo la anulación</li> <li>• TipoMensaje Integer; Tipo de mensaje (Constante con valor 4)</li> </ul> <p>Consideraciones</p> <p>En ésta esquema SIR deberá registrar la información de aquellos turnos que se han anulado en el sistema.</p> <p>Para lograr la integración entre SIR y el SGD, se implementará un sistema de mensajería a través de JMS (Servicio de Mensajería de Java). En el sistema SIR se debe implementar un publicador de mensajes, el cual se encargará de enviar el mensaje con la información al tópico el cual es administrado por JMS y quien se encargara de enviar los mensajes a los suscriptores (SGD). La publicación de la información debe de estar de acuerdo con lo expresado en la parte del esquema de datos.</p> <p>La ETB deberá configurar el sistema de mensajería e indicar a Colgrabar cual será el sistema seleccionado para la administración de la mensajería. Una vez se haya implementado la integración entre SIR y el SGD, la administración del sistema de mensajería estará a cargo de la ETB.</p> <p>Es de anotar que la ETB solamente deberá implementar lo relacionado con la publicación de los mensajes sobre el tópico ya que la lógica del componente para el envío de los mensajes a los suscriptores estará a cargo de Colgrabar, la ETB deberá facilitar el acceso al ambiente de pruebas y de la publicación de componente para el envío de los mensajes para realizar las pruebas de integración.</p> <p><b>DESARROLLO</b></p> <p>Análisis: Se debe crear una aplicación que publique estos mensajes.</p> <p>Solución: Se creó la aplicación Gestión Documental, esta aplicación se encarga de hacer la publicación de estos mensajes.</p> |  |
|--|--|

### 161 Reporte 25 y 51 Turnos anulados Masivos

#### **REQUERIMIENTO**

En los reporte 25 y 51 se observa que cuando se anula un certificado individual asociado a un turno de certificado masivo y en el mismo día se anula un turno de registro, el sistema no descuenta el valor de los certificados individuales anulados del

valor total del Masivo es decir que mantiene el valor total de la radicación.

## **DESARROLLO**

Reporte 25. (Informe Recaudos del día por usuario)

- Script Bases de Datos:

### 1. Se creó la función VAL\_MASIVO

```
create or replace
FUNCTION VAL_MASIVO (p_idSolicitudA IN VARCHAR) RETURN NUMBER IS
    v_liquidacion NUMBER;
BEGIN
/** AHERRENO 05/02/2010
ESTA FUNCION TIENE COMO OBJETIVO CALCULAR EL TOTAL DE LOS CERTIFICADOS
ASOCIADOS A UN MASIVO QUE NO SE ENCUENTREN ANULADOS */
    v_liquidacion := 0;
    SELECT SUM(L.lqdc_valor) v_liquidacion INTO v_liquidacion
    FROM sir_op_liquidacion L
        ,sir_op_sol_asociada SA
        ,sir_op_turno T
    WHERE L.id_solicitud = sa.id_solicitud1
    AND t.id_solicitud = sa.id_solicitud1
    AND SA.id_solicitud = p_idSolicitudA
    AND T.trno_anulado = 'N';

    RETURN v_liquidacion;
END;
```

Reporte 51 (Resumen diario Ingresos y Egresos)

### 2. Se modifico el procedimiento del reporte 51 (Resumen diario Ingresos y Egresos) SP\_REPORTE\_51 en la cual se hace el llamado de la función VAL\_MASIVO para que en el valor del proceso de Masivos (20) no se tenga en cuenta el valor de certificados asociados anulados.

```
CURSOR c_liq(p_sol VARCHAR2)
IS
    SELECT id_liquidacion
    ,id_solicitud
/** AHERRENO 04/02/2010
SE AGREGA CONDICION, EN LA CUAL VALIDA SI EL PROCESO DEL TURNO ES 20
SE EJECUTA LA FUNCION VAL_MASIVO QUE PIDE COMO PARAMETRO LA SOLICITUD
ESTA FUNCION TIENE COMO OBJETIVO CALCULAR EL TOTAL DE LOS CERTIFICADOS
ASOCIADOS A UN MASIVO QUE NO SE ENCUENTREN ANULADOS */
    ,DECODE(id_proceso, 20, VAL_MASIVO(id_solicitud),lqdc_valor) lqdc_valor
    ****
    ,id_tipo_certificado
    ,lqdc_valor_otro_impuesto_rg
    ,lqdc_otro_impuesto_rg
    ,id_proceso
    ,lqdc_fecha_creacion
    ,id alcance_geografico
    ,id_tipo_fotocopia
    ,lqdc_num_hojas_ft
    ,lqdc_num_copias_ft
    ,lqdc_last_id_acto_rg
    ,lqdc_valor_derechos_rg
    ,lqdc_valor_imuestos_rg
    ,lqdc_valor_mora_rg
    ,id_usuario
    ,lqdc_fecha_cancelacion
    ,lqdc_fecha_liquidacion
    ,lqdc_cancelada
    ,lqdc_tipo_tarifa_cr
    ,lqdc_justificacion_mayor_valor
    ,lqdc_uid
    ,lqdc_usuario_ant
    ,lqdc_recibo_ant
```

```

, lqdc_nrorad_ant
, id_circulo
, lqdc_valor_mv
, lqdc_valor_derechos_rg_mv
, lqdc_valor_impuestos_rg_mv
, lqdc_valor_saldo_favor
, id_nliquidacion
FROM SIR_OP_LIQUIDACION
WHERE id_solicitud = p_sol
AND id_circulo = p_circulo;
CURSOR c_app(p_sol VARCHAR2, p_liq VARCHAR2)
IS
SELECT id_documento_pago,
/** AHERRENO 04/02/2010
SE AGREGA CONDICION, EN LA CUAL VALIDA SI EL PROCESO DEL TURNO ES 20
SE EJECUTA LA FUNCION VAL_MASIVO QUE PIDE COMO PARAMETRO LA SOLICITUD
ESTA FUNCION TIENE COMO OBJETIVO CALCULAR EL TOTAL DE LOS CERTIFICADOS
ASOCIADOS A UN MASIVO QUE NO SE ENCUENTREN ANULADOS */
DECODE(substr(p_sol, 10, 2), '20', VAL_MASIVO(p_sol),plpg_valor_aplicado) plpg_valor_aplicado,
/*****************/
id_liquidacion
FROM SIR_OP_APPLICACION_PAGO
WHERE id_solicitud = p_sol
AND id_liquidacion = p_liq;

```

➤ Creación De Clases:

NINGUNO

➤ Modificación de archivos:

Se realizaron las siguientes modificaciones en los reportes 25 (Informe Recaudos del día por usuario) en la cual se hace el llamado de la función VAL\_MASIVO para que en el valor del proceso de Masivos (20) no se tenga en cuenta el valor de certificados asociados anulados..

reporte-25.jasper

```

/** AHERRENO 04/02/2010
SE AGREGA CONDICION PARA CUANDO EL TURNO SEA DEL PROCESO 20
EJECUTE LA FUNCION VAL_MASIVO/
DECODE(sir_op_proceso.id_proceso, 20, VAL_MASIVO(sir_op_solicitud.id_solicitud), NVL(sir_op_liquidacion.lqdc_valor,0 ))
liquidacion_valor,
/*****************/
NVL(sir_op_liquidacion.lqdc_valor_impuestos_rg,0) liquidacion_x_impuestos,
NVL(sir_op_liquidacion.lqdc_valor_otro_impuesto_rg,0) liquidacion_x_otrosimpuestos,
NVL(sir_op_liquidacion.lqdc_valor_mora_rg,0) liquidacion_x_mora,
NVL(sir_op_liquidacion.lqdc_valor_derechos_rg,0) liquidacion_x_derechos,
sir_op_pago.pago_fecha_pago_fecha,
sir_op_proceso.id_proceso,
fnc_IdBancosNumDocs_Solicitud(sir_op_solicitud.id_solicitud) bancos_num_docpago,
sir_op_usuario.srio_nombre_de_usuario usuario_nombre,
sir_op_solicitud.id_solicitud solicitud_hija,
sir_op_solicitud.slct_fecha_solicitud fecha_solicitud,
sir_op_sol_asociada.id_solicitud sol_padre,
NVL(SP_ACUMULADO_DOCUMENTO_PAGO(sir_op_solicitud.id_solicitud),0) saldo_documento,
( NVL(sir_op_liquidacion.lqdc_valor_impuestos_rg,0) + NVL(sir_op_liquidacion.lqdc_valor_otro_impuesto_rg,0) + NVL
(sir_op_liquidacion.lqdc_valor_derechos_rg,0) + NVL(sir_op_liquidacion.lqdc_valor_mora_rg,0) ) liquidacion_x_sum,
/*****************/
NVL(
CASE
WHEN sir_op_aplicacion_pago.id_liquidacion <=1
THEN (
CASE
WHEN sir_op_proceso.id_proceso=4
THEN
(SELECT SUM(decode(DP.id_tipo_doc_pago, 1,sir_op_liquidacion.lqdc_valor, 6,sir_op_liquidacion.lqdc_valor,
7,sir_op_liquidacion.lqdc_valor, null))
FROM SIR_OP_APPLICACION_PAGO AP,

```

```

    sir_op_documento_pago DP
    WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
    AND AP.id_documento_pago = DP.id_documento_pago
    AND DP.id_tipo_doc_pago IN (1, 6, 7)
    AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion )
ELSE
/** AHERRENO 04/02/2010
SE AGREGA CONDICION PARA CUANDO EL TURNO SEA DEL PROCESO 20
EJECUTE LA FUNCION VAL_MASIVO*/
( SELECT DECODE(sir_op_proceso.id_proceso, 20, VAL_MASIVO(sir_op_solicitud.id_solicitud), sum( decode
(DP.id_tipo_doc_pago, 1,sir_op_liquidacion.lqdc_valor, 6,sir_op_liquidacion.lqdc_valor, 7,sir_op_liquidacion.lqdc_valor, null)))
FROM SIR_OP_APPLICACION_PAGO AP,
    sir_op_documento_pago DP
    WHERE AP.ID_SOLICITUD =sir_op_solicitud.id_solicitud
    AND AP.id_documento_pago = DP.id_documento_pago
    AND DP.id_tipo_doc_pago IN (1, 6, 7)
    AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion )
END )
ELSE
CASE
WHEN sir_op_proceso.id_proceso=4
THEN
( SELECT DECODE (DP.id_tipo_doc_pago, 1, sir_op_liquidacion.lqdc_valor *1, 6, sir_op_liquidacion.lqdc_valor *1, 7,
sir_op_liquidacion.lqdc_valor *1, NULL)
    FROM SIR_OP_APPLICACION_PAGO AP,
        sir_op_documento_pago DP
    WHERE AP.ID_SOLICITUD =sir_op_solicitud.id_solicitud
    AND AP.id_documento_pago = DP.id_documento_pago
    AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
    AND DP.id_tipo_doc_pago IN (1, 6, 7) )
ELSE
/** AHERRENO 04/02/2010
SE AGREGA CONDICION PARA CUANDO EL TURNO SEA DEL PROCESO 20
EJECUTE LA FUNCION VAL_MASIVO*/
( SELECT      DECODE      (DP.id_tipo_doc_pago,      1,      DECODE(sir_op_proceso.id_proceso,      20,
VAL_MASIVO(sir_op_solicitud.id_solicitud),sir_op_liquidacion.lqdc_valor) , 6, DECODE(sir_op_proceso.id_proceso, 20,
VAL_MASIVO(sir_op_solicitud.id_solicitud),sir_op_liquidacion.lqdc_valor) , 7, DECODE(sir_op_proceso.id_proceso, 20,
VAL_MASIVO(sir_op_solicitud.id_solicitud),sir_op_liquidacion.lqdc_valor) ,NULL )
    FROM SIR_OP_APPLICACION_PAGO AP,
        sir_op_documento_pago DP
    WHERE AP.ID_SOLICITUD =sir_op_solicitud.id_solicitud
    AND AP.id_documento_pago = DP.id_documento_pago
    AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
    AND DP.id_tipo_doc_pago IN (1, 6, 7) )
END
END ,0)vr_aplicado_efectivo,
*****
NVL(
CASE
WHEN sir_op_aplicacion_pago.id_liquidacion<=1
THEN
( SELECT SUM(AP.plpg_valor_aplicado)--SUM(sir_op_liquidacion.lqdc_valor)--AP.plpg_valor_aplicado
FROM SIR_OP_APPLICACION_PAGO AP,
    sir_op_documento_pago DP
    WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
    AND AP.id_documento_pago = DP.id_documento_pago
    and AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
    AND DP.id_tipo_doc_pago = 2
    AND ap.id_liquidacion<=1
    AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N' ) )
ELSE
( SELECT SUM(AP.plpg_valor_aplicado) --SUM(sir_op_liquidacion.lqdc_valor)--AP.plpg_valor_aplicado
FROM SIR_OP_APPLICACION_PAGO AP,
    sir_op_documento_pago DP
    WHERE ap.ID_SOLICITUD = sir_op_solicitud.id_solicitud
    AND AP.id_documento_pago = DP.id_documento_pago
    AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
    AND DP.id_tipo_doc_pago = 2
    AND ap.id_liquidacion>1
    AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N' ) )
END ,0)vr_aplicado_consignacion,
*****
NVL(
CASE
WHEN sir_op_aplicacion_pago.id_liquidacion<=1

```

```

THEN
  (SELECT SUM( decode (DP.id_tipo_doc_pago, 3, sir_op_liquidacion.lqdc_valor, 4, sir_op_liquidacion.lqdc_valor, 5,
sir_op_liquidacion.lqdc_valor, 10, sir_op_liquidacion.lqdc_valor, null) )
FROM SIR_OP_APPLICACION_PAGO AP,
  sir_op_documento_pago DP
WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
AND AP.id_documento_pago = DP.id_documento_pago
AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
AND DP.id_tipo_doc_pago IN (3, 4, 5, 10)
AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N' ) )
ELSE
  (SELECT DECODE (DP.id_tipo_doc_pago, 3, sir_op_liquidacion.lqdc_valor, 4, sir_op_liquidacion.lqdc_valor, 5,
sir_op_liquidacion.lqdc_valor, 10, sir_op_liquidacion.lqdc_valor, null )
FROM SIR_OP_APPLICACION_PAGO AP,
  sir_op_documento_pago DP
WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
AND AP.id_documento_pago = DP.id_documento_pago
AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
AND DP.id_tipo_doc_pago IN (3, 4, 5, 10)
AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N' ))
END ,0)vr_aplicado_cheque,
/*********************************************************************
NVL(
(SELECT SUM(sir_op_liquidacion.lqdc_valor)
FROM SIR_OP_APPLICACION_PAGO AP,
  sir_op_documento_pago DP
WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
AND AP.id_documento_pago = DP.id_documento_pago
AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
AND DP.id_tipo_doc_pago = 2
AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N')
),0) vr_aplicado_consignacion_na,
/*********************************************************************
NVL(
(SELECT DECODE (DP.id_tipo_doc_pago, 3, sir_op_liquidacion.lqdc_valor, 4, sir_op_liquidacion.lqdc_valor, 5,
sir_op_liquidacion.lqdc_valor, 10, sir_op_liquidacion.lqdc_valor, null )
FROM SIR_OP_APPLICACION_PAGO AP,
  sir_op_documento_pago DP
WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
AND AP.id_documento_pago = DP.id_documento_pago
AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
AND DP.id_tipo_doc_pago IN (3, 4, 5, 10)
AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N')
),0) vr_aplicado_cheque_na,
/*********************************************************************
DECODE(sir_op_sol_asociada.id_solicitud ,null ,NVL(
CASE
  WHEN sir_op_aplicacion_pago.id_liquidacion <=1
  THEN (
  CASE
    WHEN sir_op_proceso.id_proceso=4
  THEN
    (SELECT SUM( decode (DP.id_tipo_doc_pago, 1,
    CASE
      WHEN DP.dcpq_valor_documento<=0
      THEN sir_op_liquidacion.lqdc_valor
      ELSE DP.dcpq_valor_documento
    END, 6,
    CASE
      WHEN DP.dcpq_valor_documento<=0
      THEN sir_op_liquidacion.lqdc_valor
      ELSE DP.dcpq_valor_documento
    END, 7,
    CASE
      WHEN DP.dcpq_valor_documento<=0
      THEN sir_op_liquidacion.lqdc_valor
      ELSE DP.dcpq_valor_documento
    END, null ) )
  FROM SIR_OP_APPLICACION_PAGO AP,
  sir_op_documento_pago DP
  WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
  AND AP.id_documento_pago=DP.id_documento_pago
  AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
  AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N' ) )
  ELSE

```

```

        (SELECT SUM( decode (DP.id_tipo_doc_pago, 1,
        CASE
            WHEN DP.dcpg_valor_documento<=0
        /** AHERRENO 04/02/2010
        SE AGREGA CONDICION PARA CUANDO EL TURNO SEA DEL PROCESO 20
        EJECUTE LA FUNCION VAL_MASIVO/
            THEN          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),sir_op_liquidacion.lqdc_valor)
            ELSE          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpg_valor_documento)
        END, 6,
        CASE
            WHEN DP.dcpg_valor_documento<=0
        /** AHERRENO 04/02/2010
        SE AGREGA CONDICION PARA CUANDO EL TURNO SEA DEL PROCESO 20
        EJECUTE LA FUNCION VAL_MASIVO/
            THEN          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),sir_op_liquidacion.lqdc_valor)
            ELSE          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpg_valor_documento)
        END, 7,
        CASE
            WHEN DP.dcpg_valor_documento<=0
        /** AHERRENO 04/02/2010
        SE AGREGA CONDICION PARA CUANDO EL TURNO SEA DEL PROCESO 20
        EJECUTE LA FUNCION VAL_MASIVO/
            THEN          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),sir_op_liquidacion.lqdc_valor)
            ELSE          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpg_valor_documento)
        END, null )
        FROM SIR_OP_APPLICACION_PAGO AP,
        sir_op_documento_pago DP
        WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
        AND AP.id_documento_pago= DP.id_documento_pago
        AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
        AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trmo_anulado = 'N' )
        END )
    ELSE
    CASE
        WHEN sir_op_proceso.id_proceso=4
        THEN
            (SELECT DECODE (DP.id_tipo_doc_pago, 1, DP.dcpg_valor_documento, 6, DP.dcpg_valor_documento, 7,
DP.dcpg_valor_documento, NULL )
            FROM SIR_OP_APPLICACION_PAGO AP,
            sir_op_documento_pago DP
            WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
            AND AP.id_documento_pago = DP.id_documento_pago
            AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
            AND DP.id_tipo_doc_pago IN (1, 6, 7)
            AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trmo_anulado = 'N' )
        ELSE
        /** AHERRENO 04/02/2010
        SE AGREGA CONDICION PARA CUANDO EL TURNO SEA DEL PROCESO 20
        EJECUTE LA FUNCION VAL_MASIVO/
            (SELECT      DECODE      (DP.id_tipo_doc_pago,      1,      DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpg_valor_documento),      6,      DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpg_valor_documento),      7,      DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpg_valor_documento),NULL )
            FROM SIR_OP_APPLICACION_PAGO AP,
            sir_op_documento_pago DP
            WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
            AND AP.id_documento_pago = DP.id_documento_pago
            AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
            AND DP.id_tipo_doc_pago IN (1, 6, 7)
            AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trmo_anulado = 'N')
        )
        END
    END,0)vr_docpago_efectivo ,
    ****
    DECODE(sir_op_sol_asociada.id_solicitud ,null , NVL(
    CASE
        WHEN sir_op_aplicacion_pago.id_liquidacion<= 1
        THEN (
        CASE

```

```

WHEN sir_op_proceso.id_proceso=4
THEN
(SELECT SUM( decode (DP.id_tipo_doc_pago, 2,
CASE
WHEN DP.dcpq_valor_documento<=0
THEN AP.plpg_valor_aplicado
ELSE DP.dcpq_valor_documento
END, null) )
FROM SIR_OP_APPLICACION_PAGO AP,
sir_op_documento_pago DP
WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
AND AP.id_documento_pago= DP.id_documento_pago
AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
AND AP.id_liquidacion<= 1
AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N' ) )
ELSE
(SELECT SUM( decode (DP.id_tipo_doc_pago, 2,
CASE
WHEN DP.dcpq_valor_documento<=0
THEN DP.dcpq_valor_documento
ELSE DP.dcpq_valor_documento
END, null) )
FROM SIR_OP_APPLICACION_PAGO AP,
sir_op_documento_pago DP
WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
AND AP.id_documento_pago= DP.id_documento_pago
AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
AND AP.id_liquidacion<= 1
AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N' ) )
ELSE
(SELECT SUM( decode (DP.id_tipo_doc_pago, 2,
CASE
WHEN DP.dcpq_valor_documento<=0
THEN DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),sir_op_liquidacion.lqdc_valor)
ELSE DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpq_valor_documento)
END, null) )
FROM SIR_OP_APPLICACION_PAGO AP,
sir_op_documento_pago DP
WHERE AP.ID_SOLICITUD =sir_op_solicitud.id_solicitud
AND AP.id_documento_pago=DP.id_documento_pago
AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
AND AP.id_liquidacion<= 1
AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N' ) )
END )
ELSE
CASE
WHEN sir_op_proceso.id_proceso=4
THEN
(SELECT SUM(DP.dcpq_valor_documento)
FROM SIR_OP_APPLICACION_PAGO AP,
sir_op_documento_pago DP
WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
AND AP.id_documento_pago = DP.id_documento_pago
AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
AND DP.id_tipo_doc_pago = 2
AND AP.id_liquidacion> 1
AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N' ) )
ELSE
(SELECT SUM(DP.dcpq_valor_documento)
FROM SIR_OP_APPLICACION_PAGO AP,
sir_op_documento_pago DP
WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
AND AP.id_documento_pago = DP.id_documento_pago
AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
AND DP.id_tipo_doc_pago = 2
AND AP.id_liquidacion> 1
AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N' ))
END
END ,0)vr_docpago_consignacion,
/*********************************************
DECODE(sir_op_sol_asociada.id_solicitud ,null ,NVL(
CASE
WHEN sir_op_aplicacion_pago.id_liquidacion<=1
THEN (
CASE
WHEN sir_op_proceso.id_proceso=4
THEN
(SELECT SUM( decode (DP.id_tipo_doc_pago, 3,
CASE
WHEN DP.dcpq_valor_documento<=0
THEN sir_op_liquidacion.lqdc_valor
ELSE DP.dcpq_valor_documento
END, 10,
CASE

```

```

WHEN DP.dcpq_valor_documento<=0
THEN sir_op_liquidacion.lqdc_valor
ELSE DP.dcpq_valor_documento
END, 4,
CASE
WHEN DP.dcpq_valor_documento<=0
THEN sir_op_liquidacion.lqdc_valor
ELSE DP.dcpq_valor_documento
END, 5,
CASE
when DP.dcpq_valor_documento<=0
THEN sir_op_liquidacion.lqdc_valor
ELSE DP.dcpq_valor_documento
END, null )
FROM SIR_OP_APPLICACION_PAGO AP,
    sir_op_documento_pago DP
WHERE AP.ID_SOLICITUD =sir_op_solicitud.id_solicitud
AND AP.id_documento_pago=DP.id_documento_pago
AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
AND AP.id_liquidacion<=1
AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trmo_anulado = 'N'))
ELSE
( SELECT SUM( decode (DP.id_tipo_doc_pago, 3,
CASE
WHEN DP.dcpq_valor_documento<=0
/** AHERRENO 04/02/2010
SE AGREGA CONDICION PARA CUANDO EL TURNO SEA DEL PROCESO 20
EJECUTE LA FUNCION VAL_MASIVO*/
        THEN          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),sir_op_liquidacion.lqdc_valor)
        ELSE          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpq_valor_documento)
        END, 10,
CASE
WHEN DP.dcpq_valor_documento<=0
/** AHERRENO 04/02/2010
SE AGREGA CONDICION PARA CUANDO EL TURNO SEA DEL PROCESO 20
EJECUTE LA FUNCION VAL_MASIVO*/
        THEN          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),sir_op_liquidacion.lqdc_valor)
        ELSE          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpq_valor_documento)
        END, 4,
CASE
WHEN DP.dcpq_valor_documento<=0
/** AHERRENO 04/02/2010
SE AGREGA CONDICION PARA CUANDO EL TURNO SEA DEL PROCESO 20
EJECUTE LA FUNCION VAL_MASIVO*/
        THEN          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),sir_op_liquidacion.lqdc_valor)
        ELSE          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpq_valor_documento)
        END, 5,
CASE
WHEN DP.dcpq_valor_documento<=0
/** AHERRENO 04/02/2010
SE AGREGA CONDICION PARA CUANDO EL TURNO SEA DEL PROCESO 20
EJECUTE LA FUNCION VAL_MASIVO*/
        THEN          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),sir_op_liquidacion.lqdc_valor)
        ELSE          DECODE(sir_op_proceso.id_proceso,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpq_valor_documento)
        END, null )
FROM SIR_OP_APPLICACION_PAGO AP,
    sir_op_documento_pago DP
WHERE AP.ID_SOLICITUD =sir_op_solicitud.id_solicitud
AND AP.id_documento_pago=DP.id_documento_pago
AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
AND AP.id_liquidacion<=1
AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trmo_anulado = 'N'))
END )
ELSE
CASE
WHEN sir_op_proceso.id_proceso=4
THEN

```

```

(SELECT DECODE (DP.id_tipo_doc_pago, 3, DP.dcpg_valor_documento, 4, DP.dcpg_valor_documento, 5,
DP.dcpg_valor_documento, 10, DP.dcpg_valor_documento, NULL )
  FROM SIR_OP_APPLICACION_PAGO AP,
       sir_op_documento_pago DP
 WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
   AND AP.id_documento_pago = DP.id_documento_pago
   AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
   AND DP.id_tipo_doc_pago IN (3,4,5,10)
   AND AP.id_liquidacion>1
   AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N'))
 ELSE
/** AHERRENO 04/02/2010
SE AGREGA CONDICION PARA CUANDO EL TURNO SEA DEL PROCESO 20
EJECUTE LA FUNCION VAL_MASIVO*/
  (SELECT      DECODE (DP.id_tipo_doc_pago,      3,      DECODE(sir_op_proceso.id_proceso,      20,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpg_valor_documento),      4,      DECODE(sir_op_proceso.id_proceso,      20,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpg_valor_documento),      5,      DECODE(sir_op_proceso.id_proceso,      20,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpg_valor_documento),      10,      DECODE(sir_op_proceso.id_proceso,      20,
VAL_MASIVO(sir_op_solicitud.id_solicitud),DP.dcpg_valor_documento),NULL )
  FROM SIR_OP_APPLICACION_PAGO AP,
       sir_op_documento_pago DP
 WHERE AP.ID_SOLICITUD = sir_op_solicitud.id_solicitud
   AND AP.id_documento_pago = DP.id_documento_pago
   AND AP.id_liquidacion = sir_op_aplicacion_pago.id_liquidacion
   AND DP.id_tipo_doc_pago IN (3,4,5,10)
   AND AP.id_liquidacion>1
   AND AP.id_solicitud in (select tt.id_solicitud from sir_op_turno tt where tt.trno_anulado = 'N'))
END
END ,0)vr_docpago_cheque,

```

## 178 Integración Gestión Documental y SIR

### **REQUERIMIENTO**

Se creó la aplicación Gestión Documental, esta aplicación se encarga de hacer la publicación de estos mensajes.

Se debe realizar la integración de esta aplicación con el sistema de información SIR.

### **DESARROLLO**

Solución: Se integra la aplicación gestión documental con el sistema de información SIR con los siguientes cambios:

1. Se agrega a las librerías del proyecto SIR en netbeans el jar gestiondocumental.jar que se encuentra en C:\WORK\LIB.
2. Esta librería también debe estar en la ruta del jboss C:\jboss\server\default\lib.
3. Se deben descargar y compilar las clase nueva:

C:\WORK\SIR-  
APP\src\gov\sir\hermod\gdocumental\integracion\PublisherIntegracion.java

4. Se debe descargar y compilar las clases modificadas:
  - a) C:\WORK\SIR-  
APP\src\gov\sir\hermod\dao\impl\jdogenie\JDOTurnosDAO.java, se modifica las líneas 2011,5056
  - b) C:\WORK\SIR-

APP\src\gov\sir\core\negocio\acciones\registro\ANLiquidacionRegistro.java,  
Se modifica la línea 1623.

DESPLIEGUE – 2.3.1 09/02/2010	
INCIDENCIAS RESUELTA	
ID GLPI	OBSERVACIONES
<b>Error al Avanzar el Turno</b>	
13422	<p><b>REQUERIMIENTO</b></p> <p>Se comunica Rodrigo Florez Garcia manifiesta que el turno 2009-100-6-30057 al tratar de avanzarlo de la mesa de control no permite se queda cargando. Por favor verificar para que el funcionario pueda avanzar el turno 2009-100-6-30057. Se adjunta imagen. Manizales.</p> <p><b>DESARROLLO</b></p> <p>Se revisó la consistencia del turno y se encuentra bien. El bloqueo de las 404 matriculas está asignado al usuario LUZ.RENGIFO quien fue el usuario que calificó el turno pero no tiene incidencia en el avance del turno.</p> <p>Análisis: este proceso no culminaba, TIME UP, por lo que se decidió revisar el proceso y se encontró que dentro del código existían llamados a la base de datos innecesarias lo cual lo hace el proceso lento, cuando existen muchos folios asociados al turno da TIME UP.</p> <p>Solución: Se realizó las modificaciones pertinentes para evitar las consultas innecesarias a la base de datos.</p> <p>Clase: C:\WORK\SIR-APP\src\gov\sir\core\negocio\acciones\registro\ANMesaControl.java</p> <p>Método: private EventoRespuesta enviarRespuestaNuevo(Evento ev) throws EventoException</p> <p>Línea: 1237</p> <pre>/**  * @author: Fernando Padilla  * @change: (Refactoring - proceso avanzar turno mesa control muy lento, no existe acta, mantis 2945).  * Este linea se comenta, ya que, la variable local turnoMayorValor nunca es usada en el método.  */ //Turno turnoMayorValor = hermod.getTurnobyWF(itemKey); linea:1326 /**  * @author: Fernando Padilla  * @change: (Refactoring - proceso avanzar turno mesa control muy lento, no existe acta, mantis 2945).  * Este linea se comenta, ya que, no es necesario volver a la base de datos  * a buscar los turnos. */ //turnoTes=hermod.getTurno( new TurnoPk((Turno)turnosByMatricula.get(i)).getIdWorkflow() );</pre> <p><b>Error al Negar Turnos</b></p> <p>13646 <b>REQUERIMIENTO</b></p> <p>Se comunica la funcionaria Venovit Mejía e informa que al tratar de negar el turno 2007-350-4-145, ANALISIS X-FRANKLIN.PEREZ, el sistema notifica error. El turno 2007-350-4-145 se encuentra en la fase ANALISIS, lo van a negar, pero al momento de realizar el proceso el sistema notifica que el turno ya se encuentra negado. Relacionado con el caso 13491. Ibagué.</p>

	<p><b>DESARROLLO</b></p> <p>Análisis: En el proceso de devoluciones al momento de negar la devolución existe una validación preguntaba por toda la historia del turno para verificar si este ya no estuviera negado; la validación solo debe hacerse sobre la última historia del turno.</p> <p>Solución: Se modifico la validación para que solo preguntara sobre la última historia.</p> <p>Clase: gov.sir.core.negocio.acciones.devolucion.ANDevolucion</p> <p>Método: public EventoRespuesta perform(Evento e) throws EventoException</p> <p>Línea 329:</p> <pre>/**  * @author: Fernando Padilla Velez  * @change: Mantis 3050, Se modifico la validación para que solo preguntara sobre la última historia.  */ break;</pre>	
<b>Campo Oficina Origen Juzgado en Mayuscula (Informativo)</b>		
13861	<p><b>REQUERIMIENTO</b></p> <p>Informa Marlene Alaguna que al imprimir el formulario de corrección en el campo oficina origen Juzgado Segundo Civil del Circuito de Guamo aparece en minúscula y debe aparecer en mayúscula. Se realiza la consulta por pantalla administrativa: Administración de Oficinas Origen con los siguientes parámetros:</p> <p>Tipos de Oficina: Juzgado</p> <p>Departamento: Tolima</p> <p>Municipio: Guamo</p> <p>Listado: juzgado segundo civil del circuito aparece en Mayuscula, por lo tanto en el formulario debería aparecer en mayúscula, favor verificar. Guamo.</p> <p><b>DESARROLLO</b></p> <p>Análisis: El nombre de esta oficina se encuentra en minúscula, debe ser modificado a mayúscula.</p> <p>Solución: Se modifica el nombre de la oficina a mayúscula:</p> <pre>update sir_ne_oficina_origen set fcrg_nombre = 'JUZGADO SEGUNDO CIVIL DEL CIRCUITO' where id_oficina_origen = '2545'</pre> <p><b>OBSERVACIONES ETB</b></p> <p>Este cambio ya se encuentra en producción.</p>	

**Matriculas Sobrepuertas en los Certificados  
(Informativo)**

	<p><b>13963</b></p> <p><b>REQUERIMIENTO</b></p> <p>Folio 300-77605. Informa Lorenzo Silva que en la mayoria de las impresiones de certificados aparecen las matriculas sobreuestas, pero unicamente cuando se imprime en la impresora en la lexmark W840. Tambien informa que al realizar una Consulta por Indice de Propietario por pantalla administrativa en la mayoria de folios aparece sobreuesta. El usuario informa que al direccionar las impresiones a otra impresora el numero de matricula si aparece bien. Se le sugiere comunicarse con ofimatica de la SNR para que revisen la impresora Lexmark W840.</p> <p>Favor verificar porque se sobrepone el numero de matricula al realizar la consulta por consulta de indice de propietario, ver anexo. Bucaramanga.</p> <p><b>DESARROLLO</b></p> <p>Análisis: El error que se presenta no se puede replicar, los formularios se imprimen de forma correcta. Verificar el buen funcionamiento de la impresora lexmark W840.</p>	
<b>Fallas con la Firma del Registrador (Informativo)</b>		
14746	<p><b>REQUERIMIENTO</b></p> <p>Se comunica Fanny Patiño e informa que al momento de expedir un certificado el sistema notifica error. Por favor corregir dicho error. Gracias. Se adjuntan imágenes de error. La ceja.</p>	
14747	<p><b>REQUERIMIENTO</b></p> <p>Informa Cesar Silva que se estan imprimiendo los certificados sin la firma del registrador. Se verifica por pantalla administrativa la configuracion de la firma del registrador y se encuentra activa, favor verificar porque no aparece la firma. Neiva.</p> <p><b>DESARROLLO</b></p> <p>Análisis: Este error se produjo, ya que, los Files Systems de las firmas no se encontraban montados en los servidores 192.168.202.1, 192.168.202.2, 192.168.202.3, 192.168.202.4, 192.168.202.5, 192.168.202.6, 192.168.202.7, 192.168.202.8, no pudiendo acceder a estas firmas.</p> <p>Solución: El ingeniero Francisco Beltran monto los Files Systems en los servidores descritos arriba.</p> <p>El comando utilizado desde consola de linux fue:</p> <pre>mount 192.168.202.12:/vol/vol_firmas /firmas/</pre>	
<b>001 Consultas de indices de propietario (Informativo)</b>		
<p><b>REQUERIMIENTO</b></p> <p>Consultas de indices de propietario</p> <p>SIR-318</p> <p>De conformidad con el tema del asunto, de manera atenta solicito realizar los ajustes necesarios para dar cumplimiento con lo requerido en la comunicación SDR 1002 INT, enviado por el Doctor Javier Darío Anaya — Superintendente Delegado para el Registro</p>		

y que se resumen así:

- Las consultas de índices de propietarios no arrojan información y están desactualizadas.

**SEGUIMIENTO ADA:**

El documento anexo no corresponde al requerimiento enunciado. Por favor verificar y especificar el requerimiento, indicando la causa de desactualización

**OBSERVACIONES SNR:**

Se considera que este caso debe cerrarse toda vez que ha verificado y se efectuaron ajustes en este proceso de consultas y reportes de este proceso buscando el mejoramiento de la funcionalidad.

**DESARROLLO**

Basada en la observación se procedió a revisar el proceso Pantallas Administrativas->OTROS-> Consulta por Índice de Propietario con la ayuda del personal de la mesa de ayuda para contrastar la información suministrada por la SNR, de lo cual se determinó que la observación del requerimiento es acertada por lo tanto el estado del requerimiento se actualiza a Solucionado.

**005 Calificación Anotaciones temporales segregadoras  
(Informativo)**

**REQUERIMIENTO**

Calificación: Anotaciones temporales segregadoras.

SIR-306.

El programa está permitiendo que se eliminen anotaciones temporales segregadoras de folios de matrícula matrices, sin eliminar los folios de matrículas segregados. Se debe validar que cuando se elimine dicha anotación, los folios derivados queden eliminados también

**SEGUIMIENTO ADA:**

Se requiere definir el escenario donde se reproduce el error. Se requiere la definición del escenario donde se produjo el error, indicando los siguientes: Fecha (tiempo en el que se generó) Proceso (Proceso del sistema donde se evidenció el error) Turno (turno que presenta la inconsistencia).

**OBSERVACIONES SNR:**

Realizar nuevamente pruebas, si se presenta, es de carácter URGENTE su arreglo, verificar información en la tabla mat\_no\_grabadas cuando se elimine la anotación segregadora. Verificar GLPI se actualmente se encuentran reportados casos con esta inconsistencia, para determinar si son casos eventuales.

**DESARROLLO**

Se realizaron pruebas y la aplicación funciona correctamente.

**014 Certificado de Pertenencia Impresión certificado  
(Informativo)**

**REQUERIMIENTO**

Certificado de Pertenencia: Impresión certificado

SIR-250

En la fase oficio de pertenencia, el sistema debería permitir configurar la plantilla de la

certificación que debe expedir el registrador conforme al Artículo 407 del Código de Procedimiento Civil, con la opción de escoger el número de copias que se desea imprimir y permitir la impresión del certificado de tradición, cuando el inmueble sobre el cual se está solicitando la certificación ya tiene matrícula asignada.

**SEGUIMIENTO ADA:** Dar mayor claridad al requerimiento, indicando lo referente a la construcción de la plantilla.

#### OBSERVACIONES SNR:

Si bien en SIR existe el campo para diligenciar el texto, guarda la información y lo imprime, no es posible entregar este oficio al ciudadano ya que al digitarlo no permite justificación del texto y al imprimir este texto actualmente sale en letras minúsculas debe implementarse la herramienta que permita justificarlo, igualmente que al imprimirla salga en mayúsculas, igualmente tenga la caja de texto para incluir el número de copias a imprimir. Actualmente el aplicativo SIR imprime la certificación ingresada en el campo de texto (se pega de un archivo diligenciado en otro programa) y el certificado cuando el inmueble sobre el cual se pide el certificado ya tiene matrícula asignada, al usuario se le entrega la certificación impresa en papel con membrete de la SNR diligenciado en la herramienta Word u office y la que imprime el SIR se guarda en la carpeta del archivo de la oficina. Como el texto de este tipo de certificado en todos los casos no es igual no es posible diligenciar una plantilla única para este tipo de certificado. (Hay certificado de pertenencia con matrícula, certificado de pertenencia sin matrícula).

#### **DESARROLLO**

Se realizaron pruebas y no fue posible realizar el cambio, debido a la forma como fue hecha la aplicación de impresión.

Para poder realizar el cambio se tendría que hacer nuevamente la parte de impresión, porque no toma el formateo del texto (TextLayout).

### **118 Liquidación Acto 38 Vivienda de Interés Social**

#### **REQUERIMIENTO**

Se requiere modificar el aplicativo SIR por cuanto en la radicación de documentos al ingresar el acto 38 “**VIVIENDA DE INTERES SOCIAL**” esta liquidando la cifra ingresada como cantidad no como cuantía (multiplica el valor digitado por la tarifa mínima de registro), adicionalmente los certificados asociados a éste acto, los está cobrando con el valor de 2.150.00 pesos, siendo lo correcto 1080 pesos). Para este acto debe permitir incluir cuantía y seleccionar la tarifa media (conforme a art 13 dcto 2280 de 2008 y resolución 35 de 2009).

No sobra aclarar que en lo referente al artículo 14 del decreto antes mencionado, el aplicativo esta liquidando adecuadamente (**identificación de inmuebles con planos prediales catastrales**).

#### **DESARROLLO**

**/SIR-APP/src/gov/sir/core/negocio/modelo/constantes/CActo.java**

```
/**  
 * @author Fernando Padilla  
 * @change Requerimiento externo 118, Caso Constante para identificar que  
 * un <code>Acto </code> es código 38 (VIVIENDA DE INTERES SOCIAL).  
 */  
public static final String VIVIENDA_INTERES_SOCIAL = "38";
```

#### **/SIR-APP/src/gov/sir/core/negocio/modelo/constantes/CTipoCertificado.java**

```
/**  
 * @author Fernando Padilla  
 * @change Requerimiento externo 118, Caso Constante para identificar que  
 * un <code>Acto </code> es código 38 (VIVIENDA DE INTERES SOCIAL).  
 */  
public static final String VIVIENDA_INTERES_SOCIAL = "38";  
public static final String TIPO_ASOCIADO_38_NOMBRE = "TIPO_ASOCIADO_38";  
public static final String TIPO_ASOCIADO_38 = "14";
```

#### **/SIR-APP/src/gov/sir/hermod/pagos/LiquidadorRegistro.java**

```
/**  
 * @author Fernando Padilla  
 * @change Requerimiento externo 118, Se asocian los actos Código 38.  
 */  
Iterator itCodigo38 = listaActos.iterator();  
while (itCodigo38.hasNext())  
{  
    TipoActo tipo = (TipoActo) itCodigo38.next();  
    if (tipo.getIdTipoActo().equals(CActo.VIVIENDA_INTERES_SOCIAL))  
    {  
        listaExcepciones.add(tipo);  
    }  
}  
  
/**  
 * @author Fernando Padilla  
 * @change caso requerimiento externo 138, ACTO CODIGO 38  
 */  
if (idActo.equals(CActo.VIVIENDA_INTERES_SOCIAL))  
{  
    nombreTipoCertificado =  
CTipoCertificado.TIPO_ASOCIADO_38_NOMBRE;  
    idTipoCertificado = CTipoCertificado.TIPO_ASOCIADO_38;  
}
```

El tipo de acto 38 Vivienda de Interés Social se puede cambiar por el aplicativo pero al realizar este cambio el sistema arroja error, también se intenta cambiar la tarifa para el certificado asociado para este acto y el sistema no hace el cambio de tarifa. Es necesario revisar a nivel de código y base de datos los cambios a realizar para este requerimiento.

Anexo imagen del error.



Este caso se debe resolver desde pantallas administrativas, ya que, el error mencionado fue resuelto en la versión de producción 2.3.0.

Por favor, asignarle al administrador nacional para que realice este cambio.

### **Administrador Nacional**

(Javier Rocha 04/01/2010)

La actualización para el acto 38 Vivienda de Interés social fue ejecutada en el sistema cambiando este acto de porcentual a absoluto cobrando una tarifa fija de 1080.

TIPO ACTO	VIVIENDA DE INTERES SOCIAL	1000.0	1	6
TIPO ACTO	VIVIENDA DE INTERES SOCIAL	2000.0	2	7
TIPO ACTO	VIVIENDA DE INTERES SOCIAL	2150.0	3	8
TIPO ACTO	VIVIENDA DE INTERES SOCIAL	1080.0	4	9

### **026 Actualización de la información de si la persona es o no propietaria (Informatito)**

#### **REQUERIMIENTO**

Actualización de la información de si la persona es o no propietaria.

SIR-191.

Se han presentado varios casos donde el sistema arroja el resultado pero en el campo propietario aparece como si el consultado NO lo fuera, cuando realmente sí lo es, ejemplo: Se consultó por cedula 41.907.917 y nombre ADRIANA ROMERO GIRALDO, arrojó los folios 280-66391 y 280-66380 pero dice que NO es propietario, cuando al revisar el folio se encuentra que SI es propietaria. La misma situación se pudo observar para los folios en los que aparece la persona consultada con derechos de cuota, ejemplo: Con la cédula 24.922.731 y el nombre LUDOVINA HOYOS MUÑOZ, arrojó la matrícula 280-78936 en la que en el resultado de la consulta aparece como si NO fuera propietaria, cuando realmente en el folio SÌ figura como propietaria de derechos de cuota.

#### **SEGUIMIENTO ADA:**

Se pregunta si se altera la consulta para que la consulta indique so los ciudadanos fueron alguna vez propietarios de esos predios.

#### **OBSERVACION SNR:**

De hecho, la lógica de la consulta indica "NO" para los registros encontrados donde no es propietaria la persona (natural o jurídica), es necesario verificar información específicamente en las tablas, si los códigos de limitación de dominio entre los que se encuentran incluidos los actos que transfieren dominio sobre derechos de cuota el sistema debe tenerlos en cuenta para la consulta de índices de propietarios.

#### **DESARROLLO**

En el requerimiento la SNR especifica problemas en las consultas de índice de propietario Pantallas Administrativas->OTROS-> Consulta por Índice de Propietario al igual que el Requerimiento 001\_141 caso mantis 03324 el cual fue cerrado basado en observación del equipo de SNR.

En este caso particular se adjunta información de consultas realizadas para:

- Cedula 41.907.917 y nombre ADRIANA ROMERO GIRALDO la cual arroja que el ciudadano se encuentra relacionado a los folios 280-66391 y 280-66380 como NO propietario. En la información suministrada se dice que esta si es propietaria de los folios mencionados.

Luego de revisar el folio 280-66380 se encontró que el propietario del mismo es CONDOMINIO SAN DIEGO BLOQUE A anotación N° 10 y en la revisión del folio 280-66391 el propietario es el mismo del folio anterior información que se puede encontrar en la anotación N° 15 del folio.

- Cedula 24.922.731 y el nombre LUDOVINA HOYOS MUÑOZ, arrojó la matrícula 280-78936 en la que en el resultado de la consulta aparece como si NO fuera propietaria.

Luego de revisar el folio 280-78936 se encontró que el propietario del mismo es NIDIA GARCIA identificada con la cedula 24.494.928 anotación N° 8.

Se deduce de las consultas realizadas que al momento de interpretar la información de los folios se incurrió en errores, se aconseja tener en cuenta para futuras interpretaciones que las anotaciones que determinan quien o quienes son los propietarios de un predio referenciado por una matrícula, son aquellas de carácter permutables que especifican el dominio sobre un predio Ej. Ventas, Sucesiones, etc.

## 029 Bloqueos y Desbloqueos de Turno

### **REQUERIMIENTO**

Cuando un numero de matricula se encuentra asociado a un turno de corrección y un turno de registro el sistema no permite tomar ninguno de los dos, cuando lo establecido es que la prioridad de turno la tiene el turno de corrección, se reporto a la mesa de ayuda y dieron la sugerencia de deshacer cambios al turno de registro aunque no se había incluido ninguna información en este folio, con esta operatividad se da solución al caso, pero esta no es la solución, el mismo sistema debe manejar estos bloqueos y desbloqueos.

### **DESARROLLO**

En el proceso de correcciones el aplicativo presenta inconvenientes cuando la matricula asociada a un turno de correcciones también se encuentra asociada a un turno de registro, presentándose un bloqueo mutuo que no permite en el avance de ninguno de los turnos.

La solución de este requerimiento es permitir que el turno de correcciones tenga prioridad sobre el turno de registro y avance hasta la fase de entrega y luego permitir que el o los turnos de registro continúen en el orden cronológico de su radicación.

Para realizar este proceso se procedió a realizar modificaciones al paquete de la Base de datos PCK\_SIR\_VALIDACION específicamente al procedimiento SP\_DELEGAR\_BLOQUEO\_FOLIOS. Las cuales están especificadas en el ítem de modificación de archivos.

- Script Bases de Datos:
  - Archivo de Modificación (MODIFICADO)PCK\_SIR\_VALIDACIONES.sql
  - Archivo respaldo sin Modificación (COPIA)cuerpo\_PCK\_SIR\_VALIDACIONES.sql
- Creación De Clases:  
NINGUNO

- Se modificaron los siguientes archivos:

<b>PAQUETE:</b>
<b>PCK_SIR_VALIDACION</b>
<b>Procedimiento:</b>
<b>SP_DELEGAR_BLOQUEO_FOLIOS</b>

**Código Incluido Línea 120:**

```
/* 10/12/2009 JAlcazar Caso Mantis 0003384
 * Valida que la matricula asociada no posee anotaciones temporales en algun turno de Registro
 */
IF FC_GET_COUNT_ANOTTEMP_FOLIO (v_bloqueo_folio.id_matricula) > 0 AND v_bloqueo_folio.id_proceso = 6 THEN
    o_errores.EXTEND;
    num_ERRORES := num_ERRORES +1;
    O_ERRORES(num_ERRORES) := t_matriculas_obj(v_folios_turno(n).ID_MATRICULA, 'La Matricula tiene
anotaciones temporales, por favor deshacer cambios en : '|| v_bloqueo_folio.BLFL_ID_WORKFLOW);
END IF;
```

**Código Modificado Línea 169:**

```
/* 10/12/2009 JAlcazar Caso Mantis 0003384
 * Se valida que el turno que bloquea el turno no sea de Registro (v_bloqueo_folio.id_proceso <> 6)
 */
IF v_bloqueo_folio.BLFL_ID_WORKFLOW <> v_turno.TRNO_ID_WORKFLOW
    AND v_bloqueo_folio.id_proceso <> 6
    AND (v_id_workflow_aux1 IS NULL OR v_bloqueo_folio.BLFL_ID_WORKFLOW <> v_id_workflow_aux1)
    AND (v_id_workflow_aux2 IS NULL OR v_bloqueo_folio.BLFL_ID_WORKFLOW <> v_id_workflow_aux2) THEN
```

**Código Modificado Línea 348:**

```
/* 10/12/2009 JAlcazar Caso Mantis 0003384
 * Se valida que el turno que bloquea el turno no sea de Registro (v_bloqueo_folio.id_proceso <> 6)
 */
IF v_usuario.ID_USUARIO <> p_id_usuario
    AND p_id_workflow <> v_bloqueo_folio.blfl_id_workflow
    AND v_bloqueo_folio.id_proceso <> 6
    AND (v_id_workflow_aux2 IS NULL OR v_id_workflow_aux2 <> v_bloqueo_folio.blfl_id_workflow)
    AND (v_id_workflow_aux1 IS NULL OR v_id_workflow_aux1 <> v_bloqueo_folio.blfl_id_workflow) THEN
```

## 030 Rol Digitador masivo (Informativo)

**REQUERIMIENTO**

El Rol Digitador masivo permitía corregir la información que el mismo usuario adicionaba y en el momento no permite esta funcionalidad.

**DESARROLLO**

Se realizaron pruebas y sí permite corregir la información. Se da por cerrado el caso.

## 061 Acto de Cancelacion de Usufructo (Informativo)

**REQUERIMIENTO**

En el proceso de radicación de documentos cuando se ingresa un acto de cancelación de Usufructo y se agrega una solicitud de certificado asociado, el aplicativo liquida el valor del certificado por un valor de \$1.080.oo, esto es un error del aplicativo, el valor que debe liquidar es el de acuerdo a la tarifa seleccionada .En el Reporte No.6 Diario radicador de Certificados esta inconsistencia se refleja con un tipo de certificado TIP el cual no existe

**DESARROLLO**

Se realizo la configuración respectiva del acto 30 – Cancelacion de Usufructo.

## 076 Impresión certificados asociados

**REQUERIMIENTO**

Impresión certificados asociados. Nos informa la funcionaria Venovit Mejía que al momento de imprimir certificados asociados no están saliendo en orden.

Se verifican las relaciones 2009-350-1153, 2009-350-1154, 2009-350-1169 en el sistema y estas aparecen con los turno en orden, pero al momento de imprimir estas relaciones de certificados asociados estos están saliendo en desorden, es decir no están saliendo de manera consecutiva. Por favor revisar esta caso, se adjuntan las relaciones para verificar el caso. Se abre caso en Mantis No 1769. Id Glpi 11174.

### **DESARROLLO**

Al momento de enviar el formulario para imprimir, no se envían los turnos de registro, se envían los turnos asociados. Los turnos de registro son ordenados para mostrar el formulario pero no al imprimir.

Se ordenan los turnos asociados recibidos, antes de imprimir, en el orden de los turnos de registro de la misma forma que se hace al renderizarlos en el formulario.

<b>CLASE:</b>
gov.sir.core.web acciones.registro. AWMesa.java
<b>Se añadieron las importaciones:</b>
<pre>import gov.sir.core.negocio.modelo.util.IDidworkflowComparator; import java.util.Collections; import java.util.Hashtable;</pre>
<b>Método:</b>
private Evento imprimirCertificadoRelacion(HttpServletRequest request)
<b>Se cambió el código:</b>
<pre>//SE AGREGAN LOS CERTIFICADOS ASOCIADOS QUE SE DESEAN IMPRIMIR for (int i = 0; i &lt; llavesSol.length; i++) {     String sIndice=llavesSol[i];     if(sIndice!=null){         certificadosAsociados.add(sIndice);     } }</pre>
<b>Por:</b>
<pre>Hashtable turnosRegistro = (Hashtable) session.getAttribute(WebKeys.LISTA_TURNOS_RELACION); Hashtable turnosValidosImpresion = (Hashtable) session.getAttribute(WebKeys.LISTA_TURNOS_VALIDOS); ArrayList keysTurnosRegistro = new ArrayList(turnosRegistro.keySet()); Collections.sort(keysTurnosRegistro, new IDidworkflowComparator()); Iterator iter = keysTurnosRegistro.iterator(); Boolean turnoValidoImpresion = null;  while(iter.hasNext()){     String key = (String)iter.next();     List turnosCertificados = (List)turnosRegistro.get(key);     if(turnosCertificados.size() &gt; 0){         Iterator it = turnosCertificados.iterator();         while(it.hasNext()){             SolicitudCertificado solicitudCertificado = (SolicitudCertificado)it.next();             Turno turnoCertificado = solicitudCertificado.getTurno();             turnoValidoImpresion = (Boolean)turnosValidosImpresion.get(turnoCertificado.getIdWorkflow());             if(turnoValidoImpresion!=null){                 if(turnoValidoImpresion.booleanValue()){                     if(turnoCertificado!=null){                         String certificadosAsociadosT = turnoCertificado.getIdWorkflow();                         //SE AGREGAN LOS CERTIFICADOS ASOCIADOS QUE SE DESEAN IMPRIMIR                         forCertificadosAsociados:                             for (int i = 0; i &lt; llavesSol.length; i++) {                                 String sIndice=llavesSol[i];                                 if(sIndice!=null &amp;&amp; sIndice.equals(certificadosAsociadosT)){                                     certificadosAsociados.add(sIndice);                                     break forCertificadosAsociados;                                 }                             }                     }                 }             }         }     } }</pre>

```

        }

/*SE AGREGAN LOS CERTIFICADOS ASOCIADOS QUE SE DESEAN IMPRIMIR
for (int i = 0; i < llavesSol.length; i++) {
    String sIndice=llavesSol[i];
    if(sIndice!=null){
        certificadosAsociados.add(sIndice);
    }
}

```

- Este cambio se realiza para mostrar el botón de eliminar en los nuevos actos agregados cuando fue ingresado un turno anterior.

## 087 Impresión Notas Informativas Correcciones (Informativo)

### **REQUERIMIENTO**

Impresión Notas Informativas Correcciones. En el proceso de correcciones se pueden crear notas informativas, al tratar de imprimirlas directamente desde el rol, el sistema no las imprime y es necesario ir a la pantalla administrativa ver detalles de turno para realizar dicha impresión. Se requiere escalar este caso a desarrollo para que sea corregido el tema de impresiones de notas informativas en el proceso de correcciones ya que estas impresiones deben salir por administrador de impresión por el tipo de impresión otros imprimibles. Mesa de Ayuda.

### **DESARROLLO**

Se realizaron pruebas en el ambiente de producción con en el rol Responsable Correcciones al dar clic en el botón imprimir este imprime sin ningún inconveniente. Por tal motivo se cierra este caso.

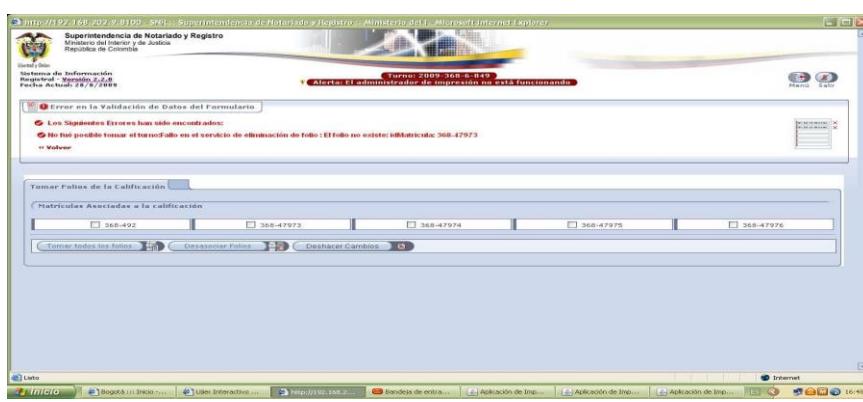
## 092 fase calificación deshacer cambios – segregar

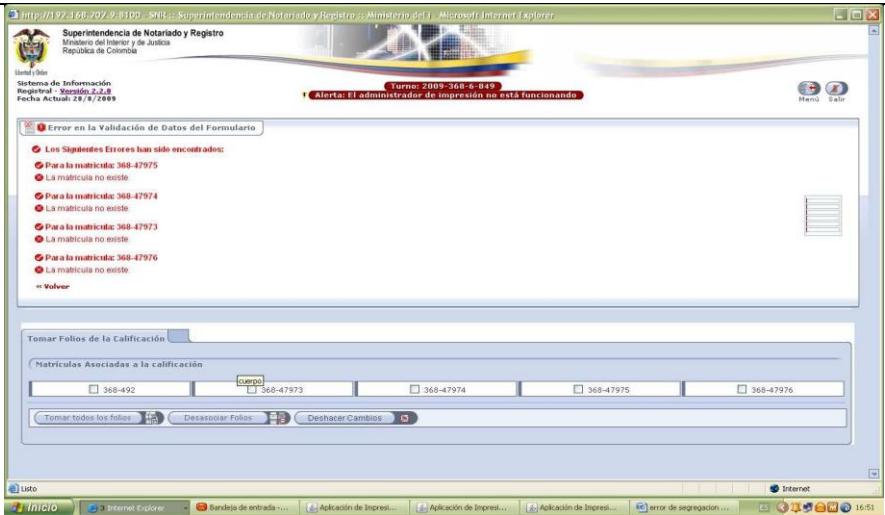
### **REQUERIMIENTO**

#### OBSERVACION SNR:

Se solicita hacer revisión de la fase de calificación al momento de deshacer cambios cuando se ha realizado una segregación, para evitar los errores generados en la prueba.

Se realizó prueba con el turno 2009-368-6-849, matrícula 368-492, en la fase de calificación se segregaron cuatro matrículas, la 368-47973 a la 368-47976 y al deshacer los cambios el sistema muestra el siguiente error:





Al regresar al menú y al retomar el turno 2009-368-6-849 se observa que se han eliminado las matrículas segregadas y toda la información temporal agregada con este turno, es decir que el sistema deshace los cambios pero no refresca la pantalla por lo que muestra los anteriores mensajes de error y confunde al usuario.

## **DESARROLLO**

El error ocurría al intentar recargar la página con los folios que ya no existían.

Después de deshacer los cambios se envía el control a la página donde se selecciona nuevamente el turno y se pueda realizar las reglas de negocio necesarias para visualizar el turno.

- Se modificaron los siguientes archivos:

<b>CLASE:</b> gov.sir.core.web.acciones.registro.AWCalificacion.java
<b>Método:</b>
<i>private Evento doProcess_BtnDeshacerCambios(HttpServletRequest request )</i>
Se agregó la línea: <i>request.getSession().setAttribute(WebKeys.DESHICIERON_CAMBIOS_CALIFICACION, true);</i>
<b>CLASE:</b> gov.sir.core.web.acciones.registro.AWCalificacion.java
<b>Método:</b>
<i>private Evento doProcess_BtnDeshacerCambios(HttpServletRequest request )</i>
Se agregó la línea: <i>request.getSession().setAttribute(WebKeys.DESHICIERON_CAMBIOS_CALIFICACION, true);</i>
<b>ARCHIVO:</b> WEB-INF/jspSeguro/comun/HEADER_FASES_LOGOUT.jsp
<ul style="list-style-type: none"> <li>• Se cambió el nombre del formulario con la ACCION CONSULTAR_PROCESO: logOut por logOutM</li> <li>• <b>Se agregó el código:</b></li> </ul> <pre>&lt;%&gt;     if(request.getSession().getAttribute(WebKeys.DESHICIERON_CAMBIOS_CALIFICACION) != null){         request.getSession().removeAttribute(WebKeys.DESHICIERON_CAMBIOS_CALIFICACION);     }     &lt;script&gt;         document.logOutM.submit();     &lt;/script&gt; &lt;%&gt;     } %&gt;</pre>

**093 Proceso Calificación - Avance de turnos cert asociados**

## (Informativo)

### **REQUERIMIENTO**

Se realizaron pruebas en el ambiente de calidad con el turno de registro 2009-300-6- 18243 se asocio la matricula 300-1250, en la cual se efectuó modificación de ciudadanos en el proceso de correcciones. El aplicativo permitió en el proceso de calificación agregar una anotación, avanzar el turno hasta la fase de firma del Registrador, presentando inconsistencia de no permitir avanzar el turno a la fase siguiente certificados asociados, muestra mensaje de error

“No se pudieron hacer definitivos los folios: Fallo en el servicio de delegar bloqueo de folio : r argument to raise\_application\_error of -1422 is out of range.

### **DESARROLLO**

El requerimiento especifica que el sistema presenta un problema en el proceso de registro al momento de proseguir a la fase de Certificados Asociados luego que el turno de registro haya realizado un proceso de correcciones en la cual se modifique información de los ciudadanos en una anotación, después de la corrección agregar una anotación y pasar a la fase de firma de Registrador.

Se reviso el proceso en los ambientes de pruebas, calidad y producción y no se logro evidenciar el problema mencionado, de igual manera se consulto con la mesa de ayuda para conocer si los usuarios reportaron un incidente de esa naturaleza y no se recibió reporte alguno. Se determina cerrar el caso.

## **096 Radicacion de Certificados Antiguo Sist-Pertenencia y Ampliacion de Tradicion**

### **REQUERIMIENTO**

Al consultar el detalle de los Turnos, se observa que para los turnos de certificados de Antiguo sistema, pertenencia y ampliación en la sección de historial de fases del turno el aplicativo erradamente esta duplicando la información en la fase de Solicitud al momento que se ejecuta la radicación.

### **DESARROLLO**

Al realizar el proceso de la pantalla administrativa MANEJO DE TURNOS -> Ver Detalles de Turnos y se solicitaba ver los detalles de un turno de certificados de antiguo sistema o un turno de certificados masivo el usuario visualizaba dos veces la fase inicial SOLICITUD en las fases del historial del turno, este problema se presentaba por que al momento de realizar la solicitud de los certificados nombradas el aplicativo agrega dos fases SOLICITUD y CER\_SOLICITUD las cuales poseen el mismo nombre de fase SOLICITUD.

Para evitar visualizar esta fase dos veces y dado que ellas corresponden al mismo proceso se opto por colocar un IF que no imprima la fila de la tabla de fases cuando esta corresponda a la fase CER\_SOLICITUD.

### **ARCHIVO JSP:**

#### **WEB\_INF.jspSeguro.administracion. TURNO\_CERTIFICADOS\_DETALLES\_VISTA.jsp**

Código Línea 1421:

```
<table width="100%">
<tr>

<td width="1%"></td>
<td width="15%" class="contenido"><B>FASE</B></td>
<td width="1%"></td>
<td width="15%" class="contenido"><B>ATENDIDO POR</B></td>
```

```

<td width="1%"></td>
<td width="15%" class="contenido"><b>FECHA</b></td>
<td width="1%"></td>
<%if(turno.getIdProceso() == Long.parseLong(CProceso.PROCESO_DEVOLUCIONES)) {%
<td width="31%" class="contenido"><b>ACCIONES REALIZADAS</b></td>
<%}else{ %
<td width="15%" class="contenido"><b>RELACION FASE</b></td>
<td width="1%"></td>
<td width="15%" class="contenido"><b>RELACION SIGUIENTE</b></td>
<%} %>

</tr>
</table>

```

**Por:**

```

<!--JALCAZAR Caso Mantis 0002866
* Se modifico la estructura de la tabla que tenia problemas de formato
<table width="100%" border="0" cellpadding="0" cellspacing="0" class="camposform">
<thead>
<tr>
<th width="1%"></th>
<th width="15%">FASE</th>
<th width="1%"></th>
<th width="15%">ATENDIDO POR</th>
<th width="1%"></th>
<th width="15%">FECHA</th>
<th width="1%"></th>
<%if(turno.getIdProceso() == Long.parseLong(CProceso.PROCESO_DEVOLUCIONES)) {%
<th width="31%">ACCIONES REALIZADAS</th>
<%}else{ %
<th width="15%">RELACION FASE</th>
<th width="1%"></th>
<th width="15%">RELACION SIGUIENTE</th>
<%} %>
</tr>
</thead>

```

**Agrego Linea 1448:**

```

/*JALCAZAR Caso Mantis 0002866
* Se argro if(!turnoAux.getFase().equals("CER_SOLICITUD")) que evita agregar dos veces la fase SOLICITUD
* a la tabla de fases esto debido a que cuando se crea un turno de certificado por antiguo sistemas o uno para
* certificados masivos la aplicación automaticamente agrega a el historial del turno la fase CER_SOLICITUD
* que corresponde al nombre de fase SOLICITUD
*/
if(!turnoAux.getFase().equals("CER_SOLICITUD")){

```

## WEB\_INF.jspSeguro.administracion. TURNO\_REGISTRO\_DETALLES\_VISTA.jsp

**Código Linea 1008:**

```

<table>
<tr>
<td width="1%"></td>
<td width="15%" class="contenido"><b>FASE</b></td>
<td width="1%"></td>
<td width="15%" class="contenido"><b>ATENDIDO POR</b></td>
<td width="1%"></td>
<td width="15%" class="contenido"><b>FECHA</b></td>
<td width="1%"></td>
<td width="15%" class="contenido"><b>RELACION FASE</b></td>
<td width="1%"></td>
<td width="15%" class="contenido"><b>RELACION SIGUIENTE</b></td>
</tr>
</table>

```

**Por:**

```

<!--JALCAZAR Caso Mantis 0002866
* Se modifico la estructura de la tabla que tenia problemas de formato
-->
<table width="100%" border="0" cellpadding="0" cellspacing="0" class="camposform">
<thead>
<tr>
<th width="1%"></th>
<th width="15%">FASE</th>
<th width="1%"></th>
<th width="15%">ATENDIDO POR</th>
<th width="1%"></th>
<th width="15%">FECHA</th>
<th width="1%"></th>

```

```

<th width="15%>RELACION FASE</th>
<th width="1%></th>
<th width="15%>RELACION SIGUIENTE</th>
</tr>
</thead>

Agrego Linea 1032:
/*JALCAZAR Caso Mantis 0002866
 * Se agrega if(!turnoAux.getFase().equals("CER_SOLICITUD")) que evita agregar dos veces la fase SOLICITUD
 * a la tabla de fases esto debido a que cuando se crea un turno de certificado por antiguo sistemas o uno para
 * certificados masivos la aplicación automáticamente agrega a el historial del turno la fase CER_SOLICITUD
 * que corresponde al nombre de fase SOLICITUD
 */
if(!turnoAux.getFase().equals("CER_SOLICITUD")){

```

## 098 Reporte 6 – Turnos Anulados de Otras ORIPS

### **REQUERIMIENTO**

Se requiere que en el reporte 6 diario radicador de certificados del día se realice los siguientes cambios:

- En el campo Total certificados otras ORIPS no se tengan en cuenta los certificados anulados
- En el campo Derechos certificados otras no se tenga en cuenta el valor de los certificados anulados

En el campo Total derechos: se debe modificar el nombre por “ total derechos recaudados” , además el valor total debe corresponder al valor Derechos certificados locales mas el valor Derechos certificados otras

### **DESARROLLO**

Se realizaron las siguientes modificaciones:

1. En el campo Total certificados otras ORIPS no se tengan en cuenta los certificados anulados.

Este caso se resolvió con el requerimiento 143.

2. En el campo Derechos certificados otras no se tenga en cuenta el valor de los certificados anulados.

Este caso se resolvió con el requerimiento 143.

3. En el campo Total derechos: se debe modificar el nombre por “ total derechos recaudados” , además el valor total debe corresponder al valor Derechos certificados locales mas el valor Derechos certificados otras

### Reporte 6 – Diario Radicador de Certificados del día

#### Campo Total Derechos Recaudos

Antes	\$V{SUM_DERECHOS_1}!=null?\$V{SUM_DERECHOS_1}:new BigDecimal(0.0).add( new BigDecimal(\$V{v_total_otras_orips}.doubleValue()) )
Ahora	/*Se agrega el valor de los derechos de otras ORIPS*/ \$V{SUM_DERECHOS_1}!=null?\$V{SUM_DERECHOS_1}.add( new BigDecimal( \$V{v_total_otras_orips}.doubleValue()):new BigDecimal(0.0).add( new BigDecimal(\$V{v_total_otras_orips}.doubleValue()) )

Se adjunta fuentes del reporte:

- SUBREPORTE\_CIRCULO.jasper
- reporte-06.jasper

Se realizan pruebas en la aplicación de calidad con:

- El circulo: BUCARAMANGA

LIQUIDADOR REGISTRO = HUGO.MANTILLA  
ENTREGA CERTIFICADOS = ZORAIDA.POVEDA

Se radicaron los siguientes certificados:

- Turno 2010-300-1-33 –Folio 490
- Turno 2010-314-1-2 –Folio 314-251 (Nacional)
- Turno 2010-300-1-34 –Folio 492
- Turno 2010-300-1-35 –Folio 493
- Turno 2010-450-1-1 –Folio 450-185 (Nacional)
- Turno 2010-314-1-3 –Folio 314-871 (Nacional)
- Turno 2010-300-1-36 –Folio 471
- Turno 2010-314-1-4 –Folio 314-112 (Nacional)

Se genera Reporte 06 – Diario Radicador de Certificados del día

Hora	Turno Anterior	Turno	Matrícula	Nombre o Razón Social	Copias	Derechos	Asociado	Cajero	Recibo	Tipo	Observaciones
10:10:16		2010-300-1-32	300-488	PRUEBAS	1	11840	2010-300-4-8	52689	42431765	ASO	IMPRESO
10:23:56		2010-300-1-33	300-490	PRUEBAS	1	11840		52689	42431767	INM	IMPRESO
10:25:16		2010-314-1-2	314-251	PRUEBAS	1	11840		52689	42431768	IMMANAL	IMPRESO
10:26:16		2010-300-1-34	300-492	PRUEBAS	1	11840		52689	42431769	INM	IMPRESO
10:27:04		2010-300-1-35	300-493	PRUEBAS	1	11840		52689	42431770	INM	IMPRESO
11:15:50		2010-450-1-1	450-185	PRUEBAS	1	11840		52689	42431771	IMMANAL	IMPRESO
11:26:14		2010-314-1-3	314-871	PRUEBAS	1	11840		52689	42431772	IMMANAL	IMPRESO
11:27:25		2010-300-1-36	300-471	PRUEBAS	1	11840		52689	42431773	INM	IMPRESO
11:28:16		2010-314-1-4	314-112	PRUEBAS	1	11840		52689	42431774	IMMANAL	IMPRESO

**CERTIFICADOS EN OTRAS ORIPS**

Total certificados expedidos:	9
Total certificados reimpresos:	0
Total certificados anulados:	0
Total certificados otras ORIPS:	4

**Derechos certificados locales:** \$ 59,200  
**Derechos certificados otras ORIPS:** \$ 47,360  
**Total derechos recaudos:** \$ 106,560

**CERTIFICADOS DE OTRAS ORIPS**

Círculo	Cantidad	Valor
PIEDECUESTA (314)	3	\$ 35,520
SAN ANDRES (450)	1	\$ 11,840
<b>Total certificados entre ORIPS'S</b>		<b>\$ 47,360</b>

Se anulan los siguientes turnos:

- 2010-314-1-3
- 2010-300-1-33

Untitled Document - Windows Internet Explorer  
 http://192.168.202.9:8100/SNR/ssp/plantillas/report-displayer/reportes.do

**OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS**  
**DIARIO RADICADOR DE CERTIFICADOS DEL DIA 2010-01-22**  
 Impreso el 22 de Enero de 2010 a las 11:39:10 AM

Página 1

Hora	Turmo Anterior	Turmo	Matrícula	Nombre o Razón Social	Copias	Derechos	Asociado	Cajero	Recibo	Tipo	Observaciones
10:10:16	2010-300-1-32	300-488	PRUEBAS		1	11840	2010-300-6-8	52689	42431765	ASO	IMPRESO
10:23:56	2010-300-1-33	300-490	PRUEBAS			0	ANULADO	52689	42431767	ANULADO	ANULADO
10:25:16	2010-314-1-2	314-251	PRUEBAS		1	11840		52689	42431768	INM-NAL	IMPRESO
10:26:16	2010-300-1-34	300-492	PRUEBAS		1	11840		52689	42431769	INM	IMPRESO
10:27:04	2010-300-1-35	300-493	PRUEBAS		1	11840		52689	42431770	INM	IMPRESO
11:15:50	2010-450-1-1	450-185	PRUEBAS		1	11840		52689	42431771	INM-NAL	IMPRESO
11:26:14	2010-314-1-3	314-871	PRUEBAS			0	ANULADO	52689	42431772	INM-ANULADO	ANULADO
11:27:25	2010-300-1-36	300-471	PRUEBAS		1	11810		52689	42431773	INM	IMPRESO
11:26:16	2010-314-1-4	314-112	PRUEBAS		1	11840		52689	42431774	INM-NAL	IMPRESO

**CERTIFICADOS EN OTRAS ORIPS**

Total certificados expedidos:	7
Total certificados reimpresso:	0
Total certificados anulados:	2
Total certificados otras ORIPS:	3
Derechos certificados locales:	\$ 47,360
Derechos certificados otras ORIPS:	\$ 35,520
Total derechos recaudados:	\$ 82,880

**CERTIFICADOS DE OTRAS ORIPS**

Círculo	Cantidad	Valor
PIEDECUESTA (314)	2	\$ 23,680
SAN ANDRES (450)	1	\$ 11,840
<b>Total certificados entre ORIPS'S</b>		<b>\$ 35,520</b>

**138 Reporte 93**

**REQUERIMIENTO**

Se comunica la funcionaria Katia Gonzalez e informa que el reporte 93 está notificando que la usuaria ELSY.RODRIGUEZ tiene el turno pendiente 2009-060-6-18154 y al verificar este turno los tiene el funcionario EDUARDO.DUQUE.

El reporte 93 se genera para verificar que turnos están pendientes por usuarios en las oficina de registro, por lo tanto la funcionaria Katia Gonzalez generó el reporte para verificar si la funcionaria ELSY.RODRIGUEZ tenía turnos pendientes, al momento de generar el reporte este notifica que tiene pendiente por trabajar el turno 2009-060-6- 18154, pero al consultarla en el sistema este turno está asignado al usuario EDUARDO.DUQUE, el cual se lo reasignaron para dejar libre de turnos a la funcionaria ELSY.RODRIGUEZ quien va a ser trasladada de oficina, por favor verificar el reporte ya que este está tomando el calificador de la fecha en que se repartió el turno y no está tomando el último calificador que lo tiene. Cartagena. Id Glpi 13336.

**DESARROLLO**

Se hace nueva consulta SQL del reporte reporte-93.jrxmL.

```

/*REPORTE 93 - Informe de Documentos Pendientes*/
/*GCABRERA-04/11/2009
SE MODIFICA LA CONSULTA DE ACUERDO AL CASO NÚMERO 290
DONDE REPORTAN QUE TURNOS QUE SE HAN REASIGNADO LE SIGUEN
APARECIENDO EN EL INFORME AL USUARIO INICIAL*/
SELECT tu.turmo_id_workflow Radicacion,
rep.rprt_fecha_creacion Fecha,
usu.srio_nombre_de_usuario Usuario
from sir.sir_op_turno tu inner join sir.sir_op_turno_ejecucion te on tu.turmo_id_workflow = te.trej_item_key,
sir_op_usuario usu, sir_op_reparto rep
where tu.id_circulo = $P{P_CIRCULO}
AND tu.turmo_fecha_fin IS NULL
AND TU.turmo_anulado = 'N'
and usu.id_usuario = DECODE($P{P_USUARIO}, 'TODOS', USU.id_usuario, $P{P_USUARIO})
and te.trej_fase = 'CAL_CALIFICACION'
AND te.trej_estacion = 'X-'|| usu.srio_nombre_de_usuario
AND usu.srio_activo = 1
  
```

```
and rep.anio= tu.anio and rep.id_circulo= tu.id_circulo and rep.id_proceso= tu.id_proceso and rep.id_turno= tu.id_turno  
ORDER BY usu.srio_nombre,  
tu.TRNO_FECHA_CREACION
```

## 140 Salvedades en folios migrados

### REQUERIMIENTO

Salvedades en Folios Migrados. La salvedad del folio no se está ingresando (en el proceso de migración), El numero de radicación ya que obviamente no existe, ingresando por esto null, generando diversos NullPointerExceptions por toda la aplicación cada vez que usamos uno de estos folios en algún turno.

Solución 1: Se debe generar procedimiento que actualice el número de radicación para estos folios migrados.

Solución 2: Modificar en el proceso de migrado el numero de radicación a S/N

### DESARROLLO

Se valida primero si es diferente de null para realizar la comparación.

UBICACIÓN: SIR-APP\src\gov\sir\core\negocio\acciones\correccion

AnCorr\_MesaControl.java

CÓDIGO:

```
/**  
 * @author: Guillermo Cabrera.  
 * @change: Se adiciona la condición salvedadFolio.getNumRadicacion() != null en el if para evitar que se haga el equals()  
 * cuando el NumRadicacion tenga valor nulo. ya que al realizar esta comparación se presenta un nullpointerexception.  
 * CASO MANTIS 2935  
 */  
if(salvedadFolio.getNumRadicacion() != null &&  
salvedadFolio.getNumRadicacion().equals(turno.getIdWorkflow())){  
/**  
 * @author: Guillermo Cabrera.  
 * @change: Se adiciona la condición salvedadAnotacion.getNumRadicacion() en el if para evitar que se haga el equals()  
 * cuando el NumRadicacion tenga valor nulo. ya que al realizar esta comparación se presenta un nullpointerexception.  
 * CASO MANTIS 2935**/  
if(salvedadAnotacion.getNumRadicacion() != null &&  
salvedadAnotacion.getNumRadicacion().equals(turno.getIdWorkflow())){
```

## 143 Reporte Diario Radicador de Certificados

### REQUERIMIENTO

Reporte Diario Radicador de Certificados. Se comunica Leidy Clavijo e informa que al sacar el reporte diario radicador de certificados del día 2009-11-09 el sistema no está restando un certificado anulado de otra ORIP 2009-300-1-233715.

En el consolidado del certificado de otros ORIPS de Simití se expedieron 4 (47360) certificados y el sistema está cobrando 5 (59200). Aguachica.

### DESARROLLO

Para este caso se corrige el subreporte SUBREPORTE\_OTRAS\_ORIPS\_06.jrxml para que en el cálculo no se sumen valores para turnos anulados.

Se adjunta el archivo fuente y el compilado (SUBREPORTE\_OTRAS\_ORIPS\_06.jrxml,

SUBREPORTE\_OTRAS\_ORIPS\_06.jasper)

Para que la oficina pueda generar correctamente el reporte, se debe esperar hasta que el cambio se apruebe y se suba en el despliegue.

Para la solución temporal se envía el archivo .pdf (reporte-06\_AGUACHICA.pdf).

### **145 Modificacion de Actos de Naturaleza Juridica (Informativo)**

#### ***REQUERIMIENTO***

Mediante Resolución 4906 del 30 de junio de 2009 de la SNR , se modificaron dos códigos de Actos de Naturaleza Jurídica, objetos de inscripción en las Oficinas de Registro de Instrumentos Públicos:

Codigo 0474 y 0927 los cuales deben quedar con la siguiente especificación :

CODIGO	NATURALEZA JURÍDICA
400	<b>Medidas Cautelares (Grupo)</b>
0474	“PROHIBICIÓN DE ENAJENAR DERECHOS INSCRITOS EN PREDIO DECLARADO EN ABANDONO POR CAUSA DE LA VIOLENCIA POR EL TITULAR DE ESOS DERECHOS “
900	<b>Otros (Grupo)</b>
0927	“MEDIDA DE PROTECCIÓN SOBRE PREDIO DECLARADO EN ABANDONO POR CAUSA DE LA VIOLENCIA POR POSEEDOR, OCUPANTE O TENEDOR NO INSCRITO”

Se deben efectuar las modificaciones en el Aplicativo SIR de acuerdo a la especificación de los códigos.

#### ***DESARROLLO***

Este Requerimiento se modificó satisfactoriamente por pantallas administrativas con el rol Administrador Nacional en la sección FOLIOS ANOTACIONES Y ACTOS opción Administración de Tipos de Naturalezas Jurídicas se seleccionaron los grupos de Naturaleza Jurídica:

1. Medida Cautelar
  - Código Naturaleza Jurídica 0474 del cual se modifco el nombre de Naturaleza Jurídica: “PROHIBICIÓN DE ENAJENAR DERECHOS INSCRITOS EN PREDIO DECLARADO EN ABANDONO POR CAUSA DE LA VIOLENCIA POR EL TITULAR DE ESOS DERECHOS “
2. Otro
  - Código Naturaleza Jurídica 0927 del cual se modifco el nombre de Naturaleza Jurídica: “MEDIDA DE PROTECCIÓN SOBRE PREDIO DECLARADO EN ABANDONO POR CAUSA DE LA VIOLENCIA POR POSEEDOR, OCUPANTE O TENEDOR NO INSCRITO”

**159 Reporte 25 Devoluciones**

## **REQUERIMIENTO**

### REPORTE 25 –DEVOLUCIONES DE DINERO

Se generaron los turnos de devolución de dinero 2009-300-4-417, 2009-300-4-374, 2009-300-4-399 del 28 de octubre de 2009 de la Orip de Bucaramanga, en la base de pruebas de calidad, luego se genero el reporte 25 Informe Recaudos del día por Usuario –De Elizabeth.torres, en el cual se observa un resumen Turnos de Devolución de Dinero, se listan cada uno de ellos, con la información requerida.

Sin embargo en el reporte 25, el sistema muestra error en la columna valor liquidado, las cantidades las refleja negativas, lo correcto es que se suprima el signo menos y el otro error es que en la columna Banco. No. Doc. Del Ítem Nota Debito no muestra el código del Banco, debe mostrar el código y el número de la nota debido, para ajustar esta funcionalidad la SNR gestiona formato de requerimiento para el desarrollo correspondiente.

## **DESARROLLO**

Debe aplicarse también para el reporte 51.

➤ Script Bases de Datos:

5. Se modifco una función del reporte 25 (Informe Recaudos del día por usuario) FNC\_IDBANCOSNUMDOCS\_SOLICITUD en el cual se agrego el campo SIR\_OP\_DOCUMENTO\_PAGO.DCPG\_NO\_NOTA\_DEBITO para que en la columna Banco. No. Doc. Del Ítem Nota Debito el número de la nota debido ya que el codigo del banco se estaba mostrando.
  - Se adjuntan el archivo del procedimiento actual FNC\_IDBANCOSNUMDOCS\_SOLICITUD\_Marcha Atras.sql
  - Se adjuntan el archivo del procedimiento a modificar FNC\_IDBANCOSNUMDOCS\_SOLICITUD.sql
6. Se modifco el procedimiento del reporte 51 (Resumen diario Ingresos y Egresos) SP\_Reporte\_51 en la cual se multiplican los siguientes valores por -1 para que los valores en el proceso de devoluciones (4) no se sigan desplegando negativas.

```
/*AHERRENO 30/11/2009 MODIFICACION REQ 159
LOS VALORES APlicados NO SE DEBEN DESPLEGAR
NEGATIVOS CUANDO SON DEVOLUCIONES
POR TAL MOTIVO SE MULTIPLICAN POR -1*/
efectivo.valor_aplicado :=efectivo.valor_aplicado*-1;
efectivo.valor_documento := efectivo.valor_documento ;
consignacion.valor_aplicado := consignacion.valor_aplicado *-1;
consignacion.valor_documento:= consignacion.valor_documento ;
```

```
cheque.valor_aplicado := cheque.valor_aplicado *-1;
```

**NOTA:** No se adjunta script ya que este ya se corrió en el ambiente de producción por solicitud de Javier Rocha; por lo tanto solo se aprovara en este requerimiento el despliegue del numero de la Nota Debito en la columna Banco. No. Doc. Del Ítem Nota Debito.

- Creación De Clases:

NINGUNO

- Modificación de archivos:

Se realizaron las siguientes modificaciones en los reportes 25 (Informe Recaudos del día por usuario) para que los valores en el proceso de devoluciones (4) no se sigan desplegando negativas.

```
SUBREPORTE_25_TURNOSDEVOLUCION.JASPER
/*AHERRENO 30/11/2009 MODIFICACION REQ 159
LOS VALORES APLICADOS NO SE DEBEN DESPLEGAR NEGATIVOS
CUANDO SON DEVOLUCIONES
POR TAL MOTIVO SE MULTIPLICAN POR -1*/
NVL(sir_op_liquidacion.lqdc_valor,0 )*-1 liquidacion_valor,
/**********************/
```

## 163 Reporte 74

### REQUERIMIENTO

Reporte 74. Al tratar de generar el reporte 74 informe de ingresos por certificados del año 2007 y 2008 con los siguientes parámetros fecha inicial: 1/01/2007 fecha final 31/12/2007 circulo regstral Manizales y fecha inicial 01/01/2008 fecha final 31/12/2008 circulo regstral Manizales, la página se cae y queda en blanco después de un tiempo largo de espera. Se requiere su revisión para poder generar dicho reporte. Manizales. Id Glpi 13781.

### DESARROLLO

Se realizaron pruebas para este reporte y se concluyó que el proceso es demorado a la hora de realizar los insert en la tabla SIR.SIR\_OP\_REPORTES\_74 ya que el reporte que quiere generar el usuario es de 1 año e inserta en la tabla 105255 registros.

Se optimizaron consultas del procedimiento almacenado SP\_REPORTES\_74, del subreporte SUBREPORTE\_74\_OTRAS\_ORIPSRANGOFECHAS y de la función SIR.PX\_SIR\_REPORTES.R74\_Get\_AnotIdMutaMasActual().

### CÓDIGO MODIFICADO:

#### 1. SIR.SP\_REPORTES\_74

```
/*GCABRERA 02/12/2009
SE AGREGA LA CONDICIÓN turno.id_circulo = p_circulo
CASO MANTIS 3108
SIN LA CONDICIÓN EL TIEMPO DE EJECUCIÓN ES 657.50 SEGUNDOS.
CON LA CONDICIÓN EL TIEMPO DE EJECUCIÓN ES 0.043.
CON LOS PARAMETROS FECHA INICIO '2007-01-01' FECHA FIN '2007-12-31' CIRCULO 100, LOS CUALES REPORTÓ
EL USUARIO.
```

```

*/
turno.id_circulo = p_circulo

2. WORK\reports\Fuentes
SUBREPORTE_74_OTRAS_ORIPSRANGOFECHAS.jrxml

/*SUBREPORTE_74_OTRAS_ORIPSRANGOFECHAS */
/*GCABRERA 01/12/2009
SE MODIFICA LA CONSULTA DE ACUERDO AL CASO NÚMERO 3108
YA QUE AL REVISAR LA CONSULTA EL TIEMPO DE EJECUCIÓN ES MUY ALTO 1952.023 SEGUNDOS.
AL REALIZAR LA MODIFICACION EL TIEMPO SE BAJÓ A 210.577 SEGUNDOS. */

SELECT sir_ne_circulo.CRCL_NOMBRE, sir_ne_circulo.id_circulo, count(*) cantidad,
       sum(sir_op_liquidacion.lqdc_valor) valor,
       sum(sir_op_solicitud.slct_numero_copias_cr) copias
  FROM --sir_op_solicitud_folio,
       sir_op_solicitud,
       sir_op_liquidacion,
       sir_op_turno,
       --sir_ne_folio,
       sir_ne_circulo
 WHERE sir_op_turno.id_solicitud = sir_op_solicitud.id_solicitud
   AND sir_op_solicitud.id_solicitud = sir_op_liquidacion.id_solicitud
   AND sir_op_solicitud.id_circulo = sir_ne_circulo.id_circulo
   AND sir_op_solicitud.id_circulo <> SIR_OP_TURNO.id_circulo
   AND sir_op_solicitud.id_circulo = $P{P_CIRCULO_NOMBRE}
--AND TO_CHAR(SIR_OP_TURNO.trno_fecha_inicio, 'YYYY-MM-DD') between :P_PAGO_FECHA_INI and
:P_PAGO_FECHA_FIN -- costo 64928
   AND to_char(sir_op_solicitud.slct_fecha_creacion, 'YYYY-MM-DD') between $P{P_PAGO_FECHA_INI} and
$P{P_PAGO_FECHA_FIN}
   AND SIR_OP_SOLICITUD.id_proceso = 1
   AND sir_op_turno.anio = SUBSTR($P{P_PAGO_FECHA_INI},1,4)
   AND sir_op_turno.id_proceso = 1
   AND SIR_OP_TURNO.TRNO_NACIONAL = 1
   AND sir_op_turno.trno_anulado = 'N'
--AND sir_op_solicitud.id_solicitud = sir_op_solicitud_folio.id_solicitud
--AND sir_op_solicitud.id_circulo <> sir_ne_folio.id_circulo
-- AND sir_op_solicitud_folio.id_matricula = sir_ne_folio.id_matricula
--AND sir_ne_circulo.id_circulo = sir_ne_folio.id_circulo
--AND sir_ne_folio.id_circulo <> :P_CIRCULO_NOMBRE
--AND (sir_op_turno.id_circulo=:P_CIRCULO_NOMBRE or sir_op_solicitud.id_circulo=:P_CIRCULO_NOMBRE)
 GROUP BY sir_ne_circulo.CRCL_NOMBRE, sir_ne_circulo.id_circulo

```

### 3. SIR.PX\_SIR\_REPORTS.R74\_Get\_AnotIdMutaMasActual()

```

/*GCABRERA 02/12/2009
CASO MANTIS 3108
SE COMENTA EL SELECT UTILIZADO ANTERIORMENTE,
PARA SU REMPLAZO SE CREA ESTE.
*/
SELECT *
  FROM SIR.SIR_NE_ANOTACION ANT
 WHERE ANT.NTCN_ORDEN = (SELECT max(to_number(an.NTCN_ORDEN))
                           FROM sir_ne_anotacion AN INNER JOIN SIR_NE_NATURALEZA_JURIDICA NJ
                                         ON an.id_naturaleza_juridica = nj.id_naturaleza_juridica
                           WHERE an.id_matricula = p_Folio_Id
                             and nj.idDominio_nat_jur = 'X')
 AND ANT.id_matricula = p_Folio_Id
 AND ROWNUM < 2 -- si hay mas de un ordenpad repetido

```

## 164 Reporte de Reparto (Informativo)

### **REQUERIMIENTO**

Reporte de Reparto. El usuario Lorenzo Silva nos informa que al imprimir el reparto de documentos me aparecen los turnos correspondientes a cada usuario, pero al ingresar a verificar en el sistema dichos turnos no aparecen al usuario correspondiente. Por ejemplo el turno 2009-300-6-50475 que en el reparto fisico corresponde a la usuaria GLADYS.PINILLA en el sistema aparece asignado a -SUSANA.AWAD, favor verificar

el inconveniente. Bucaramanga. Id Glpi 13789.

### **DESARROLLO**

Se verifica el reparto número 6696016 en la tabla SIR.SIR\_OP\_REPARTO y este se encuentra correctamente de acuerdo al reporte generado por la oficina, los turnos se encuentran correctamente asignados a cada uno de los usuarios.

El problema se presentó en la tabla SIR.SIR\_OP\_TURNO\_HISTORIA al momento de realizar la asignación de la estación ya que el sistema también generó relación en la fase CAL\_CALIFICACIÓN, siendo esto errado ya que esta es una fase que no genera relación.

Este caso no se está presentando a nivel general ya que es solo la ORIP de Bucaramanga la que ha reportado este tipo de error.

Se verifica el reporte-155 y este se encuentra correctamente.

## **165 Blogs de Noticias**

### **REQUERIMIENTO**

Se necesita modificar las noticias que se despliegan en el inicio del aplicativo SIR. Con el objetivo de ser actualizadas dinámicamente por el personal de la superintendencia.

### **DESARROLLO**

Se creó un Blog de noticias, el cual puede ser modificado directamente por los funcionarios de la Superintendencia de notariado y registró.

El usuario se conectara al blog a través del siguiente link:  
<http://192.168.202.26/wordpress>

Ingresara al blog a través del Usuario: snradmin con Contraseña: snradmin2010  
Nota: La contraseña puede modificarse.

Después de configurada una noticia el usuario debe enviar un correo al funcionario Javier Rocha ([javier.rocha@intek.com.co](mailto:javier.rocha@intek.com.co)) indicando el titulo de la noticia que debe publicarse para que esta se vea reflejada en la aplicación.

Las noticias mostradas al inicio de la aplicación SIR son antiguas e imposible de modificar, ya que, estaban en un blog con usuario y claves desconocidas.

Se creó un blog en la intranet que reflejara estas noticias controlando la administración de estas por parte de los funcionarios de la superintendencia.

Se modifico en el código de SIR:

```
JSP: C:\WORK\SIR-APP\web\WEB-INF\jspSeguro\comun\LOGIN.jsp
linea: 207 /**
 * @author: Fernando Padilla Velez
 * @change: Se modifica la pagina donde obtiene el xml para el resumen de las noticias mostradas.
 */
final URL url = new URL("http://192.168.150.119:8080/utilidades/rssNoticias");
```

DESPLIEGUE 26/11/2009 – 2.3.0	
INCIDENCIAS RESUELTA	
ID GLPI	OBSERVACIONES
Pantalla administración – Reportes – Listado de Reportes	
<b>Reporte 25 – Informe Recuados del dia</b>	
11349	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Jose Luis García informa que el Reporte 25 del usuario Hugo.mantilla del día 05/08/09 muestra un turno que fue anulado 2009-300-1- 144479 el día 15/07/2009, y en el día 15 no lo referencia. Por favor verificar los reportes....</p> <p>Pues el turno 2009-300-1-144479, tenía unas fases después de anulado, que fueron eliminadas por Base de datos para solucionar el inconveniente, pero de igual forma el reporte sigue mostrando el turno anulado el día 05/08/09 cuando lo anularon el 15/07/09. Bucaramanga.</p> <p><b>DESARROLLO</b></p> <p>Este requerimiento se soluciona con el requerimiento 101 Despliegue Turnos Anulados.</p>
<b>Reporte 51 – Resumen Diario de Ingresos y Egresos</b>	
	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>El turno masivo 2009-300-20-1294 fue anulado por un valor de 438080 pesos, pero al generar el reporte 51 este valor lo está sumando en el usuario de JOSE.MUNOZ en la columna de efectivo de doc.radicados. El turno masivo 2009-300-20-1294 fue anulado por un valor de 438080 pesos, pero al generar el reporte 51 este valor lo está sumando en el usuario de JOSE.MUNOZ en la columna de efectivo de doc.radicados y a su vez lo suma al total de ingresos de la oficina, favor verificar, se anexa pdf del reporte. Bucaramanga.</p> <p><b>DESARROLLO</b></p> <p>Este caso es informativo ya que su solución se realizó en menos de 12 horas y no dependía de un nuevo deploy.</p> <p>Se modificó el procedimiento del reporte 51 - Resumen Diario de Ingresos y Egresos, por favor indicarle al usuario que ingrese a la aplicación y genere de nuevo el reporte. Se agregaron las siguientes líneas al procedimiento SP_REPORTE_51</p> <pre> /*AHERRENO 24/09/2009 Se agrega condición en la cual se verifica si el turno de Certificado Masivo se encuentra anulado y valida que trhs_resuesta no sea igual a 'SIR_ROL_SUPER_DOC' en el id_turno_historia = 1*/ ELSIF t(a).trno_anulado = 'S' AND v_super=0 THEN     v_concepto := 'Documentos Anulados';     efectivo.valor_aplicado :=0;     efectivo.valor_documento :=0;     consignacion.valor_aplicado :=0;     consignacion.valor_documento :=0;     cheque.valor_aplicado :=0;     cheque.valor_documento :=0;     act.acto_cuantia :=0;     act_val_imp.acto_valor_derechos:=0;     act_val_imp.acto_valor_impuestos:=0; </pre>

	<pre> l(c).lqdc_valor_derechos_rg := 0; l(c).lqdc_valor_otro_impuesto_rg :=0; l(c).lqdc_valor_impuestos_rg := 0; </pre>	
12627	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>El 23 de Septiembre se generó un turno de devolución 2009-143-4-14 por un valor de 22,500 pero al generar el reporte 51 de esta fecha este valor lo está sumando al total. El 23 de Septiembre se generó un turno de devolución 2009-143-4-14 por un valor de 22,500 pero al generar el reporte 51 de esta fecha este valor lo está sumando al total recaudado, aparece en el usuario de CARMEN DE LA ROSA, se requiere su revisión y poder verificar el inconveniente. Cerete.</p>	
12743	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Nos informa la usuaria Ligia Estella Suarez que en el reporte 51 generado el día 29/09/2009 el sistema está sumando las devoluciones realizadas por la usuaria LIGIA.SANCHEZ. Nos informa la usuaria Ligia Estella Suarez que en el reporte 51 generado el día 29/09/2009 el sistema está sumando las devoluciones realizadas por la usuaria LIGIA.SANCHEZ, se requiere su respectiva revisión para que estos valores no sean sumados en dicho reporte. Piedecuesta.</p>	
12992	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Nos informa el usuario Fredy Hernandez que al generar el reporte 51 del día 14-10-09 el sistema está sumando en el valor aplicado \$76,880 el cual es una devolución de dinero realizada por la usuaria ROSAH.PEREZ. Por favor revisar el reporte 51 del día 14-10-09 en el valor aplicado el cual está sumando \$76,880 de una devolución realizada por la usuaria ROSAH.PEREZ. Se anexa reporte.</p>	
12753	<p><b>DESARROLLO</b></p> <p>Este requerimiento se soluciona con el requerimiento 102 Reporte 51 - Devoluciones.</p> <p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Se comunica con la mesa de ayuda Betty Ramirez por que la suma de los reportes 6 y 9 no son iguales al repote de tira caja. Por favor verificar si los valores de los reporte 6 y 9 coinciden con el tiracaja por que el usuario nos reporta que los valores no coinciden, gracias. Lorica.</p> <p><b>DESARROLLO</b></p> <p>Revisando las imágenes adjuntas del caso se detectó que si existen diferencias, en el análisis se detectó lo siguiente:</p> <ul style="list-style-type: none"> <li>*El usuario DAVID.VEGA radicó el turno 2009-146-1-11775 (Valor 11840)</li> <li>*El día 01/10/2009 se radicaron 3 consultas por tal motivo se generó el reporte 08: Diario Radicador de Consultas (Valor 3.240)</li> </ul> <p>Teniendo en cuenta esto el valor en derechos es de 1.848.460 el cual se muestra en el reporte 51 adjunto.</p>	

	<p>Reporte 25 BETTY y DAVID Derechos 1.848.460</p> <p>-----</p> <p>Reporte 51 BETTY y DAVID Derechos 1.848.460</p> <p>-----</p> <p>Reporte 06 438.080</p> <p>Reporte 08 3.240</p> <p>reporte 09 1.407.140</p> <p>-----</p> <p style="text-align: center;">1.848.460</p> <p>Se adjuntan reportes:</p> <p>Reporte 06 Lorica 01_10_2009.pdf</p> <p>Reporte 09 Lorica 01_10_2009.pdf</p> <p>Reporte 25 Betty Lorica 01_10_2009.pdf</p> <p>Reporte 25 DAVID VEGA 01_10_2009.pdf</p> <p>Reporte 51 Lorica 01_10_2009.pdf</p> <p>Reporte 08 Lorica 01_10_2009.pdf</p>	
12786	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>La usuaria Amanda nos informa que la firma del registrador está muy grande. La usuaria Amanda nos informa que la firma del registrador está muy grande, se requiere dejarla un poco más pequeña, se anexa imagen de la firma actual. Guamo.</p> <p><b>DESARROLLO</b></p> <p>Este caso es informativo ya que su solución se realizó en menos de 12 horas y no dependía de un nuevo deploy.</p> <p>Informar a la oficina que se modificó el tamaño de la firma. Se anexa imagen "Firma Guamo.JPG"</p>	
12804	<p><b>REQUERIMIENTO</b></p> <p>El sistema no genera el Reporte 90: Diario Desanotador de Certificados del día 30 de septiembre de 2009. Se intentó generar el Reporte 90 en diferentes nodos con los siguientes parámetros: Fecha Desanotación: 30/09/2009, Página: 1 Círculo Registral: VILLAVICENCIO. Pero no se genera, se queda la pantalla en blanco. Favor verificar, gracias. Villavicencio</p> <p><b>DESARROLLO</b></p> <p>Se modificó el script del reporte 99. Se adjunta reporte; verificar si el usuario lo necesita para enviarlo; y se les aclara que el cambio será reflejado en el próximo despliegue en producción. El archivo adjunto es reporte-99 villavicencio.pdf</p> <pre>  ****  /* AHERRENO 07/10/2009 Se agrega validación en la cual se verifica que el asociado no se encuentre nulo*/  AND turno_padre.trno_anulado = 'N'  **** </pre>	

## APLICACION

<b>Id de Impresion</b>	
11887	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Turno 2009-146-1-10499. Informa Bety Ramirez que al intentar reimprimir el certificado del turno 2009-146-1-10499, CER IMPRIMIR DAVID.VEGA pero no le sale, intento por imprimir pendientes pero no sale impreso. Favor enviar el id de impresion para imprimir el certificado. Lorica.</p> <p><b>DESARROLLO</b></p> <p>Adjunto envío el certificado correspondiente al id de impresión 18261940.</p>
<b>Tomar el turno en la fase de CUSTODIA PAGOS MAYORES</b>	
12207	<p><b>REQUERIMIENTO</b></p> <p>Tunro 2009-350-6-15637. Al tratar de tomar el turno en la fase de CUSTODIA PAGOS MAYORES la pantalla se queda en blanco y no carga la información del turno. Al tratar de tomar el turno en la fase de CUSTODIA PAGOS MAYORES la pantalla se queda en blanco y no carga la información del turno, se requiere verificar. Ibagué.</p> <p><b>DESARROLLO</b></p> <p>Análisis: Error en esta pantalla cuando se intenta ingresar, genera un NullPointerException al acceder al objeto LiquidacionTurnoRegistro.</p> <p>Solución: Se optimiza este caso obteniendo el objeto antes de entrar al if.</p> <p>Código COR_CUSTODIA_PAGOS_MAYORES.jsp:</p> <pre>     /**      * @author: Fernando Padilla      * @change: Se modificó la construcción de la variable, se estaba generando      NullPointerException.     */     LiquidacionTurnoRegistro objLiquidacionTurnoRegistro     (LiquidacionTurnoRegistro)iteraSolicitudLiquidacion.next();   </pre>
<b>Tomar el turno en la fase de REVISION Y ANALISIS</b>	
12401	<p><b>REQUERIMIENTO(INFORMATIVO)</b></p> <p>Turno 2009-350-3-1118. Se comunica la funcionaria Venovit Mejía e informa que al tratar de trabajar el turno 2009-350-3-1118 el sistema notifica error. El turno 2009-350-3-1118 se encuentra en la fase revisión y análisis y tiene asociado el folio 350-98708, pero al tratar de trabajar el sistema notifica como error, null. Se adjunta pantalla del error. Ibagué.</p> <p><b>DESARROLLO</b></p> <p>Existe una regla de negocio que genera una exception: "El usuario no puede tomar el bloqueo de los folios puesto que no es el usuario destino, el usuario autorizado para tomar el bloqueo es: LUZ.ESPINOSA",</p>
<b>Impresión de Formularios de Calificación</b>	
12790	<p><b>REQUERIMIENTO(INFORMATIVO)</b></p> <p>Turno 2009-260-6-26865. 2009-260-6-26865, 200-260-6-25683,</p>

	<p>2009-260-6-26859, 2009-260-6-26871, 2009-260-6-26815, 200-260-6-26851 y 2009-260-6-26925, estos turnos fueron calificados por la usuaria AURA.TRUJILLO, y no salió impreso el formulario de calificación, intentó por todos los nodos, en todos los computadores de la oficina y no salen impresos. Se requiere revisar ya que solo pasa con esta usuaria. Cucuta.</p> <p><b>DESARROLLO</b></p> <p>Adjunto los formularios en formato pdf no se presento ningún inconveniente con este usuario. Se asume que tuvo algún problema local en su oficina.</p>	
<b>Error Reparto Registro</b>		
	<p><b>REQUERIMIENTO</b></p> <p>Se comunica Luz Dary Osorno e informa que al momento de realizar el reparto notarial se cae el sistema notifica error. Se verifica remotamente procedimiento y efectivamente notifica error: No se pudo hacer el reparto: null.</p> <p>Por favor verificar error. Se adjunta imagen de error. Quibdo.</p> <p><b>DESARROLLO</b></p> <p>Solución: No existen subtipos de atención para calificadores tipo normal.</p> <p>Se creo una excepción cuando suceda esto, dejando un mensaje aclarando este error.</p> <pre>Clase: C:\WORK\SIR-APP\src\gov\sir\core\negocio\acciones\registro\ANReparto.java Método: private boolean abogadoPuedeAtender(gov.sir.core.negocio.modelo.Usuario abogado, Turno t, Map mapSubtipoUsuarios) throws EventoException{ Líneas: /*  * @author: fpadilla  * @chages: mantis 2705, no se validaba la existencia de subtipos.  */ List abogados = (List)mapSubtipoUsuarios.get(sb); if(abogados == null){     throw new EventoException("No existen subtipos de atención para calificadores del tipo: " + sr.getSubtipoAtencion().getNombre()); }</pre>	
<b>Error en la impresión del certificado</b>		
	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Folio 442-6425. En el folio 442-6425 se saco un certificado inmediato pero en la impresión sale con fecha de 14/02/1993 la que debe quedar es 15/02/1993 que es la misma del sistema. se verifica en el sistema y esta bien por favor colaborarnos si se puede cambiar esta fecha. Puerto Asís.</p> <p><b>DESARROLLO</b></p> <p>No se pudo replicar este error, se genero el certificado y se observo bien.</p> <p>Adjunto envío PDF del certificado.</p>	
<b>REQUERIMIENTOS RESUELTOS</b>		

<b>010 Entrega Masiva de documentos</b>
<p><b>REQUERIMIENTO</b> Entrega Masiva de documentos.</p> <p><b>SIR-274</b> Cuando se radican en forma masiva documentos que tienen alguna relación entre sí, porque su número de radicación es continua, a manera de ejemplo que el solicitante sea una constructora, que varios turnos con radicación continua afectan la misma matrícula, se sugiere que el aplicativo permita que el funcionario digite uno de los turnos que se pretende entregar y cuando el sistema le cargue el grupo de turnos continuos al digitado, pueda marcar todos los turnos o una parte de ellos y desanotarlos en forma masiva.</p> <p><b>SEGUIMIENTO ADA:</b> Especificar de manera mas clara el requerimiento, indicando la funcionalidad deseada.</p> <p><b>OBSERVACIONES SNR:</b> Se requiere que la funcionalidad de entrega de documentos sea modificada para que permita al usuario marcar más de un turno y luego con un solo clic, registrar la entrega en el sistema.</p> <p><b>DESARROLLO</b> La Superintendencia solicita modificar el proceso ENTREGA DOCUMENTOS dándole la posibilidad a los usuarios de entregar de forma masiva turnos en el proceso. El proceso realizado era seleccionar un turno del listado de turnos disponibles para entrega y luego realizar el proceso de actualización en la BD del estado del turno ha FINALIZADO. La solución del requerimiento fue basada en la implementación utilizada por el proceso ENTREGA DOCUMENTOS para un solo turno, para ello se procedió a realizar un clon del proceso mencionado realizando cambios para que este utilice una lista de turnos. Si el usuario selecciona solo un turno el proceso tiene el comportamiento conocido por los usuarios, en el caso que este seleccione más de un turno el proceso es modificado para que acepte una lista de turnos y ejecute un ciclo realizando la misma tarea utilizada para un solo turno con la excepción de que la información sobre el turno previa a la entrega no es visualizada. Para hacer efectivo el cambio solicitado por la Superintendencia se hizo necesario las siguientes modificaciones: Modificación Presentación: Archivos LISTAR_TURNOS.jsp y CONFIRMACION_COPIA.jsp ubicación SIR-APP\web\WEB-INF\jspSeguro\comun LISTAR_TURNOS.jsp</p> <p><b>Funciones JavaScript</b> //JALCAZAR 01/sept/2009 función encargada de guardar cada uno de los turnos seleccionados por el usuario //en diferentes páginas de turnos disponibles para entrega function TurnosSeleccion(value)</p> <p>//JALCAZAR 01/sept/2009 función encargada de la confirmación por parte del usuario del proceso function Confirmation()</p>

```
//JALCAZAR 01/sept/2009 función encargada de seleccionar el proceso a ejecutar dependiendo del numero de turnos:  
//Si el usuario selecciona un solo turno el proceso es CONTINUAR_INDIVIDUAL si selecciona mas de un turno  
ENTREGAR_REGISTRO_MULT  
function Turnosentrega()
```

### Cambio tipo INPUT

```
<!--JALCAZAR Cambio del tipo de input a type="checkbox" -->  
<input type="checkbox" name="<%-- gov.sir.core.web.acciones.comun.AWTurno.ID_TURNO %>2" value="<%--  
turno.getIdWorkflow() %>" onClick="javascript: TurnosSeleccion(this.value);"/>
```

### Cambios Botón y agrega nuevo input

```
<!--JALCAZAR agrega type="hidden" name="turnosentrega" para guardar el(los) turno(s) ha entregar -->  
<input type="hidden" name="turnosentrega" id="turnosentrega" value="<%=(request.getParameter("turnosentrega") !=null ?  
request.getParameter("turnosentrega") : "")%>">  
<!--JALCAZAR cambio del tipo input de la visualizacion del boton por img para evitar el efecto submit -->  

```

- CONFIRMACION\_COPIA.jsp

```
//JALCAZAR agrega solicita WebKeys.LISTAS_ENTREGA_MULTIPLE de la session  
List turnos = (List) session.getAttribute(WebKeys.LISTAS_ENTREGA_MULTIPLE);  
  
//JALCAZAR proceso realizado para solo un turno  
if(turnos == null){  
if(turno!=null){  
List listaUsuarios = (List)session.getAttribute(AWTurno.LISTA_USUARIOS_TURNO);  
String nombreFase = (String) session.getAttribute(AWTurno.FASE_TURNO);  
ProcesadorEventosNegocioProxy proxy = new ProcesadorEventosNegocioProxy();  
EvnTurno evento = new EvnTurno(turno.getIdWorkflow(), AWTurno.CONSULTAR_CONFIRMACION);  
try{  
EvnRespTurno respuesta = (EvnRespTurno)proxy.manejarEvento(evento);  
turno = respuesta.getTurno();  
  
session.setAttribute(WebKeys.TURNO,turno);  
}  
catch(Throwable t){  
return;  
}  
}  
}  
//JALCAZAR Mensaje de proceso depende de la utilizacion de turnos simple o multiples  
if(turnos == null)  
{  
mensaje = "La operaci&oacute;n sobre el turno "  
+ turno.getIdWorkflow()  
+ " fue realizada satisfactoriamente";  
}  
else  
{  
mensaje = "La operaci&oacute;n sobre los turnos: ";  
Iterator i = turnos.iterator();  
while (i.hasNext())  
{  
mensaje = mensaje + (String) i.next() + ", ";  
}  
mensaje = mensaje + " fue realizada satisfactoriamente";  
}
```

- WebKeys.java ubicación SIR-APP\src\gov\sir\core\web

```
/**  
 * Constante que identifica el listado de turnos en el proceso de entrega de documentos.  
 */  
public final static String LISTAS_ENTREGA_MULTIPLE = "LISTAS_ENTREGA_MULTIPLE";
```

- AWCalificacion.java ubicación SIR-APP\src\gov\sir\core\web\acciones\registro

```
/**JAlcazar 13/07/2009 Constante que indica que se realizara el proceso de entrega a mas de un turno*/  
public static final String ENTREGAR_REGISTRO_MULT = "ENTREGAR_REGISTRO_MULT";
```

```
//JALCAZAR CASO MANTIS 0002225  
else if (accion.equals(AWCalificacion.ENTREGAR_REGISTRO_MULT)) {  
return entregarRegistroMult(request);  
}
```

```

/* JALCAZAR CASO MANTIS 0002225 13/07/2009 Función Entrega Multiples Registros */
private Evento entregarRegistroMult(HttpServletRequest request) {
    HttpSession session = request.getSession();
    String TurnosEntrega = request.getParameter("turnosentrega");
    org.auriga.core.modelo.transferObjects.Usuario usuarioAuriga = (org.auriga.core.modelo.transferObjects.Usuario)
    session.getAttribute(SMARTKeys.USUARIO_EN_SESION);
    List turnos = new ArrayList();
    Fase fase = (Fase) session.getAttribute(WebKeys.FASE);
    String[] partes = TurnosEntrega.split(",");
    for (int i = 0 ;i< partes.length;i++)
    {
        turnos.add(partes[i]);
    }
    session.setAttribute(WebKeys.LISTAS_ENTREGA_MULTIPLE, turnos);
    Usuario userNeg = (gov.sir.core.negocio.modelo.Usuario)request.getSession().getAttribute(WebKeys.USUARIO);

    EvnCalificacion ev= new EvnCalificacion(usuarioAuriga, AWCalificacion.ENTREGAR_REGISTRO_MULT);
    ev.setTurnos(turnos);
    ev.setFase(fase);
    ev.setUsuarioNeg(userNeg);
    return ev;
}

```

- ControlNavegacionCalificacion.java ubicación SIR-APP\src\gov\sir\core\web\acciones\registro

```

// AVANZAR DESDE ENTREGA DE REGISTRO MULTIPLE SELECCION.
// JALCAZAR caso MANTIS 0002225
else if (accion.equals(AWCalificacion.ENTREGAR_REGISTRO_MULT)){
    return CONFIRMACION;
}

```

## 1. Modificación Eventos:

EvnCalificacion.java ubicación SIR-APP\src\gov\sir\core\eventos\registro

```

/* JALCAZAR Requerimiento MANTIS 0002225
* se agrega la variable List turnos para
* mantener los turnos en el proceso masivo de entrega
*/
private List turnos = null;
/* JALCAZAR Requerimiento MANTIS 0002225
* se agrega get y set variable List turnos para
* mantener los turnos en el proceso masivo de entrega
*/
public List getTurnos() {
    return turnos;
}
public void setTurnos(List turnos) {
    this.turnos = turnos;
}
Modificación Negocio
ANCAlificacion.java Ubicación SIR-APP\src\gov\sir\core\negocio\acciones\registro
/* JALCAZAR Requerimiento MANTIS 0002225
* se consulta si el evento es de tipo AWCalificacion.ENTREGAR_REGISTRO_MULT
* si lo es, se procede a utilizar la función entregarRegistroNuevoMult (evento)
*/
else if (evento.getTipoEvento().equals(AWCalificacion.ENTREGAR_REGISTRO_MULT))
{
    return entregarRegistroNuevoMult (evento);
}
/* JALCAZAR Requerimiento MANTIS 0002225
* función entregarRegistroNuevoMult (evento) encargada de actualizar la fase de los turnos *(multiple entrega) ha fase
FINALIZADO esto se logra modificando la función *entregarRegistroNuevo colocandolo un ciclo para pasarle cada uno de los
turnos del evento *evento.getTurnos();
*/
private EventoRespuesta entregarRegistroNuevoMult(EvnCalificacion evento) throws EventoException
{

```

## 021 Entrega Masiva de Consulta (Informativo)

### **REQUERIMIENTO**

Entrega Masiva de Consulta.

SIR-217

No funciona esta pantalla.

**SEGUIMIENTO ADA:**

Es necesario evaluar si esta pantalla realmente se necesita, o si es mejor modificar el proceso de consultas complejas de modo que al terminar la consulta, se finalice el turno. Se sugiere verificar con detenimiento el requerimiento, debido a que en el documento de simplificación, se propone la entrega de la consulta al momento de terminar dicho proceso. Simplificación que inhabilitaría la pantalla de entrega masiva.

**OBSERVACIONES SNR:**

Las consultas que tienen cobro, se finalizan cuando se finaliza el turno, verificar si las exentas también, de ser así eliminar la pantalla administrativa.

**DESARROLLO**

La funcionalidad de la pantalla administrativa OTROS -> Entrega Masiva De Consulta puede ser utilizada para los turnos de consulta exentos en el caso que el USUARIO OPERATIVO escoja la opción CONSULTA COMPLEJA O EXENTA seleccione un turno y realice una consulta, luego escoja ver resultado y luego seleccione aceptar, además esta pantalla permite realizar la entrega de más de un turno a diferencia de la opción ENTREGA CONSULTA COMPLEJA O EXENTA.

De esta manera el turno creado podrá ser entregado por la opción ENTREGA CONSULTA COMPLEJA O EXENTA por la pantalla administrativa ENTREGA MASIVA DE CONSULTA.

**043 Se requiere modificación de querys de reportes 25 y 51 Bucaramanga  
(Informativo)**

**REQUERIMIENTO**

Se requiere modificación de querys de reporte 25 y 51. ID GLPI 6890.  
Bucaramanga.

Con el caso 6823 se nos anulo el turno 2009-300-6-7953 y con el caso 6835 se nos corrigió el rep 51 el cual no descontaba el turno anulado.

En el día de hoy se revisa el turno nuevamente y aparece en cero y con un valor de \$1000 por pagar, y además cuando se vuelve a genera el tiracaja de el día 20/02/2009 de el usuario hugo.mantilla este turno no aparece ahí.

**DESARROLLO**

El reporte 25 – Informe de recaudos del día, su objetivo es mostrar los turnos de los cuales se obtuvo un recaudo. Para el turno 2009-300-6-7953 no se obtuvo ningún recaudo por eso no se despliega en el reporte. Se verificó con las siguientes consultas:

```
select *  
from sir_op_turno turno  
where turno.TRNO_ID_WORKFLOW = '2009-300-6-7953';
```

```
select *  
FROM sir_op_pago
```

```
where id_solicitud = '2009-300-6-1089';
```

#### **044 Se requiere modificación de querys de reportes 25 Piedecuesta (Informativo)**

##### **REQUERIMIENTO**

Se requiere modificación de querys de reporte 25. ID GLPI 6871. Piedecuesta. El turno 2009-314-1-6136 se encuentra anulado pero en el reporte 25 del día 24-02-09 no aparece, se escala este caso para verificación por desarrollo.

##### **DESARROLLO**

El reporte 25 – Informe de recaudos del día, su objetivo es mostrar los turnos de los cuales se obtuvo un recaudo. Para el turno 2009-314-1-6136 no se obtuvo ningún recaudo por eso no se despliega en el reporte. Se verifico con las siguientes consultas:

```
select *  
from sir_op_turno turno  
where turno.TRNO_ID_WORKFLOW = '2009-314-1-6136';
```

```
select *  
FROM sir_op_pago  
where id_solicitud = '2006-314-1-10020';
```

#### **055 Crear un rol en SIR que permita generar los reportes**

##### **REQUERIMIENTO**

Crear un rol en el aplicativo SIR que permita-generar los reportes: 1-Estadística por Actos de Registro (por oficina) actual reporte 11. 2.- Estadística por indicadores de Registro (por oficina) en un rango de fechas reportar: total folios de matricula existentes en la base de datos, folios de matricula abiertas en el rango de fecha, total radicaciones de correcciones, total de radicaciones de consulta de índices pagas, total de radicaciones de consulta de índices exentas, total de documentos radicados, total de documentos devueltos, total de documentos inscritos, total de documentos con mayor valor, total de certificados exentos, total de certificados de tradición, total de certificados de pertenencia, total de certificados expedidos supercades, total de certificados de antiguo sistema, total de certificados con mayor valor, total de devoluciones certificados, total de devolución documentos.

Con el fin de atender requerimientos internos de la SNR, es necesario que la oficina de planeación tenga un usuario en el SIR que le permita generar la información estadística por medio de los reportes solicitados.

En primera instancia, se requiere la creación de un rol llamado CONSULTA\_ESTADISTICA\_SNIR.

Asociar a él la opción de generar el reporte Estadística por Actos de Registro (por oficina) actual reporte 11.

Crear un reporte que permita en un rango de fechas, generar la información estadística de:

: total folios de matricula existentes en la base de datos, folios de matricula abiertas en el rango de fecha, total radicaciones de correcciones, total de radicaciones de consulta de índices pagas, total de radicaciones de consulta de índices exentas, total de documentos radicados, total de documentos devueltos, total de documentos

inscritos, total de documentos con mayor valor, total de certificados exentos, total de certificados de tradición, total de certificados de pertenencia, total de certificados expedidos supercades, total de certificados con mayor valor, total de devoluciones certificados, total de devolución documentos.

## **DESARROLLO**

- **Análisis de Requerimiento:** Se agrego el rol

SIR\_ROL\_CONSULTA\_ESTADISTICA\_SNRL con el siguiente script:

```
INSERT INTO sir_op_rol(id_rol, roll_nombre, roll_descripcion) VALUES (
'SIR_ROL_CONSULTA_ESTADISTICA_SNRL','ROL CONSULTA ESTADISTICA
SNRL','SIR_ROL_CONSULTA_ESTADISTICA_SNRL');
```

- Se asigna la pantalla "Listado de Reportes" al Rol

SIR\_ROL\_CONSULTA\_ESTADISTICA\_SNRL con el siguiente script:

```
INSERT INTO
SIR_OP_ROL_PANTALLA(id_rol,id_pantalla_administrativa)VALUES('SIR_ROL_CONSULTA_ESTADISTICA_SNRL',201);
```

- Se asigna la pantalla "Reporte 11 - Estadística Actos por Registro" al Rol

SIR\_ROL\_CONSULTA\_ESTADISTICA\_SNRL con el siguiente script:

```
INSERT INTO
SIR_OP_ROL_PANTALLA(id_rol,id_pantalla_administrativa)VALUES('SIR_ROL_CONSULTA_ESTADISTICA_SNRL',1010
);
```

- Configuración de la pantalla “Reporte-E01 - Estadística por Indicadores de Registro”:

```
INSERT INTO SIR_OP_PANTALLA_ADMINISTRATIVA(id_pantalla_administrativa, id_tipo_pantalla, ptdm_nombre,
ptdm_vista, ptdm_pagina)
VALUES(1113, 13, 'Reporte E01 - Estadística por Indicadores de Registro', 'admin.reportes.report-E01.param-form.view',
'PG_REPORTES');
```

- Se asigna la pantalla "Reporte E01 - Estadística por Indicadores de Registro" al Rol SIR\_ROL\_CONSULTA\_ESTADISTICA\_SNRL con el siguiente script:

```
INSERT INTO
SIR_OP_ROL_PANTALLA(id_rol,id_pantalla_administrativa)VALUES('SIR_ROL_CONSULTA_ESTADISTICA_SNRL',1113
);
```

- Por pantallas Administrativas – Administración Perfiles; se asigna el Rol SIR\_ROL\_CONSULTA\_ESTADISTICA\_SNRL al usuario

DESARROLLO.PRUEBAS

- Se creo el procedimiento SP\_Reporte\_E01

```
create or replace
PROCEDURE SP_Reporte_E01
( P_FECHA_INICIO IN VARCHAR2,
P_FECHA_FIN IN VARCHAR2,
P_CIRCULO IN VARCHAR2,
SEC_PROCESO IN NUMBER )
IS
/*****************************************************************/
/* AHERRENO 15/10/2009
  Reporte E01: Estadísticas por Indicadores de Registro*/
/*****************************************************************/
/* Total Folios de Matriculas Existentes en Base de Datos*/
/*****************************************************************/
BEGIN
BEGIN
  INSERT
  INTO SIR_OP_Reporte_E01
  (
    DESCRIPCION,
    TOTAL,
    SEC_PROCESO
  )
  SELECT 'Total Folios de Matriculas Existentes en Base de Datos' AS DESCRIPCION,
  COUNT(*) AS TOTAL,
  SEC_PROCESO AS SEC_PROCESO
  FROM sir_ne_folio f
  WHERE f.id_circulo = P_CIRCULO;
EXCEPTION
  WHEN OTHERS THEN
```

```

        dbms_output.put_line(sqlcode||sqlerrm);
END;

/*************************************************************************************************/
/* Total Folios de Matriculas Abiertas */
/*************************************************************************************************/
BEGIN
    INSERT
    INTO SIR_OP_REPORTE_E01
    (
        DESCRIPCION,
        TOTAL,
        SEC_PROCESO
    )
    SELECT 'Total Folios de Matriculas Abiertas' AS DESCRIPCION,
           COUNT(*) AS TOTAL,
           SEC_PROCESO AS SEC_PROCESO
    FROM sir_ne_folio f
    WHERE f.id_circulo = P_CIRCULO
      AND f.folio_fecha_apertura BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-dd')+1;
EXCEPTION
    WHEN OTHERS THEN
        dbms_output.put_line(sqlcode||sqlerrm);
END;

/*************************************************************************************************/
/* Total Radicaciones de Correcciones */
/*************************************************************************************************/
BEGIN
    INSERT
    INTO SIR_OP_REPORTE_E01
    (
        DESCRIPCION,
        TOTAL,
        sec_proceso
    )
    SELECT 'Total Radicaciones de Correcciones' AS DESCRIPCION,
           COUNT(*) AS TOTAL,
           SEC_PROCESO AS SEC_PROCESO
    FROM sir_op_turno t, sir_op_solicitud s
    WHERE t.id_solicitud = s.id_solicitud
      AND s.id_proceso = '3'
      AND s.id_circulo = P_CIRCULO
      AND t.trno_fecha_inicio BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-dd')+1;
EXCEPTION
    WHEN OTHERS THEN
        dbms_output.put_line(sqlcode||sqlerrm);
END;

/*************************************************************************************************/
/* Total Radicaciones de Consultas de Indices Pagadas */
/*************************************************************************************************/
BEGIN
    INSERT
    INTO SIR_OP_REPORTE_E01
    (
        DESCRIPCION,
        TOTAL,
        SEC_PROCESO
    )
    SELECT 'Total Radicaciones de Consultas de Indices Pagadas' AS DESCRIPCION,
           COUNT(*) AS TOTAL,
           SEC_PROCESO AS SEC_PROCESO
    FROM sir_op_turno t,
         sir_op_solicitud s,
         sir_op_liquidacion l,
         sir_op_pago p
    WHERE t.id_solicitud = s.id_solicitud
      AND t.trno_anulado = 'N'
      AND s.id_proceso = '2'
      AND l.id_liquidacion = 1
      --
      AND l.id_solicitud = s.id_solicitud

```

```

AND l.id_proceso = s.id_proceso
--
AND l.id_solicitud = p.id_solicitud

AND l.id_liquidacion = p.id_liquidacion
--
AND s.id_circulo = P_CIRCULO
AND p.pago_fecha BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-
dd')+1;
EXCEPTION
WHEN OTHERS THEN
    dbms_output.put_line(sqlcode||sqlerrm);
END;

/*************************************************************************************************/
/* Total Radicaciones de Consultas de Indices Exentas */
/*************************************************************************************************/
BEGIN
INSERT
INTO SIR_OP_Reporte_E01
(
DESCRIPCION,
TOTAL,
SEC_PROCESO
)
SELECT 'Total Radicaciones de Consultas de Indices Exentas' AS DESCRIPCION,
COUNT(*) AS TOTAL,
SEC_PROCESO AS SEC_PROCESO
FROM sir_op_turno t,
    sir_op_solicitud s,
    sir_op_liquidacion l,
    sir_op_pago p
WHERE t.id_solicitud = s.id_solicitud
AND t.trno_anulado = 'N'
AND s.id_proceso = '2'
AND l.id_liquidacion = 1
--
AND l.id_solicitud = s.id_solicitud
AND l.id_proceso = s.id_proceso
--
AND l.id_solicitud = p.id_solicitud
AND l.id_liquidacion = p.id_liquidacion
--
AND t.id_turno IN (SELECT th.id_turno
    FROM sir_op_turno_historia th
    WHERE l.id_proceso = th.id_proceso
    AND t.anio = th.anio
    AND s.id_circulo = th.id_circulo
    AND th.trhs_respuesta = 'CONSULTA_EXENTA')
--
AND s.id_circulo = P_CIRCULO
AND p.pago_fecha BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-
dd')+1;
EXCEPTION
WHEN OTHERS THEN
    dbms_output.put_line(sqlcode||sqlerrm);
END;

/*************************************************************************************************/
/* Total Documentos Radicados */
/*************************************************************************************************/
BEGIN
INSERT
INTO SIR_OP_Reporte_E01
(
DESCRIPCION,
TOTAL,
SEC_PROCESO
)
SELECT 'Total Documentos Radicados' AS DESCRIPCION,
COUNT(*) AS TOTAL,
SEC_PROCESO AS SEC_PROCESO
FROM sir_op_turno t,
    sir_op_solicitud s,
    sir_op_liquidacion l,

```

```

    sir_op_pago p
WHERE t.id_solicitud = s.id_solicitud
AND t.trmo_anulado = 'N'
AND s.id_proceso = '6'
AND l.id_liquidacion = 1
--
AND l.id_solicitud = s.id_solicitud
AND l.id_proceso = s.id_proceso
--
AND l.id_solicitud = p.id_solicitud
AND l.id_liquidacion = p.id_liquidacion
--
AND s.id_circulo = P_CIRCULO
AND p.pago_fecha BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-
dd')+1;
EXCEPTION
WHEN OTHERS THEN
dbms_output.put_line(sqlcode||sqlerrm);
END;

/*************************************************************************************************/
/* Total Documentos Devueltos */
/*************************************************************************************************/
BEGIN
INSERT
INTO SIR_OP_REPORTE_E01
(
DESCRIPCION,
TOTAL,
sec_proceso
)
SELECT 'Total Documentos Devueltos' AS DESCRIPCION,
COUNT(*) AS TOTAL,
SEC_PROCESO AS SEC_PROCESO
FROM sir_op_turno t,
sir_op_solicitud s,
sir_op_liquidacion l,
sir_op_pago p
WHERE t.id_solicitud = s.id_solicitud
AND t.trmo_anulado = 'N'
AND s.id_proceso = '6'
AND l.id_liquidacion = 1
--
AND l.id_solicitud = s.id_solicitud
AND l.id_proceso = s.id_proceso
--
AND l.id_solicitud = p.id_solicitud
AND l.id_liquidacion = p.id_liquidacion
--
AND t.id_turno IN (SELECT th.id_turno
FROM sir_op_turno_historia th
WHERE l.id_proceso = th.id_proceso
AND t.anio = th.anio
AND s.id_circulo = th.id_circulo
AND th.trhs_respuesta = 'DEVOLUCION')
--
AND s.id_circulo = P_CIRCULO
AND p.pago_fecha BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-
dd')+1;
EXCEPTION
WHEN OTHERS THEN
dbms_output.put_line(sqlcode||sqlerrm);
END;

/*************************************************************************************************/
/* Total Documentos Inscritos */
/*************************************************************************************************/
BEGIN
INSERT
INTO SIR_OP_REPORTE_E01
(
DESCRIPCION,
TOTAL,
SEC_PROCESO
)

```

```

SELECT 'Total Documentos Inscritos' AS DESCRIPCION,
       COUNT(*) AS TOTAL,
       SEC_PROCESO AS SEC_PROCESO
  FROM sir_op_turno t,
       sir_op_solicitud s,
       sir_op_liquidacion l,
       sir_op_pago p
 WHERE t.id_solicitud = s.id_solicitud
   AND t.trno_anulado = 'N'
   AND s.id_proceso = '6'
   AND l.id_liquidacion = 1
   --
   AND l.id_solicitud = s.id_solicitud
   AND l.id_proceso = s.id_proceso
   --
   AND l.id_solicitud = p.id_solicitud
   AND l.id_liquidacion = p.id_liquidacion
   --
   AND t.id_turno IN (SELECT th.id_turno
                      FROM sir_op_turno_historia th
                     WHERE l.id_proceso      = th.id_proceso
                       AND t.anio          = th.anio
                       AND s.id_circulo    = th.id_circulo
                       AND th.trhs_id_fase = 'CAL_CALIFICACION'
                       AND th.trhs_respuesta = 'OK')
   --
   AND s.id_circulo = P_CIRCULO
   AND p.pago_fecha BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-
dd')+1;
  EXCEPTION
  WHEN OTHERS THEN
    dbms_output.put_line(sqlcode||sqlerrm);
END;

/*****************/
/* Total Documentos Mayor Valor */
/*****************/
BEGIN
  INSERT
  INTO SIR_OP_Reporte_E01
  (
    DESCRIPCION,
    TOTAL,
    SEC_PROCESO
  )
  SELECT 'Total Documentos Mayor Valor' AS DESCRIPCION,
         COUNT(*) AS TOTAL,
         SEC_PROCESO AS SEC_PROCESO
  FROM sir_op_turno t,
       sir_op_solicitud s,
       sir_op_liquidacion l,
       sir_op_pago p
 WHERE t.id_solicitud = s.id_solicitud
   AND t.trno_anulado = 'N'
   AND s.id_proceso = '6'
   AND l.id_liquidacion = 1
   --
   AND l.id_solicitud = s.id_solicitud
   AND l.id_proceso = s.id_proceso
   --
   AND l.id_solicitud = p.id_solicitud
   AND l.id_liquidacion = p.id_liquidacion
   --
   AND t.id_turno IN (SELECT th.id_turno
                      FROM sir_op_turno_historia th
                     WHERE l.id_proceso      = th.id_proceso
                       AND t.anio          = th.anio
                       AND s.id_circulo    = th.id_circulo
                       AND th.trhs_respuesta = 'MAYOR_VALOR')
   --
   AND s.id_circulo = P_CIRCULO
   AND p.pago_fecha BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-
dd')+1;
  EXCEPTION
  WHEN OTHERS THEN

```

```

        dbms_output.put_line(sqlcode||sqlerrm);
END;

/*************************************************************************************************/
/* Total Certificados Exentos */
/*************************************************************************************************/
BEGIN
    INSERT
    INTO SIR_OP_Reporte_E01
    (
        DESCRIPCION,
        TOTAL,
        SEC_PROCESO
    )
    SELECT 'Total Certificados Exentos' AS DESCRIPCION,
        COUNT(*) AS TOTAL,
        SEC_PROCESO AS SEC_PROCESO
    FROM sir_op_turno t,
        sir_op_solicitud s,
        sir_op_liquidacion l,
        sir_op_pago p
    WHERE t.id_solicitud = s.id_solicitud
    AND t.tmo_anulado = 'N'
    AND s.id_proceso = 'I'
    AND l.id_liquidacion = 1
    AND l.lqdc_tipo_tarifa_cr = 'EXENTO'
    --
    AND l.id_solicitud = s.id_solicitud
    AND l.id_proceso = s.id_proceso
    --
    AND l.id_solicitud = p.id_solicitud
    AND l.id_liquidacion = p.id_liquidacion
    --
    AND s.id_circulo = P_CIRCULO
    AND p.pago_fecha BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-
dd')+1;
    EXCEPTION
    WHEN OTHERS THEN
        dbms_output.put_line(sqlcode||sqlerrm);
END;

/*************************************************************************************************/
/* Total Certificados de Tradición */
/*************************************************************************************************/
BEGIN
    INSERT
    INTO SIR_OP_Reporte_E01
    (
        DESCRIPCION,
        TOTAL,
        SEC_PROCESO
    )
    SELECT 'Total Certificados de Tradición' AS DESCRIPCION,
        COUNT(*) AS TOTAL,
        SEC_PROCESO AS SEC_PROCESO
    FROM sir_op_turno t,
        sir_op_solicitud s,
        sir_op_liquidacion l,
        sir_op_pago p
    WHERE t.id_solicitud = s.id_solicitud
    AND t.tmo_anulado = 'N'
    AND s.id_proceso = 'I'
    AND l.id_liquidacion = 1
    --
    AND l.id_solicitud = s.id_solicitud
    AND l.id_proceso = s.id_proceso
    --
    AND l.id_solicitud = p.id_solicitud
    AND l.id_liquidacion = p.id_liquidacion
    --
    AND s.id_circulo = P_CIRCULO
    AND p.pago_fecha BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-
dd')+1;
    EXCEPTION
    WHEN OTHERS THEN

```

```

        dbms_output.put_line(sqlcode||sqlerrm);
END;

/*************************************************************************************************/
/* Total Certificados de Pertenencia */
/*************************************************************************************************/
BEGIN
    INSERT
    INTO SIR_OP_Reporte_E01
    (
        DESCRIPCION,
        TOTAL,
        SEC_PROCESO
    )
    SELECT 'Total Certificados de Pertenencia' AS DESCRIPCION,
        COUNT(*) AS TOTAL,
        SEC_PROCESO AS SEC_PROCESO
    FROM sir_op_turno t,
        sir_op_solicitud s,
        sir_op_liquidacion l,
        sir_op_pago p
    WHERE t.id_solicitud = s.id_solicitud
    AND t.trno_anulado = 'N'
    AND s.id_proceso = 'I'
    AND l.id_liquidacion = 1
    --
    AND l.id_solicitud = s.id_solicitud
    AND l.id_proceso = s.id_proceso
    --
    AND l.id_solicitud = p.id_solicitud
    AND l.id_liquidacion = p.id_liquidacion
    --
    AND t.id_turno IN (SELECT th.id_turno
        FROM sir_op_turno_historia th
        WHERE l.id_proceso      = th.id_proceso
        AND t.anio            = th.anio
        AND s.id_circulo       = th.id_circulo
        AND th.trhs_responsable = 'PERTENENCIA')
    --
    AND s.id_circulo = P_CIRCULO
    AND p.pago_fecha BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-
dd')+1;
EXCEPTION
WHEN OTHERS THEN
    dbms_output.put_line(sqlcode||sqlerrm);
END;

/*************************************************************************************************/
/* Total Certificados con Mayor Valor */
/*************************************************************************************************/
BEGIN
    INSERT
    INTO SIR_OP_Reporte_E01
    (
        DESCRIPCION,
        TOTAL,
        SEC_PROCESO
    )
    SELECT 'Total Certificados con Mayor Valor' AS DESCRIPCION,
        COUNT(*) AS TOTAL,
        SEC_PROCESO AS SEC_PROCESO
    FROM sir_op_turno t,
        sir_op_solicitud s,
        sir_op_liquidacion l,
        sir_op_pago p
    WHERE t.id_solicitud = s.id_solicitud
    AND t.trno_anulado = 'N'
    AND s.id_proceso = 'I'
    AND l.id_liquidacion = 1
    --
    AND l.id_solicitud = s.id_solicitud
    AND l.id_proceso = s.id_proceso
    --
    AND l.id_solicitud = p.id_solicitud
    AND l.id_liquidacion = p.id_liquidacion

```

```

-- AND t.id_turno IN (SELECT th.id_turno
--                      FROM sir_op_turno_historia th
--                     WHERE l.id_proceso      = th.id_proceso
--                       AND t.anio          = th.anio
--                       AND s.id_circulo    = th.id_circulo
--                       AND th.trhs_respuesta = 'MAYOR_VALOR')
--
-- AND s.id_circulo = P_CIRCULO
-- AND p.pago_fecha BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-
dd')+1;
EXCEPTION
WHEN OTHERS THEN
  dbms_output.put_line(sqlcode||sqlerrm);
END;

/*****************/
/* Total Devoluciones Documentos */
/*****************/
BEGIN
  INSERT
  INTO SIR_OP_Reporte_E01
  (
  DESCRIPCION,
  TOTAL,
  SEC_PROCESO
  )
  SELECT 'Total Devoluciones Documentos' AS DESCRIPCION,
  COUNT(*) AS TOTAL,
  SEC_PROCESO AS SEC_PROCESO
  FROM sir_op_turno t,
  sir_op_solicitud s,
  sir_op_liquidacion l,
  sir_op_pago p
  WHERE t.id_solicitud = s.id_solicitud
  AND t.trno_anulado = 'N'
  AND s.id_proceso = '6'
  AND l.id_liquidacion = 1
  --
  AND l.id_solicitud = s.id_solicitud
  AND l.id_proceso = s.id_proceso
  --
  AND l.id_solicitud = p.id_solicitud
  AND l.id_liquidacion = p.id_liquidacion
  --
  AND t.id_turno IN (SELECT os.id_turno_ant
                      FROM sir_op_solicitud os
                     WHERE os.id_proceso_ant = s.id_proceso
                       AND os.id_circulo_ant = s.id_circulo
                       AND os.anio_ant       = t.anio
                       AND os.id_proceso     = 4)
  --
  AND s.id_circulo = P_CIRCULO
  AND p.pago_fecha BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-
dd')+1;
EXCEPTION
WHEN OTHERS THEN
  dbms_output.put_line(sqlcode||sqlerrm);
END;

/*****************/
/* Total Devoluciones Certificados */
/*****************/
BEGIN
  INSERT
  INTO SIR_OP_Reporte_E01
  (
  DESCRIPCION,
  TOTAL,
  SEC_PROCESO
  )
  SELECT 'Total Devoluciones Certificados' AS DESCRIPCION,
  COUNT(*) AS TOTAL,
  SEC_PROCESO AS SEC_PROCESO
  FROM sir_op_turno t,

```

```

        sir_op_solicitud s,
        sir_op_liquidacion l,
        sir_op_pago p
    WHERE t.id_solicitud = s.id_solicitud
    AND t.trmo_anulado = 'N'
    AND s.id_proceso = 'I'
    AND l.id_liquidacion = 1
    --
    AND l.id_solicitud = s.id_solicitud
    AND l.id_proceso = s.id_proceso
    --
    AND l.id_solicitud = p.id_solicitud
    AND l.id_liquidacion = p.id_liquidacion
    --
    AND t.id_turno IN (SELECT os.id_turno_ant
                        FROM sir_op_solicitud os
                        WHERE os.id_proceso_ant = s.id_proceso
                        AND os.id_circulo_ant = s.id_circulo
                        AND os.anio_ant      = t.anio
                        AND os.id_proceso     = 4)
    --
    AND s.id_circulo = P_CIRCULO
    AND p.pago_fecha BETWEEN to_date(P_FECHA_INICIO, 'yyyy-mm-dd') AND to_date(P_FECHA_FIN, 'yyyy-mm-
dd')+1;
    EXCEPTION
    WHEN OTHERS THEN
        dbms_output.put_line(sqlcode||sqlerrm);
END;
END;

```

- Se creo la secuencia SEQ\_Reporte\_E01

```

CREATE SEQUENCE "SIR"."SEQ_Reporte_E01" MINVALUE 1 MAXVALUE 9999999999999999999999999999
INCREMENT BY 1 START WITH 1 CACHE 2 NOORDER NOCYCLE ;

```

- Creación Tabla SIR\_OP\_Reporte\_E01

```

CREATE TABLE "SIR"."SIR_OP_Reporte_E01"
(
    "DESCRIPCION" VARCHAR2(100 BYTE),
    "TOTAL" NUMBER,
    "SEC_PROCESO" NUMBER
) PCTFREE 10 PCTUSED 40 INITTRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "TBS_OWB_RUNTIME" ;

```

Configuración Reporte E01 – Estadística por Indicadores de Registro	
SIR-APP\web\WEB-INF\classes\org.auriga.smart.vistas.xml	
<!-- AHERRENO 22/09/2009 report-E01 -->	
<vista>	
<autenticacion>true</autenticacion>	
<expcion>gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector</expcion>	
<expcion>gov.sir.core.web.acciones.excepciones.ValidacionParametrosException</expcion>	
<nombre>admin.reportes.report-E01.param-form.view</nombre>	
<parametro><llave>error</llave><valor>/WEB-INF/jspSeguro/comun/ERROR_MULTIPLE.jsp</valor></parametro>	
<parametro><llave>head</llave><valor>/WEB-INF/jspSeguro/comun/HEADER_LOGOUT.jsp</valor></parametro>	
<parametro><llave>body</llave><valor>/WEB-INF/jspSeguro/administracion/ADMINREPORTES_REPORTE01_PARAMFORM.jsp</valor></parametro>	
<parametro><llave>bottom</llave><valor>/WEB-INF/jspSeguro/comun/FOOTER_DEFAULT.jsp</valor></parametro>	
</vista>	
SIR-APP\web\WEB-INF\classes\org.auriga.smart.web.xml	
<!-- AHERRENO 22/09/2009 report-E01 -->	
<navegacion>	
<resultado>REPORTE_E01_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK</resultado>	
<url-redirect>admin.reportes.report-E01.param-form.view</url-redirect>	
</navegacion>	
SIR-APP\web\WEB-INF\jspSeguro\ADMINREPORTES_REPORTE01_PARAMFORM	
Archivo Nuevo.	
SIR-APP\src\gov\sir\core\web\acciones\administracion\AwReportes.java	
// AHERRENO 22/09/2009	
// Reporte E01	
public static final String REPORTE_E01_SEND_ACTION = "REPORTE_E01_SEND_ACTION";	
public static final String REPORTE_E01_BACK_ACTION = "REPORTE_E01_BACK_ACTION";	

```

// AHERRENO 22/09/2009
// Reporte E01
public static final String REPORTE_E01_LOADCIRCULOSBYUSUARIO =
"REPORTE_E01_LOADCIRCULOSBYUSUARIO";
// AHERRENO 22/09/2009
// Declaración de las Variables para el reporte E01
public static final String REPORTE_E01_PARAM_CMDKEY = "reporte-E01.jasper";
public static final String REPORTE_E01_NOMBRE = "";
public static final String REPORTE_E01_PARAM_PFECHAINI = "V_FECHAINI";
public static final String REPORTE_E01_PARAM_PFECHAFIN = "V_FECHAFIN";
public static final String REPORTE_E01_PARAM_PCIRCULONOMBRE = "V_CIRCULO";
//AHERRENO 22/09/2009
//Reporte E01
if (REPORTE_E01_LOADCIRCULOSBYUSUARIO.equals(accion)) {
    return do_ReporteXX_LoadCirculosByUsuario_Action(request);
}
// AHERRENO 22/09/2009
// Reporte E01
if (REPORTE_E01_SEND_ACTION.equals(accion)) {
    do_ReporteE01_Send_Action(request);
}
if (REPORTE_E01_BACK_ACTION.equals(accion)) {
    do_ReporteE01_Back_Action(request);
}
//AHERRENO 22/09/2009
//Reporte E01
private void do_ReporteE01_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {
    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();
    String param_01 = request.getParameter(REPORTE_E01_PARAM_PFECHAINI);
    String param_02 = request.getParameter(REPORTE_E01_PARAM_PFECHAFIN);
    String param_03 = request.getParameter(REPORTE_E01_PARAM_PCIRCULONOMBRE);
    // apply validators
    if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {
        exception.addError("parametro: Fecha Inicio Radicación: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_01);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_01 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Inicio Radicación: La fecha es inválida" + e.getMessage());
        }
    }
    // apply validators
    if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
        exception.addError("parametro: Fecha Fin Radicación: Debe digitar un valor");
    } else {
        String send1 = null;
        Date fecha = null;
        try {
            fecha = DateFormatUtil.parse(param_02);
            send1 = DateFormatUtil.format("yyyy-MM-dd", fecha);
            param_02 = send1;
        } catch (Exception e) {
            exception.addError("parametro: Fecha Fin Radicación: La fecha es inválida" + e.getMessage());
        }
    }
    // apply validators
    if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    } else {
        if (WebKeys.SIN_SELECCIONAR.equals(param_03)) {
            exception.addError("parametro: Circulo: Debe digitar un valor");
        }
    }
    do_ReporteE01_SaveState(request);
// raise -----
if (exception.getErrores().size() > 0) {
    throw exception;
}

```

```

// -----
String[] params = new String[]{ 
    REPORTE_XX_PARAM_CMDKEY, REPORTE_E01_PARAM_PFECHAINI,
    REPORTE_E01_PARAM_PFECHAFIN, REPORTE_E01_PARAM_PCIRCLONOMBRE
};
String[] values = new String[]{ 
    REPORTE_E01_PARAM_CMDKEY, param_01, param_02, param_03
};
String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);
do_ReporteE01_RemoveState(request);
do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
}
private void do_ReporteE01_Back_Action(HttpServletRequest request) {
    do_ReporteE01_RemoveState(request);
    do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);
} // :do_ReporteE01_Back_Action
private void do_ReporteE01_SaveState(HttpServletRequest request) {
    HttpSession session;
    String[] itemIds = new String[]{ 
        REPORTE_E01_PARAM_PFECHAINI, REPORTE_E01_PARAM_PFECHAFIN,
        REPORTE_E01_PARAM_PCIRCLONOMBRE
    };
    session = request.getSession();
    save_PageItemsState(itemIds, request, session);
} // :do_ReporteE01_SaveState
private void do_ReporteE01_RemoveState(HttpServletRequest request) {
    HttpSession session;
    String[] itemIds = new String[]{ 
        REPORTE_E01_PARAM_PFECHAINI, REPORTE_E01_PARAM_PFECHAFIN,
        REPORTE_E01_PARAM_PCIRCLONOMBRE
    };
    session = request.getSession();
    delete_PageItemsState(itemIds, request, session);
} // :do_ReporteE01_RemoveState

```

#### JASPER – reporte-E01.jasper

##### Archivo Nuevo.

##### SIR-APP\src\gov\sir\core\web\navegacion\administracion\ControlNavegacionReportes.java

```

//AHERRENO 22/09/2009
//Reporte E01
public static final String REPORTE_E01_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK =
    "REPORTE_E01_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK";

//AHERRENO 22/09/2009
//Reporte E01
public static final String REPORTE_E01_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK =
    "REPORTE_E01_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK";

//AHERRENO 22/09/2009
//Reporte E01
listaReportesJasper.add(AWReportes.REPORTE_E01_SEND_ACTION);

//AHERRENO 22/09/2009
// -- navegacion sobre pagina
if( AWReportes.REPORTE_E01_LOADCIRCULOSBYUSUARIO.equals( accion ) ){
    return REPORTE_E01_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK;
}

//AHERRENO 22/09/2009
//Reporte E01
if( AWReportes.REPORTE_E01_BACK_ACTION.equals( accion ) ) {
    return REPORTE_XX_BACK_EVENTRESP_OK;
}

```

##### SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDORreportesJasperDAO.java

```

//AHERRENO 22/09/2009
//Reporte E01
//Se obtienen los parametros para enviar al procedimiento
listaParametros = new ArrayList();
listaParametros.add(AWReportes.REPORTE_E01_PARAM_PFECHAINI);
listaParametros.add(AWReportes.REPORTE_E01_PARAM_PFECHAFIN);
listaParametros.add(AWReportes.REPORTE_E01_PARAM_PCIRCLONOMBRE);
mapaReportes.put(AWReportes.REPORTE_E01_PARAM_CMDKEY, new Object[]{
    "SELECT seq_reporte_E01.NEXTVAL FROM dual",
});

```

<pre>"sp_reporte_E01(?, ?, ?, ?)", listaParametros, "DELETE FROM sir_op_reporte_E01 WHERE sec_proceso = "});</pre>	
<b>SIR-APP\src\org.auriga.smart.vistas.xml</b>	
<pre>&lt;!-- AHERRENO 22/09/2009 report-E01 --&gt; &lt;vista&gt; &lt;autenticacion&gt;true&lt;/autenticacion&gt; &lt;excepcion&gt;gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector&lt;/excepcion&gt; &lt;excepcion&gt;gov.sir.core.web.acciones.excepciones.ValidacionParametrosException&lt;/excepcion&gt; &lt;nombre&gt;admin.reportes.report-E01.param-form.view&lt;/nombre&gt; &lt;parametro&gt;&lt;llave&gt;error&lt;/llave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/ERROR_MULTIPLE.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;parametro&gt;&lt;llave&gt;head&lt;/llave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/HEADER_LOGOUT.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;parametro&gt;&lt;llave&gt;body&lt;/llave&gt;&lt;valor&gt;/WEB- INF/jspSeguro/administracion/ADMINREPORTES_REPORTE01_PARAMFORM.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;parametro&gt;&lt;llave&gt;bottom&lt;/llave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/FOOTER_DEFAULT.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;/vista&gt;</pre>	
<b>SIR-APP\src\org.auriga.smart.web.xml</b>	
<pre>&lt;!-- AHERRENO 22/09/2009 report-E01 --&gt; &lt;navegacion&gt; &lt;resultado&gt;REPORTE_E01_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK&lt;/resultado&gt; &lt;url-redireccion&gt;admin.reportes.report-E01.param-form.view&lt;/url-redireccion&gt; &lt;/navegacion&gt;</pre>	
<b>SIR-APP\web\WEB-INF\jspseguro\administracion\old.REPORT_LIST</b>	
<pre>&lt;%-- + + + + + + + -%&gt; &lt;%-- AHERRENO 22/09/2009 --%&gt; &lt;%-- report_E01 --%&gt; &lt;tr&gt; &lt;td&gt;&amp;nbsp;&lt;/td&gt; &lt;td onMouseOver="this.style.backgroundColor="#F7F9FD"" onMouseOut="this.style.backgroundColor=""&gt;&lt;a href="admin.reportes.report-E01.param-form.view" class="links"&gt;Reporte E01: Estad&amp;iacute;sticas por Indicadores de Registro&lt;/a&gt; &lt;/td&gt; &lt;td&gt; &lt;% legendHelper.setMessage( "" ); legendHelper.setRegionId( "reportE01" ); legendHelper.setRegionName( ": Estad&amp;iacute;sticas por Indicadores de Registro" ); try { legendHelper.render( out ); } catch( java.io.IOException e ){ out.write( e.toString() ); } %&gt; &lt;/td&gt; &lt;td&gt;&amp;nbsp;&lt;/td&gt; &lt;td onMouseOver="this.style.backgroundColor="#F7F9FD"" onMouseOut="this.style.backgroundColor=""&gt;&amp;nbsp;&lt;/td&gt; &lt;/tr&gt;</pre>	
<b>mapas.jar\ org.auriga.smart.vistas.xml</b>	
<pre>&lt;!-- AHERRENO 22/09/2009 report-E01 --&gt; &lt;vista&gt; &lt;autenticacion&gt;true&lt;/autenticacion&gt; &lt;excepcion&gt;gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector&lt;/excepcion&gt; &lt;excepcion&gt;gov.sir.core.web.acciones.excepciones.ValidacionParametrosException&lt;/excepcion&gt; &lt;nombre&gt;admin.reportes.report-E01.param-form.view&lt;/nombre&gt; &lt;parametro&gt;&lt;llave&gt;error&lt;/llave&gt;&lt;valor&gt;/WEB- INF/jspSeguro/comun/ERROR_MULTIPLE.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;parametro&gt;&lt;llave&gt;head&lt;/llave&gt;&lt;valor&gt;/WEB- INF/jspSeguro/comun/HEADER_LOGOUT.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;parametro&gt;&lt;llave&gt;body&lt;/llave&gt;&lt;valor&gt;/WEB- INF/jspSeguro/administracion/ADMINREPORTES_REPORTE01_PARAMFORM.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;parametro&gt;&lt;llave&gt;bottom&lt;/llave&gt;&lt;valor&gt;/WEB- INF/jspSeguro/comun/FOOTER_DEFAULT.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;/vista&gt;</pre>	
<b>mapas.jar\ org.auriga.smart.web.xml</b>	
<pre>&lt;!-- AHERRENO 22/09/2009 report-E01 --&gt; &lt;navegacion&gt; &lt;resultado&gt;REPORTE_E01_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK&lt;/resultado&gt; &lt;url-redireccion&gt;admin.reportes.report-E01.param-form.view&lt;/url-redireccion&gt; &lt;/navegacion&gt;</pre>	

### **REQUERIMIENTO**

En el diario radicador de consultas (Reporte 08) cuando se existen turnos anulados del día, el aplicativo suma los valores de estos turnos en el total de la columna de derechos. Esta inconsistencia debe ajustarse toda vez que el reporte no refleja la información correcta.

Lo correcto es que el valor del turno anulado en la columna de recaudo se refleje en 0 y seguido de la columna solicitante se debe indicar “anulado”.

### **DESARROLLO**

Se agrego las siguientes validaciones al script del reporte:

```
***** AHERRENO 08/09/2009 *****/  
/* Se valida si el turno se anulo */  
DECODE (anulado, 'N', nombre_ciudadano, 'ANULADO - '||nombre_ciudadano ) AS "NOMBRE CIUDADANO",  
DECODE (anulado, 'N', recaudo, 0 ) AS recaudo,  
*****
```

Con esta validación se desplegará en la columna solicitante "ANULADO - Nombre del Solicitante" y en la columna Recaudo el valor \$0. este cambio será reflejado en el próximo despliegue.

## 067 Agregar Salvedad Crear Anotación

### **REQUERIMIENTO**

Cuando se realiza una modificación o cambio en un folio de matrícula o se incluya anotaciones en el proceso de correcciones el sistema debe obligar agregar una salvedad, no debe dejar avanzar el turno a fases posteriores sin que el auxiliar de correcciones se la incluya

### **DESARROLLO**

En el proceso de crear una nueva anotación en corrección de un folio, esta no valida que se haya escrito en el campo salvedad.

Solución: Se agrego al proceso de validación de esta pantalla el campo salvedad, para que no se puedan crear anotaciones en el proceso de corrección sin esta salvedad.

Clase:

C:\WORK\SIR-  
APP\src\gov\sir\core\web\acciones\correccion\AWModificarFolio.java

Método:

```
private EvnCorreccion agregarAnotacion(HttpServletRequest request) throws AccionWebException
```

Código linea 10307:

```
if (salvedad == null || salvedad.trim().equals("")) {  
    exception.addError("anotacion:salvedad; debe escribir un valor en la salvedad");  
}
```

## 069 Anulación Turnos Exentos

### **REQUERIMIENTO**

Se requiere modificar el aplicativo SIR en cuanto a la funcionalidad de Anular turnos, para permitir que los turnos de documentos que tiene radicados actos exentos, puedan anularse. Igualmente, debe permitir anular los turnos de certificados exentos asociados, al turno de documento que tiene actos exentos. Para los demás casos debe

mantenerse la regla de no permitir anular turnos exentos.

Esta solicitud se hace teniendo en cuenta el alto volumen de casos reportados a mesa de ayuda y que son solucionados puntualmente modificando la base de datos como solución al mismo.

### **DESARROLLO**

En el proceso de anular turnos de registro se valida si este turno posee actos exentos y no los deja anular.

Se quito esta validación para poder anular estos turnos.

Clase:

C:\WORK\SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDOTurnosDAO.java

Método:

public boolean validarAnulacionTurno(TurnoPk idTurno) throws DAOException

Código linea 4758:

```
/**  
 * @author: Fernando Padilla  
 * @change: Mantis 2096: Acta - Requerimiento No 069 - Anulacion_Turnos_Exentos,  
 * por lo cual se deshabilita la validación de existencia de actos de tipo exento  
 * en el turno de registro.  
 */  
/*SolicitudRegistroEnhanced solReg = (SolicitudRegistroEnhanced) tr.getSolicitud();  
List liquidaciones = solReg.getLiquidaciones();  
Iterator itliquidaciones = liquidaciones.iterator();  
while (itliquidaciones.hasNext()) {  
    LiquidacionTurnoRegistroEnhanced lTReg = (LiquidacionTurnoRegistroEnhanced) itliquidaciones.next();  
    List actos = lTReg.getActos();  
    Iterator itactos = actos.iterator();  
    while (itactos.hasNext()) {  
        ActoEnhanced acto = (ActoEnhanced) itactos.next();  
        TipoActoDerechoRegistralEnhanced tadr = (TipoActoDerechoRegistralEnhanced) acto.getTipoDerechoReg();  
        TipoDerechoRegEnhanced tdr = (TipoDerechoRegEnhanced) tadr.getTipoDerechoRegistral();  
        if(tdr.getIdTipoDerechoReg().equals(CTipoTarifa.IMPUESTO_EXENTO)){  
            throw new JDOException ("El turno tiene una Acto de tipo EXENTO.");  
        }  
    }*/
```

## **072 Liberar consignaciones turnos anulados**

### **REQUERIMIENTO**

Liberar consignaciones utilizadas en turnos anulados. Cuando un turno es pagado en consignación y/o cheque y este luego es anulado al tratar de ingresar nuevamente la consignación y/o cheque el sistema indica que es ya fueron utilizados con otro turno.

Se requiere que este caso sea escalado a desarrollo para que al anular un turno y este halla sido en consignación y/o cheque queden liberados para poder utilizarlos nuevamente en otro pago de un turno nuevo validando que no hayan sido utilizados en un turno activo.  
Id Glpi 11299.

### **DESARROLLO**

El problema radica en que solo se esta validando si el número de consignación ya existe en la tabla SIR\_OP\_DOCUMENTO\_PAGO y no se verifica si esta consignación esta relacionada con un turno anulado.

Para resolver este problema se creo un método que validara si el turno relacionado esta

anulado o no.

Nuevo Método:

```
gov.sir.hermiod.dao.impl.jdogenie.JDOPagosDAO:  
protected boolean isDocPagoInTurnoAnulado(String  
aIdDocumentoPago,PersistenceManager pm) throws DAOException
```

y se modifco el método:

```
gov.sir.hermiod.dao.impl.jdogenie.JDOPagosDAO:  
protected DocPagoConsignacionEnhanced getDocumentosPagoByConsignacion(  
DocPagoConsignacionEnhanced dp,  
PersistenceManager pm)  
throws DAOException  
  
rta = (DocPagoConsignacionEnhanced) iter.next();  
/**/  
 * @author: Fernando Padilla  
 * @change: Se valida si el documento de pago es utilizado en  
 * un turno anulado, esto se realiza por el  
 * Acta - Requerimiento No 072  
 */  
if(isDocPagoInTurnoAnulado(rta.getIdDocumentoPago(), pm)){  
    rta = null;  
}
```

## 077 Reporte 156 Anulados Del Día

### **REQUERIMIENTO**

Señores  
mesa de ayuda sir  
snr

El reporte de anulados de el día 03/08/2009 esta relacionando turnos de el año 2008 los cuales no fueron anulados en el día de ayer. por favor revisar que sucede con este reporte...agradezco su pronta colaboración. se anexo vía fax dicho reporte.

Se genera el reporte y se revisa, y efectivamente referencia turnos del 2008, que no fueron anulados el día de ayer.. 2008-300-1-1, 2008-300-1-2 y 2008-300-1-3.... 2008-300-6-1; 2008-300-6-2; 2008-300-6-3 Por favor verificar. Bucaramanga. Id Glpi 11186.

Informa Josefina Sánchez que al generar el reporte 156 anulados del día, con los siguientes parámetros: fecha de radicación 10/08/2009, No de Pagina: 1, circulo Registral: Neiva, aparecen turnos de años anteriores. Se anexa reporte para su verificación, gracias. Neiva. Id Glpi 11334.

### **DESARROLLO**

Se modifco el script del reporte 156 - DIARIO ANULADOS DEL DIA. Se adjunta reporte; verificar si el usuario lo necesita para enviarlo; y se les aclara que el cambio será reflejado en el próximo despliegue en producción. El archivo adjunto es Reporte\_156\_Neiva\_10\_08\_2009.pdf

```
*****  
/* Se agrega esta condición ya que la aplicación al momento de reasignar turnos a diferentes usuarios se están reasignando  
los turnos anulados y este historial esta siendo registrado en la tabla histórica del turno. Para esto se tuvo en cuenta  
el caso 11223 de GLPI. El reporte siempre tomara la primera fecha de anulación*/  
AND sir_op_turno_historia.TRHS_FECHA_REGISTRO IN  
(SELECT MIN(h.TRHS_FECHA_REGISTRO)  
FROM sir_op_turno_historia h
```

```

WHERE h.id_circulo = sir_op_turno_historia.id_circulo
AND h.id_proceso = sir_op_turno_historia.id_proceso
AND h.TRHS_ID_FASE ='ANULADO'
AND h.id_turno = sir_op_turno_historia.id_turno
)
/*****

```

## 079 Inconsistencias con el Reporte 25

### REQUERIMIENTO

Inconsistencias con el Reporte 25. José Luis García informa que el Reporte 25 del usuario Hugo.mantilla del día 05/08/09 muestra un turno que fue anulado 2009-300-1-144479 el día 15/07/2009, y en el día 15 no lo referencia. Por favor verificar los reportes. Pues el turno 2009-300-1-144479, tenía unas fases después de anulado, que fueron eliminadas por Base de datos para solucionar el inconveniente, pero de igual forma el reporte sigue mostrando el turno anulado el día 05/08/09 cuando lo anularon el 15/07/09. Bucaramanga. Id Glpi 11349.

### DESARROLLO

Se modifco el script de la sección de turnos anulados, se adjunta reporte, verificar si el usuario lo necesita y se le envía y se la aclara que este cambio será reflejado en el próximo despliegue a producción. El archivo adjunto es reporte-25-Bucaramanga-15-07-2009.pdf.

Se agrego al script la tabla SIR\_OP\_TURNO\_HISTORIA

```
-- Se agrega la tabla SIR_OP_TURNO-HISTORIA ya que anteriormente se tenían en cuenta la fecha_fin de la tabla SIR_OP_TURNO
sir_op_turno_historia h,
```

```
--
```

Se agregaron las siguientes condiciones:

```

/*****
WHERE sir_op_turno.id_circulo = h.id_circulo
AND sir_op_turno.anio = h.anio
AND sir_op_turno.id_proceso = h.id_proceso
AND sir_op_turno.id_turno = h.id_turno
AND h.trhs_id_fase = 'ANULADO'
--Se modifco la condición de la fecha ya que tiene que ir a la tabla SIR_OP_TURNO_HISTORIA
AND h.trhs_fecha_registro BETWEEN TO_DATE ('$P{P_PAGO_FECHA}', 'YYYY-MM-DD') AND TO_DATE
('$P{P_PAGO_FECHA}', 'YYYY-MM-DD')+ 1-- costo 64928
*****/

```

## 080 Avance de turnos para hacer definitivos cambios segregaciones (Informativo)

### REQUERIMIENTO

Avance de turnos para hacer definitivos cambios segregaciones. Cuando se hace una segregación en calificación (turno de registro) o en ejecutar corrección simple (turno de correcciones) y estos turnos son avanzados hasta firma de registro o revisar y aprobar, al tratar de avanzarlo de estas fases el sistema arroja error (imagen adjunta), como sucede con el caso 11354. Mesa de ayuda. Id Glpi 11425.

### DESARROLLO

Para este caso se ha tratado de reproducir en los ambientes de desarrollo y calidad pero hasta el momento no se ha presentado. Se cierra hasta que se reporte nuevamente en otra orip y así hacerle un seguimiento a los datos de producción y

<p>determinar si es un problema de datos o de aplicación.</p>	<b>081 Avance de turno calificación (Informativo)</b>	
	<p><b>REQUERIMIENTO</b></p> <p>Avance de turno calificación. Se está presentando el error: RuntimeException; CausedByException is: null cuando tratan de avanzar turnos de la fase de calificación, se reportaron los casos 11284, 11238, 11280, 11285, 11286, 11288, 11289 de la oficina de Neiva y el 11369 de la oficina de Popayán, con los turnos de Neiva los calificadores dieron clic varias veces y los turnos avanzaron sin inconveniente pero al tratar de hacer lo mismo con el caso de Popayán esto no funciona. Id Glpi 11431.</p>	
	<p><b>DESARROLLO</b></p> <p>Se reviso los casos puntuales referenciados en el caso GLPI-11431 ocurridas en las oficinas de Neiva y Popayán en el caso específico del turno 2009-120-6-7503 de la oficina de Popayán, se verifico que los usuarios de la oficina lograron realizar el proceso satisfactoriamente. Por lo que se optó por revisar todo el proceso de los turnos de registro desde la fase de CREACION hasta ENTREGA, y no se logró replicar en el ambiente de desarrollo y aun en el de producción el error reportado por los usuarios de la oficina.</p>	
	<p>Se toma la decisión de reportar el caso como resuelto con la salvedad de que si ocurre el inconveniente otra vez, por favor reportar el caso a quien lo revisó.</p>	
	<b>084 Error de impresión en relación de documento</b>	
	<p><b>REQUERIMIENTO</b></p> <p>Error de impresión en relación de documento. Nos informa Rodrigo Flores por que al imprimir la relación de documento esta saliendo con la ORIP de Bucaramanga y la relación, la fecha y la fase aparecen por la mitad de las letras. Rodrigo Flores nos comenta que la oficina necesita una solución puntual y definitiva. Manizales. Id Glpi 11415.</p>	
	<p><b>DESARROLLO</b></p> <p>Se modificó la configuración de los campos anteriormente estaban con tipo de posición Fijar Relativo al Tope y se cambió a Flotante; se adjunta reporte y se le indica al usuario que el cambio será reflejado en el próximo despliegue.</p>	
	<p>Se adjunto reporte: Reporte Rel 01 - 2009-100-3329- Manizales.pdf</p>	
	<b>085 Segregación Folios Calificación</b>	
	<p><b>REQUERIMIENTO</b></p> <p>Segregación Folios Calificación. Cuando se hace una segregación en calificación, en algunas ocasiones el sistema no está segregando los folios, realiza el proceso correctamente, al terminar el proceso el sistema indica que se ha realizado satisfactoriamente y en folio matriz se crean las anotaciones generadoras pero al revisar el turno en inscripción no muestra las matrículas segregadas. Se requiere revisar el modulo de segregación a nivel general en el sistema para corregir estos problemas presentados en el sistema ya que no es con todas las segregaciones si no en algunas ocasiones. Por favor escalar este caso a desarrollo y crear acta de requerimiento para este caso general.</p>	
	<p><b>DESARROLLO</b></p>	

Se revisa el caso de acuerdo a lo reportado por diferentes oficinas donde informaban que la segregación finalizaba correctamente pero no se generaban los folios segregados. El problema se presentaba debido al paginador que me permite cambiar de pagina para seleccionar las anotaciones a heredar ya que cuando se avanzaba de pagina en la aplicación solo obtenía el orden de las anotaciones ubicadas en la pagina actual, para las anotaciones seleccionadas en paginas diferentes a la actual insertaba el id donde correspondía el orden, por esta razón se incluían datos incorrectos en la tabla SIR.SIR\_OP\_WEB\_ANOTA\_HEREDA, al ejecutar el procedimiento almacenado SIR.SP\_SG se generaba una excepción que se manejaba con null, por esta razón no se podía determinar el error presentado.

Se decidió cargar en sesión todas las anotaciones para obtener el orden correcto y no perder información por el manejo que hace el paginador, cuando ya se finaliza la segregación se remueve de la sesión la información.

Clases modificadas:

- AWSegregacion.java

\*Ubicación: SIR-APP\web\WEB-INF\jspSeguro\registro

Código:

```
/**
 * @author: Guillermo Cabrera.
 * @change: Al direccion a la accion segregacion
 * se setea el valor a true, para identificar posteriormente
 * la accion a realizar.
 */
document.SEGREGACION.<%=WebKeys.CAMPO_ES_SEGREGACION%>.value =
"TRUE";
<!!--@author: Guillermo Cabrera
    @change: SE ADICIONA EL CAMPO ES SEGREGACION.
    MANTIS: 1726
-->
<td width="150"><input type="hidden"
name="<%=WebKeys.CAMPO_ES_SEGREGACION%">
id="<%=WebKeys.CAMPO_ES_SEGREGACION%" value="FALSE" /><a
href="javascript:esSegregacion();"></a></td>
```

- AWPaginadorAnotaciones.java

\*Ubicación : SIR-APP\src\gov\sir\core\web\acciones\comun

Código:

```
/**
 * @author: Guillermo Cabrera.
 * @change: si es segregación se sube a sesion las
anotaciones definitivas.
    * Se remueve de la sesión el campo para identificar si es
una segregación
    * ya que en este punto no se utiliza mas.
*/
if("TRUE".equals(request.getSession().getAttribute(WebKeys.CAMPO_ES_SEGREGACION))){
request.getSession().setAttribute(WebKeys.ANOTACIONES_A_HEREDAR,
respuesta.getAnotacionesDefinitivas());
}
request.getSession().removeAttribute(WebKeys.CAMPO_ES_SEGREGACION);
```

- WebKeys.java

ubicación: SIR-APP\src\gov\sir\core\web  
código:

```
/**
 * Constante que representa el nombre del campo del formulario para verificar si está en segregación.
*/
```

```

public static String CAMPO_ES_SEGREGACION = "CAMPO_ES_SEGREGACION";
/**
 * Constante con el nombre del atributo en sesion que contiene las anotaciones a heredar.
 */
public static String ANOTACIONES_A_HEREDAR = "ANOTACIONES_A_HEREDAR";

• ANOTACION_SEGREGACION.jsp
*ubicación: SIR-APP\src\gov\sir\core\web\acciones\registro
código:
/**/
* @author: Guillermo Cabrera.
    * @change: Se adiciona un campo para identificar si se va a
realizar una segregación
        * para cargar en sesión todas las anotaciones del folio
        * ya que anteriormente solo se cargaban las anotaciones del
paginador actual, quedando
            * incompletas las anotaciones a heredar.
/**/
request.getSession().setAttribute(WebKeys.CAMPO_ES_SEGREGACION,
request.getParameter(WebKeys.CAMPO_ES_SEGREGACION));
/**/
* @author: Guillermo Cabrera.
* @change: Se remueve de sesión las anotaciones
* que se cargaron inicialmente.
* MANTIS: 1726
/**/
request.getSession().removeAttribute(WebKeys.ANOTACIONES_A_HEREDAR);

```

Igualmente se Modificó el procedimiento almacenado SIR.SP\_SG, ya que la anotación generadora se crea con un id avanzado y le asigna el mismo orden a la anotación cuando en realidad el folio tiene menos anotaciones por lo tanto el orden debe ser diferente al id.

Ejemplo Folio 017-4402.

Se adjunta el sp modificado y se deja copia en el directorio  
C:\WORK\Documentos\DESARROLLO\REQUERIMIENTOS  
SNR\DOCUMENTOS ADJUNTOS\REQUERIMIENTO MANTIS 2071

### **086 Proceso Antiguo Sistema (Informativo)**

#### **REQUERIMIENTO**

Proceso Antiguo Sistema. Se requiere modificar el aplicativo SIR en el proceso de Antiguo sistema, toda vez que al ingresar anotaciones en la hoja de ruta en el campo valor (precio) del acto, cuando este tiene valores mínimos ejemplo un peso (\$ 1.00) el sistema permite ingresar este dato y es reflejado en la hoja de ruta, pero al pasar a definitivo el folio, este valor es remplazado por cero (\$ 0.), conllevando que se tenga que ingresar un turno de correcciones e ingresar nuevamente esta información; por lo tanto el aplicativo debe permitir incluir los valores que correspondan sin restricción y ser almacenados de manera adecuada, con el fin que al ser publicitados en el certificado y se refleje la información de manera adecuada. (caso turno 2009-200-1-66299 matricula 200-202087, tal como se refleja en la tabla de auditoría en el campo dtra\_valor\_id2.)

#### **DESARROLLO**

Se reviso el proceso al que se refiere el requerimiento y no se logro observar el error reportado por los usuario, para el caso puntual del turno 2009-200-1-66299 matricula 200-202087, tal como se refleja en la tabla de auditoría en el campo dtra\_valor\_id2 si se realizo una modificación del campo valor pero no se posee la suficiente

información para determinar donde se produce el error reportado o si esto obedeció a un error de usuario.

Se toma la decisión de reportar el caso como resuelto con la salvedad de que si ocurre el inconveniente otra vez, por favor reportar el caso a quien lo reviso.

### 088 Inconsistencia con el Reporte 93

#### **REQUERIMIENTO**

Inconsistencia con el Reporte 93. John nos informa indicando que el Reporte 93, al seleccionar la opción de todos, genera una información que es falsa, no muestra los turnos que tiene realmente el usuario y además contiene usuarios que no están activos en el SIR, como es el caso de Clemencia.estupinan. Se adjuntan los reportes generados desde la Mesa de Ayuda, al tomar la opción de todos, salen los datos inconsistentes, cruce de usuario con respecto a turnos, y cuando se genera tomando la opción de uno por uno, sale el reporte en blanco como en el caso del usuario Esperanza Ibarra. Popayán.

#### **DESARROLLO**

El reporte estaba trayendo información errada ya que al momento de mostrar el reporte tenia como encabezado el nombre del usuario eso quiere decir que el nombre del usuario se iba a iterar teniendo en cuenta el cambio de hoja mas no por turnos asignados para solucionar este inconveniente se creo un grupo en el reporte: "USUARIO" este grupo esta configurado por el campo USUARIO; entonces el reporte va a desplegar por usuario una hoja nueva desplegando sus turnos asignados, a demás de agrego una condición al script para que solo desplegara los usuario que se encuentran activos:

```
/* AHERREN 07/09/2009 */  
/* Se valida si el usuario se encuentra activo*/  
AND usuario.SRIO_ACTIVO = 1  
*****
```

Se adjunta el reporte de Popayán y se le informa al usuario que este cambio se vera reflejado en el próximo despliegue los archivos adjuntos son:

reporte-93- POPAYAN-JADHY-MERA.pdf  
reporte-93- POPAYAN-JUAN-ASTUDILLO.pdf  
reporte-93- POPAYAN-LUZ-SANDOVAL.pdf  
reporte-93- POPAYAN-MARIA-PERAFAN.pdf  
reporte-93- POPAYAN-MARTHA-ARDILA.pdf  
reporte-93- POPAYAN-TODOS.pdf

### 089 Certificados Asociados

#### **REQUERIMIENTO**

Habilitar turno. Cuando se habilita un turno de registro que tiene solicitud de certificado asociado, el aplicativo debe automáticamente habilitar el turno de certificado asociado,

Además el aplicativo debe listar el turno o turnos con la información de las respectivas columnas de cada uno de los reportes a los cuales corresponden al proceso del turno.

En el reporte 51debe reflejar la información correcta o coherente de cada uno de los

turnos ya sea anulados o habilitados, actualmente presenta error incrementando o disminuyendo el valor de los turnos mencionados.

### **DESARROLLO**

El Sistema permite que cuando los usuarios soliciten la Anulación de un Turno de Registro los turnos de Certificados Asociados sean anulados en el mismo proceso (Pantalla Administrativa -> Anular Turnos), pero en el caso de la solicitud para habilitar el turno de Registro los turnos de Certificados Asociados no eran habilitados (Pantalla Administrativa -> Habilitar Turnos).

Para realizar el Proceso completo de habilitar el turno de Registro junto a los Certificados Asociados se procedió a realizar la modificación de los siguientes métodos

- private EvnRespTrasladoTurno habilitarTurno(EvnTrasladoTurno evento) de la clase AnTrasladoTurno.java: SIR-APP\src\gov\sir\core\negocio\acciones\administracion encargada de manejar los eventos de tipo <code>EvnTrasladoTurno</code> destinados a manejar , el traslado de turnos entre usuarios, se agrega el siguiente código dentro del cuerpo del método para verificar que en caso de un turno de registro, se habilite los turnos de certificados asociados.

```
/*JAlcazar 11/09/2009 Caso MANTIS 0002358
 * Se agrega al proceso de habilitar turnos de Registro la posibilidad de habilitar
 * los turnos de certificados asociados si los hay, en el mismo instante de la solicitud
 * para el turno de registro
 //----- HABILITAR TURNOS HIJOS -----/
 */
if(turno.getSolicitud() instanceof SolicitudRegistro ){
    long idProceso = turno.getIdProceso();
    if(idProceso != Long.parseLong(CProceso.PROCESO_DEVOLUCIONES)){
        List solicitudesHijas = turno.getSolicitud().getSolicitudesHijas();
        Iterator ithijas = solicitudesHijas.iterator();
        while(ithijas.hasNext())
        {
            SolicitudAsociada solAsoc = (SolicitudAsociada) ithijas.next();
            Solicitud sol = solAsoc.getSolicitudHija();
            SolicitudPk sid = new SolicitudPk();
            sid.idSolicitud = sol.getIdSolicitud();
            Turno turnohijo = hermod.getTurnoBySolicitud(sid);
            if (CTurno.TURNO_ANULADO.equals(turnohijo.getAnulado()))
            {
                turnohijo = hermod.getTurnobyWF(turnohijo.getIdWorkflow());
                TurnoPk turnohijoId = new TurnoPk();
                turnohijoId.anio = turnohijo.getAnio();
                turnohijoId.idCirculo = turnohijo.getIdCirculo();
                turnohijoId.idProceso = turnohijo.getIdProceso();
                turnohijoId.idTurno = turnohijo.getIdTurno();
                turnohijo.setObservacionesAnulacion(evento.getObservacionesHabilitar());
                turnohijo.setAnulado(CTurno.TURNO_NO_ANULADO);
                turnohijo.setUsuarioAnulacion(evento.getUsuarioNegocio());
                turnohijo.setFechaFin(null);
            //----- HABILITAR TURNO -----/
                hermod.habilitarTurno(turnohijo);
            //----- FIN HABILITAR TURNO -----/
            }
        }
    }
}
```

```

//----- FIN ANULAR TURNOS HIJOS ---//  

}
}

```

- La segunda modificación realizada es realizada en el método protected void updatePagoPorHabilitarTurno(TurnoEnhanced tr, PersistenceManager pm) de la clase JDOTurnosDAO.java: SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie este método es invocado por el método hermod.habilitarTurno(turno) en la misma clase para validar que la consignación o el cheque asignada a un turno sea única y no puede estar registrada para otro turno, se modifco este proceso por que los turnos de registro si poseen certificados asociados comparten la misma consignación o el cheque y se debe evitar lanzar una excepción en este caso.

```

/*JAlcazar 11/09/2009 Caso MANTIS 0002358 se agrega en el if la siguiente validación:  

* tr.getSolicitud().getSolicitudesHijas()== null && tr.getSolicitud().getSolicitudesPadres() == null  

* debido a que los turnos de registro y sus hijos comparten el mismo recibo  

*/  

if(turnos!=null && !turnos.trim().equals("") && dp.getSaldoDocumento()<1000 &&  

dp.getSaldoDocumentoAnulacion()<1000 && tr.getSolicitud().getSolicitudesHijas()== null &&  

tr.getSolicitud().getSolicitudesPadres() == null )  

{  

throw new DAOException("Existen consignaciones y/o cheques que se usaron en la radicacion de  

nuevos turnos: " + turnos);  

}

```

## Análisis y Solución Reporte 51

Se realizan pruebas en la aplicación de calidad con:

El circulo: PRUEBAS

Usuario: USUARIO.PRUEBAS

Se radicaron 3 turnos con certificados asociados, cada uno de ellos se registró con tipos de pagos diferentes:

2009-999-6-314 – Asociado 2009-999-1-509 – Pago Efectivo 2009-

999-6-315 – Asociado 2009-999-1-510 – Pago Consignación 2009-

999-6-316 – Asociado 2009-999-1-511 – Pago Cheque

Se observo que para los pagos en efectivo y consignación en la sección Resumen General por conceptos se esta calculando mal, para corregir esto se hicieron las siguientes modificaciones:

Reporte 51		
1. Campo Efectivo - \$V{V_SUM_VALOR_EFECTIVOxCONCEPTO}		
Antes	\$F{ASOCIADOS_ANULADOS}!=null? (\$F{ASOCIADOS_ANULADOS}.subtract(\$F{EFECTIVO_APPLICADO})).doubleValue()<=(0.0)? new BigDecimal( (\$F{ASOCIADOS_ANULADOS}.subtract(\$F{EFECTIVO_APPLICADO})).doubleValue()*(- 1.0) ) :\$F{EFECTIVO_APPLICADO}!=null?\$F{EFECTIVO_APPLICADO}:new BigDecimal(0.0) :\$F{EFECTIVO_APPLICADO}!=null?\$F{EFECTIVO_APPLICADO}:new BigDecimal(0.0)	
Ahora	\$F{EFECTIVO_APPLICADO}!=null?\$F{EFECTIVO_APPLICADO}:new BigDecimal(0.0)	
2. Campo Consignación - \$V{v_consignacion_aplicado}		
Antes	\$F{ASOCIADOS_ANULADOS}!=null && \$V{v_saldo_asociados_anulados}.subtract(\$F{EFECTIVO_APPLICADO}).doubleValue()>=(0.0) ? \$F{ASOCIADOS_ANULADOS}.subtract( \$F{EFECTIVO_APPLICADO}.add(\$F{CONSIGNACION_APPLICADO})).doubleValue()<=(0.0) ? new BigDecimal(((\$F{ASOCIADOS_ANULADOS}.subtract( \$F{EFECTIVO_APPLICADO}.add(\$F{CONSIGNACION_APPLICADO}))).doubleValue() *(-1.0))	

	<pre>:new BigDecimal(0.0) :\$F{CONSIGNACION_APPLICADO}!=null?\$F{CONSIGNACION_APPLICADO}:new BigDecimal(0.0)</pre>
Ahora	<pre>\$F{CONSIGNACION_APPLICADO}!=null?\$F{CONSIGNACION_APPLICADO}:new BigDecimal(0.0)</pre>
<b>090 Digitación Masivo Direcciones (Informativo)</b>	
<b>REQUERIMIENTO</b>	
<p>Digitación Masiva: el aplicativo esta presentando en la fase de Digitación Masiva inconsistencias, cuando el usuario digitador masivo agrega una dirección, luego la elimina el sistema no la muestra en pantalla, luego se agrega una segunda dirección en esta fase, observando que el proceso es correcto , pero al enviarlo a la fase del calificador y revisarla se refleja la dirección eliminada y la dirección agregada lo cual no es correcto ya que en la fase de Digitación Masiva se hizo una eliminación de una dirección.</p>	
<p>El aplicativo debe permitir al digitador masivo editar o eliminar la información temporal agregada, siempre y cuando lo realice el mismo usuario que la ingreso.</p>	
<b>DESARROLLO</b>	
Se hicieron diferentes pruebas y no se pudo replicar el error.	
<p>Mesa de ayuda se comunicó con la oficina que presentó esta incidencia, pero no recibimos colaboración. Estuvimos esperando desde el Jueves 03 hasta hoy, Martes 08 de Septiembre.</p>	
<b>095 Proceso Correcciones - Visualizacion cambios</b>	
<b>REQUERIMIENTO</b>	
<p>En la fase de Revisar y Aprobar del proceso de Correcciones se deben observar todos los cambios realizados en el folio o folios cuando se da clic en el botón consultar. Se efectuó prueba en ambiente de producción con el turno de corrección 2009-999-3-90 y matricula 999-254, se ejecuto la corrección en la anotación numero 5 modificando el numero de anotación que cancela, en la anotación numero7 se modifco la especificación y código de naturaleza jurídica convirtiendo la anotación de estándar a canceladora, el turno se avanza a la fase de revisar y aprobar al consultar en el botón correspondiente cambios realizados el sistema muestra únicamente los de la anotación numero 7 lo cual es una inconsistencia. Al seleccionar el folio para consultar el sistema despliega toda la información, observándose los cambios temporales que se le ha realizado de las anotaciones 5 y 7, El turno se avanza hasta finalizarlo, luego se consulto el folio 999-254 y se corroboro los cambios realizados en las anotaciones 5 y 7.</p>	
<b>DESARROLLO</b>	
<p>El problema que tenía el proceso de correcciones radicaba en el objeto que se encargaba de guardar los cambios realizados entre la anotación definitiva y la temporal, el cual no era actualizado cada vez que se comparaban nuevas anotaciones. Esto se soluciono instanciando un nuevo objeto en la variable que guarda los cambios de las anotaciones en cada iteración del ciclo.</p>	
<p>Después de la Revisión de todo el proceso se realizaron unas mejoras a la forma como se determina los cambios que se hacen en las anotaciones al eliminar, agregar o actualizar los ciudadanos involucrados en las anotaciones.</p>	
<ul style="list-style-type: none"> <li>➤ Script Bases de Datos:</li> </ul>	
<p>NINGUNO</p>	

- Creación De Clases:  
NINGUNO
- Se modificaron los siguientes archivos:

**CLASE:**

**gov.sir.forseti.dao.impl.jdogenie.JDOGenieCiudadanoDAO.java**

**Método:**

```
public void
reemplazarCiudadanoPorTemporal(AnotacionCiudadanoEnhanced
anCiudAux, PersistenceManager pm)
```

**Código Línea 1394:**

```
if(ciudTMP!=null){
    ciudEnh = ciudTMP.getDefinitivo();
    anCiudAux.setCiudadano(ciudEnh);
}
Por:
if(ciudTMP!=null){
    ciudEnh = ciudTMP.getDefinitivo();
    anCiudAux.setCiudadano(ciudEnh);
/*JALCAZAR 20/10/2009 Caso Mantis 0002852
 * Marca como true los ciudadanos que fueron Actualizados */
    anCiudAux.setToUpdate(true);
}
```

**CLASE:**

**gov.sir.core.negocio.modelo.DeltaFolio.java**

**Método:**

***private void procesar()***

```
/*
 * Método que permite comparar cómo estaba un folio antes y después para poder
 * determinar las diferencias y así ver los cambios que se realizarón sobre el folio
 ****
 * Modificado por JALCAZAR 20/10/2009 caso Mantis 00028
 * Problemas para Consultar las Correciones realizadas en las anotaciones de los folios
 */
```

**Código Línea 191:**

```
Anotacion anotacionDelta = new Anotacion();
Por:
/*JALCAZAR 20/10/2009 caso Mantis 0002852 (Modificación N° 1)
 * El objeto que representa los cambios en cada Anotacion debe ser creado en el siguiente
 * ciclo que recorre las anotaciones temporales (ver Modificación N° 2).
 */
Anotacion anotacionDelta;
```

**Nuevo Código Línea 201:**

```
/*JALCAZAR 20/10/2009 caso Mantis 0002852 (Modificación N° 2)
 * (ver Modificación N° 1).
 */
anotacionDelta = new Anotacion();
```

**Nuevo Código Línea 272:**

```
/*JALCAZAR 20/10/2009 caso Mantis 0002852 (Modificación N° 3)
 * Se invoca el metodo anotacionesCuidadanosActualizadas(Anotacion anotacion) el cual muestra al
 usuario
 * si los datos de algun ciudadano de una anotacion dada fueron actualizados.
 */
anotacionesCuidadanosActualizadas(anotacionDefinitiva);
```

**Código Línea 296:**

```
else {
    this.diferencias.add(new Diferencias( id, "-", deltaCiudadanoTxt ) );
}
```

**Por:**

```
/*JALCAZAR 20/10/2009 caso Mantis 0002852 (Modificación N° 4)
 * Se invoca el metodo diferenciasAnotacionesCiudadanos(String IdAnotTemp, String deltaCiudTemp,
 Anotacion anotacion )
 * el cual verifica si el ciudadano fue agregado agregado a la anotacion en las Correciones.
```

```

        */
        else if (!diferenciasAnotacionesCiudadanos(anotCiud, anotacionDefinitiva )) {
            this.diferencias.add(new Diferencias( id, "-", deltaCiudadanoTxt ) );
        }
    }

```

### Nuevo Método:

#### **private boolean diferenciasAnotacionesCiudadanos(AnotacionCiudadano anotCiud, Anotacion anotacionDef)**

```

/**JALCAZAR 20/10/2009 caso Mantis 0002852 (Nuevo Metodo Nº 1)
 * Verifica si un ciudadano existe en la anotacion definitiva
 * @param anotCiud Ciudadano de la anotacion temporal
 * @param anotacionDef Anotacion Definitiva
 * @return
 * true: si el ciudadano existe.
 * false: si el ciudadano no existe.
 */
private boolean diferenciasAnotacionesCiudadanos(AnotacionCiudadano anotCiud, Anotacion anotacionDef)


```

```

List CiudaDefinitivos = anotacionDef.getAnotacionesCiudadanos();
Iterator itTemp = CiudaDefinitivos.iterator();
while (itTemp.hasNext()) {
    AnotacionCiudadano anotDef = (AnotacionCiudadano) itTemp.next();
    if(anotCiud.getIdCiudadano().equals(anotDef.getIdCiudadano())){
        return true;
    }
}
return false;
}

```

### Nuevo Método:

#### **private void anotacionesCuidadanosActualizadas(Anotacion anotacionDef)**

```

/**JALCAZAR 20/10/2009 caso Mantis 0002852 (Nuevo Metodo Nº 2)
 * Agrega a la lista de modificaciones de ciudadanos en la anotacion
 * aquellos usuarios que su informacion fue actualizada
 * @param anotacionDef Anotacion Definitiva
 */
private void anotacionesCuidadanosActualizadas(Anotacion anotacionDef){
    StringBuffer deltaCiudadanoTxt;
    String id;
    String local_AnotacionId = filterAnotacion_ExtractIdentifier(anotacionDef);
    List CiudaDefinitivos = anotacionDef.getAnotacionesCiudadanos();
    Iterator itTemp = CiudaDefinitivos.iterator();
    while (itTemp.hasNext()) {
        AnotacionCiudadano anotCiud = (AnotacionCiudadano) itTemp.next();
        id = FOLIO + " " + anotacionDef.getIdMatricula() + " - " + ( ANOTACION + " " +
        local_AnotacionId ) + " - " + CIUDADANO;
        deltaCiudadanoTxt = new StringBuffer();
        if (null != anotCiud.getIdCiudadano().getTipoDoc() )
            deltaCiudadanoTxt.append(anotCiud.getIdCiudadano().getTipoDoc());
        deltaCiudadanoTxt.append(" ");
        if(null != anotCiud.getIdCiudadano().getDocumento())
            deltaCiudadanoTxt.append(anotCiud.getIdCiudadano().getDocumento());
        deltaCiudadanoTxt.append(": ");
        if(null != anotCiud.getIdCiudadano().getNombreCompletoCiudadano())
            deltaCiudadanoTxt.append(anotCiud.getIdCiudadano().getNombreCompletoCiudadano());
        if(anotCiud.isToUpdate()){
            this.diferencias.add(new Diferencias( id, "MODIFICACION CIUDADANO", deltaCiudadanoTxt.toString() )
        }
    }
}

```

### 097 Radicación Nuevas Entradas

#### **REQUERIMIENTO**

Radicación registro, para la radicación de turnos de nuevas entradas se requiere que el aplicativo no permita al usuario liquidador registro eliminar actos ingresados en el turno anterior.

#### **DESARROLLO**

EL liquidador registro podía eliminar actos ingresados en el turno anterior, debido a que el helper(ActosHelper.java), que renderiza el formulario, tenia por defecto el atributo "mostrarEliminar" en true.

En la página /WEB-INF/jspSeguro/registro/REGISTRO\_LIQUIDAR.jsp se agregó la línea "actoHelper.setMostrarEliminar(false); para cambiar el valor del atributo "mostrarEliminar" a false, de la instancia de ActosHelper que se utiliza para renderizar el formulario en esta página.

## 100 Reporte 25 - Columna Proceso Mayor Valor

### **REQUERIMIENTO**

Se requiere que en el reporte 25, que en la columna proceso se concatene el proceso con los pagos de mayores valores Ejemplo: REG-MAV.

### **DESARROLLO**

#### **Análisis de Requerimiento:**

Se realizan pruebas en la aplicación de calidad con:

El circulo: PRUEBAS

Usuario: USUARIO.PRUEBAS

Se radico el turno de registro 2009-999-6-317

Radicación	Recibo	Vr. Aplicado			Vr.Docu.Pago	Diferencia Vr. Docu. Pago Vr. Aplicado	Vr. Liquidado	Hora Pago	Proc
		Efectivo	Consignación	Cheque					
2009-999-6-317	200000000000225	\$ 14,000	\$ 0	\$ 0	\$ 14,000	\$ 0	\$ 14,000	8:46:21 AM	REG
2009-999-6-317	200000000000226	\$ 250,000	\$ 0	\$ 0	\$ 250,000	\$ 0	\$ 250,000	9:38:10 AM	REG-MAV

Los cambios son los siguientes:

## Reporte 25 – Informe de recaudos del día por Usuario

### Campo Proceso

Antes	<pre> \$F{SOL_PADRE}!=null?(     (         (\$F{VR_APPLICADO_EFECTIVO}.equals(new java.math.BigDecimal("0")) )&amp;&amp;         (\$F{VR_APPLICADO_CONSIGNACION}.equals(new java.math.BigDecimal("0")) )&amp;&amp;         (\$F{VR_APPLICADO_CHEQUE}.equals(new java.math.BigDecimal("0")) ))     ?\$F{PROCESO_NOMBRE}+"-MSV"     :\$F{PROCESO_NOMBRE}+"-ASC"     :\$F{ID_PROCESO}.equals(new java.math.BigDecimal("6"))    ?("REG")     :(!\$F{CIRCULO_TURNO}.equals(\$P{P_CIRCULO_NOMBRE})     ?\$F{PROCESO_NOMBRE}+"-NAL"     :\$F{PROCESO_NOMBRE})     \$F{SOL_PADRE}!=null?(         (             (\$F{VR_APPLICADO_EFECTIVO}.equals(new java.math.BigDecimal("0")) )&amp;&amp;             (\$F{VR_APPLICADO_CONSIGNACION}.equals(new java.math.BigDecimal("0")) )&amp;&amp;             (\$F{VR_APPLICADO_CHEQUE}.equals(new java.math.BigDecimal("0")) ))         ?\$F{PROCESO_NOMBRE}+"-MSV"         :\$F{PROCESO_NOMBRE}+"-ASC"         :\$F{ID_PROCESO}.equals(new java.math.BigDecimal("6"))         ?\$F{PROCESO_NOMBRE}.equals("MAV")?("REG-MAV")?("REG")         :(!\$F{CIRCULO_TURNO}.equals(\$P{P_CIRCULO_NOMBRE})         ?\$F{PROCESO_NOMBRE}+"-NAL"         :\$F{PROCESO_NOMBRE})</pre>
Ahora	<pre> \$F{SOL_PADRE}!=null?(     (         (\$F{VR_APPLICADO_EFECTIVO}.equals(new java.math.BigDecimal("0")) )&amp;&amp;         (\$F{VR_APPLICADO_CONSIGNACION}.equals(new java.math.BigDecimal("0")) )&amp;&amp;         (\$F{VR_APPLICADO_CHEQUE}.equals(new java.math.BigDecimal("0")) ))     ?\$F{PROCESO_NOMBRE}+"-MSV"     :\$F{PROCESO_NOMBRE}+"-ASC"     :\$F{ID_PROCESO}.equals(new java.math.BigDecimal("6"))     ?\$F{PROCESO_NOMBRE}.equals("MAV")?("REG-MAV")?("REG")     :(!\$F{CIRCULO_TURNO}.equals(\$P{P_CIRCULO_NOMBRE})     ?\$F{PROCESO_NOMBRE}+"-NAL"     :\$F{PROCESO_NOMBRE})</pre>

## 101 Reporte 25 - Despliegue Turnos Anulados

### **REQUERIMIENTO**

En el reporte 25, se debe reflejar los turnos anulados y la fecha en que se anula el turno; es decir cuando se anula en fechas posteriores a la fecha de radicación del turno, debe listarse en el reporte del día de su radicación. Además debe reflejarse en el reporte 51, y todos los reportes del proceso al que corresponde el turno.

### **DESARROLLO**

Se realizan pruebas en la aplicación de calidad con:

El circulo: PRUEBAS

## Usuario: USUARIO.PRUEBAS

Se radicaron 3 turnos con certificados asociados, cada uno de ellos se registró con tipos de pagos diferentes; en la prueba estos turnos se anularon y se habilitaron:

2009-999-6-314 – Asociado 2009-999-1-509 – Pago Efectivo 2009-  
 999-6-315 – Asociado 2009-999-1-510 – Pago Consignación 2009-  
 999-6-316 – Asociado 2009-999-1-511 – Pago Cheque

En el análisis se detecto que el despliegue de los turnos anulados en el reporte 51 se realizaba teniendo en cuenta la fecha de radicación asumiendo esto no se realizo ningún cambio en este reporte. Se modiflico el script en el reporte 25 en el subReporte - "SUBREPORTE\_25\_TURNOSANULADOS":

```
*****  

/*16/09/2009 AHERRENO  

Se agrega subquery para consultar la fecha en que se realizo la anulación*/  

(SELECT MAX(H.TRHS_FECHA_REGISTRO)  

FROM sir_op_turno_historia H  

WHERE sir_op_turno.id_circulo = h.id_circulo  

AND sir_op_turno.anio = h.anio  

AND sir_op_turno.id_proceso = h.id_proceso  

AND sir_op_turno.id_turno = h.id_turno  

AND h.trhs_id_fase = 'ANULADO')fecha_anulacion  

*****  

*****  

/*16/09/2009 AHERRENO  

Se modiflico la condición de la fecha ya que tiene que ir a la  

tabla SIR_OP_PAGO - Fecha en que se registro el pago*/  

WHERE sir_op_pago.pago_fecha BETWEEN TO_DATE ($P{P_PAGO_FECHA}, 'YYYY-MM-DD') AND  

TO_DATE ($P{P_PAGO_FECHA}, 'YYYY-MM-DD')+ 1-- costo 64928  

*****
```

Se agrego la columna de Fecha de anulación:

Radicación	Recibo	Vr. Aplicado			Banco - No	Vr.Docu.Pago	Vr. Liquidado	Fecha Anulación	Hora Pago	Proc.						
		Efectivo	Consignación	Cheque												
2009-999-1-509	200000000000220	\$ 11,840	\$ 0	\$ 0		\$ 0	\$ 11,840	15/09/2009	09:59:15	CER-ASC						
2009-999-1-510	200000000000222	\$ 0	\$ 11,840	\$ 0	02-6353234 /	\$ 0	\$ 11,840	15/09/2009	13:28:55	CER-ASC						
2009-999-6-316	200000000000223	\$ 0	\$ 0	\$ 100,000	02-6353234 /	\$ 200,000	\$ 100,000	15/09/2009	14:52:25	REG						
2009-999-1-511	200000000000224	\$ 0	\$ 0	\$ 11,840	02-6353234 /	\$ 0	\$ 11,840	15/09/2009	14:52:25	CER-ASC						
Subtotales Turnos Anulados:		\$ 11,840	\$ 11,840	\$ 111,840		\$ 200,000	\$ 135,520									
<b>Certificados Otras ORIPS</b>																
<table border="1"> <thead> <tr> <th>Círculo</th> <th>Cantidad</th> <th>Valor</th> </tr> </thead> <tbody> <tr> <td>Total certificados entre ORIPS'S</td> <td></td> <td>\$ 0</td> </tr> </tbody> </table>											Círculo	Cantidad	Valor	Total certificados entre ORIPS'S		\$ 0
Círculo	Cantidad	Valor														
Total certificados entre ORIPS'S		\$ 0														

## 102 Reporte 51 - Devoluciones

### REQUERIMIENTO

Reporte 51, Resumen diario de Ingresos y Egresos.

En el resumen por usuarios el ítem del proceso de devoluciones de dinero debe reflejarse después del total del ítem de los demás procesos es decir que estos valores de las devoluciones no deben sumar ni restar de la sumatoria de los ingresos. En el Resumen general por Bancos el ítem del proceso de devoluciones de dinero debe reflejarse después del Total General en donde se indique el banco o bancos al que corresponde el cheque girado o en su defecto la Nota Debito cuando el pago se realice por este medio, estos valores solo se pueden sumar para totalizar lo referente a este proceso, pero no deben afectar para nada los ingresos de la OriP.

## DESARROLLO

Se realizan pruebas en la aplicación de calidad con:

El círculo: PRUEBAS

Usuario: USUARIO.PRUEBAS

Se radicaron 3 turnos de devolución:

2009-999-4-41

2009-999-4-44

2009-999-4-45

Como se efectuaron cambios en el reporte 51 también se realizaron para el reporte 25 para que al momento de evaluarlos no se desplieguen diferencias. Los cambios son los siguientes:

### Reporte 25 – Informe de recaudos del día por Usuario

#### Detalle de Turnos

Antes	Anteriormente los valores de las devoluciones se reflejaban negativas									
	Radicación	Recibo	Vr. Aplicado			Banco - No	Vr.Docu.Pago	Diferencia Vr. Docu. Pago Vr. Aplicado	Vr. Liquidado	Hora Pago
			Efectivo	Consignación	Cheque					
2009-999-6-314	2000000000000219	\$ 100,000	\$ 0	\$ 0			\$ 111,840	\$ 11,840	\$ 100,000	9:59:15 AM REG
2009-999-4-41	9000000000000001	\$ 0	\$ 0	\$ 1,400		07-123456 /	\$ 1,400	\$ 2,800	-\$ 14	11:30:51 AM DEV
2009-999-6-315	2000000000000221	\$ 0	\$ 100,000	\$ 0		02-53532234 /	\$ 200,000	\$ 100,000	\$ 100,000	1:28:54 PM REG
2009-999-4-44	9000000000000002	\$ 0	\$ 10,100	\$ 0		07-32654 /	-\$ 10,100	-\$ 20,200	-\$ 101	3:59:27 PM DEV
2009-999-4-45	9000000000000003	\$ 0	\$ 0	\$ 0		05-1 /	\$ 0	\$ 0	-\$ 808,800,000	5:13:15 PM DEV

#### Nueva Sección de Turnos de Devolución

Ahora	Se modifico la columna Cheque Los valores ya no se despliegan negativos Los únicos valores que se despliegan negativos son los de la liquidación ya que estos se almacenan así en la tabla SIR_OP_LIQUIDACION									
	<u>Turnos de Devolución</u>									
Radicación	Recibo	Vr. Aplicado			Banco - No	Vr.Docu.Pago	Vr. Liquidado	Hora Pago	Proc.	
		Efectivo	Consignación	Cheque/Nota Débito						
2009-999-4-41	9000000000000001	\$ 0	\$ 0	\$ 1,400	07-123456 /	\$ 1,400		-\$ 14	11:30:51	DEV
2009-999-4-44	9000000000000002	\$ 0	\$ 10,100	\$ 0	07-32654 /	\$ 10,100		-\$ 101	15:59:27	DEV
2009-999-4-45	9000000000000003	\$ 0	\$ 0	\$ 808,800,000	05-1 /	\$ 808,800,000		-\$ 808,800,000	17:13:15	DEV
Subtotales Turnos Devolución:		\$ 0	\$ 10,100	\$ 808,801,400			\$ 808,811,000		-\$ 808,800,115	
TOTALES										
Total Efectivo:				\$ 0						
Total Consignación:				\$ 10,100						
Total Cheque/Nota Débito:				\$ 909,901,400						
Total Oficina Vr:				\$ 808,811,500						

En la sección de turnos anulados se modifico el campo cheque

<u>Turnos Anulados</u>										
Radicación	Recibo	Vr. Aplicado			Banco - No	Vr.Docu.Pago	Vr. Liquidado	Fecha Anulación	Hora Pago	Proc.
		Efectivo	Consignación	Cheque/ Nota Débito						
2009-999-1-500	2000000000000220	\$ 11,840	\$ 0	\$ 0		\$ 0	\$ 11,840	15/09/2009	09:59:15	CERASC
Subtotales Turnos Anulados:		\$ 11,840	\$ 0	\$ 0		\$ 0	\$ 11,840			

Se modifico el script del reporte

reporte-25.jrxml

\*\*\*\*\*

/\*16/09/2009 - AHERRENO

Se agrega condición para que no despliegue turnos con el proceso 4 – Devoluciones; ya que se agrego una sección para los turnos de este proceso\*/

AND SUBSTR(sir\_op\_turno.trno\_id\_workflow, 10,1) != 4

	<pre>***** SUBREPORTE_25_BANCO.jrxml ***** /*16/09/2009 - AHERRENO Se agrega condición para que no tenga en cuenta valores de turnos del proceso 4 – Devoluciones; ya que se agrego una sección para los turnos de este proceso*/ AND p.id_solicitud IN (SELECT ot.id_solicitud FROM sir_op_turno ot WHERE SUBSTR(ot.trno_id_workflow, 10,1) != 4) *****</pre>																																																																																														
Antes	<p><b>Reporte 51 – Resumen Diario de Ingresos y Egresos</b></p> <table border="1"> <thead> <tr> <th colspan="12">RESUMEN GENERAL POR CONCEPTOS</th> </tr> <tr> <th>Concepto</th> <th>Cant. Turno</th> <th>Cuantia</th> <th>Derechos</th> <th>Impuesto + Sanción</th> <th>Liquidación</th> <th>Efectivo</th> <th>Consignación</th> <th>Cheque</th> <th>Total Aplicado</th> <th>Total Consignación + Cheque</th> <th>Diferencia</th> </tr> </thead> <tbody> <tr> <td>Certificados Anulados (3)</td> <td>3</td> <td>\$0</td> </tr> <tr> <td>Devoluciones Reg (2)</td> <td>2</td> <td>\$0</td> <td>\$0</td> <td>\$0</td> <td>\$0</td> <td>\$0</td> <td>\$0</td> <td>\$0</td> <td>-\$1,400</td> <td>-\$1,400</td> <td>-\$1,400</td> </tr> <tr> <td>Doc. Radicados (3)</td> <td>3</td> <td>\$62,300,000</td> <td>\$300,000</td> <td>\$0</td> <td>\$300,000</td> <td>\$100,000</td> <td>\$100,000</td> <td>\$200,000</td> <td>\$100,000</td> <td>\$200,000</td> <td>\$40,000</td> </tr> <tr> <td><b>Total USUARIO:PRUEBAS</b></td> <td></td> <td><b>\$ 62,300,000</b></td> <td><b>\$ 300,000</b></td> <td><b>\$ 0</b></td> <td><b>\$ 300,000</b></td> <td><b>\$ 100,000</b></td> <td><b>\$ 100,000</b></td> <td><b>\$ 200,000</b></td> <td><b>\$ 98,600</b></td> <td><b>\$ 198,600</b></td> <td><b>\$ 398,600</b></td> </tr> </tbody> </table> <p><b>CERTIFICADOS OTRAS ORIPS</b></p> <table border="1"> <thead> <tr> <th>Círculo</th> <th>Cantidad</th> <th>Valor</th> </tr> </thead> <tbody> <tr> <td>Total certificados entre ORIPs</td> <td></td> <td>\$ 0</td> </tr> </tbody> </table> <p><b>RESUMEN GENERAL POR BANCOS</b></p> <table border="1"> <thead> <tr> <th>Banco</th> <th>Vr Consignado</th> <th>Vr Cheque</th> <th>Vr Total</th> </tr> </thead> <tbody> <tr> <td>02 BANCO POPULAR</td> <td>\$ 200,000</td> <td>\$ 200,000</td> <td>\$ 400,000</td> </tr> <tr> <td>07 BANCOLOMBIA</td> <td>\$ 0</td> <td>\$ 1,400</td> <td>\$ 1,400</td> </tr> <tr> <td><b>Total General:</b></td> <td><b>\$ 200,000</b></td> <td><b>\$ 201,400</b></td> <td><b>\$ 401,400</b></td> </tr> </tbody> </table> <p>* (vr Consignación + vr Cheque) - (Vr Aplicado Consignación + Vr Aplicado Cheque)</p>	RESUMEN GENERAL POR CONCEPTOS												Concepto	Cant. Turno	Cuantia	Derechos	Impuesto + Sanción	Liquidación	Efectivo	Consignación	Cheque	Total Aplicado	Total Consignación + Cheque	Diferencia	Certificados Anulados (3)	3	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0	Devoluciones Reg (2)	2	\$0	\$0	\$0	\$0	\$0	\$0	\$0	-\$1,400	-\$1,400	-\$1,400	Doc. Radicados (3)	3	\$62,300,000	\$300,000	\$0	\$300,000	\$100,000	\$100,000	\$200,000	\$100,000	\$200,000	\$40,000	<b>Total USUARIO:PRUEBAS</b>		<b>\$ 62,300,000</b>	<b>\$ 300,000</b>	<b>\$ 0</b>	<b>\$ 300,000</b>	<b>\$ 100,000</b>	<b>\$ 100,000</b>	<b>\$ 200,000</b>	<b>\$ 98,600</b>	<b>\$ 198,600</b>	<b>\$ 398,600</b>	Círculo	Cantidad	Valor	Total certificados entre ORIPs		\$ 0	Banco	Vr Consignado	Vr Cheque	Vr Total	02 BANCO POPULAR	\$ 200,000	\$ 200,000	\$ 400,000	07 BANCOLOMBIA	\$ 0	\$ 1,400	\$ 1,400	<b>Total General:</b>	<b>\$ 200,000</b>	<b>\$ 201,400</b>	<b>\$ 401,400</b>
RESUMEN GENERAL POR CONCEPTOS																																																																																															
Concepto	Cant. Turno	Cuantia	Derechos	Impuesto + Sanción	Liquidación	Efectivo	Consignación	Cheque	Total Aplicado	Total Consignación + Cheque	Diferencia																																																																																				
Certificados Anulados (3)	3	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0																																																																																				
Devoluciones Reg (2)	2	\$0	\$0	\$0	\$0	\$0	\$0	\$0	-\$1,400	-\$1,400	-\$1,400																																																																																				
Doc. Radicados (3)	3	\$62,300,000	\$300,000	\$0	\$300,000	\$100,000	\$100,000	\$200,000	\$100,000	\$200,000	\$40,000																																																																																				
<b>Total USUARIO:PRUEBAS</b>		<b>\$ 62,300,000</b>	<b>\$ 300,000</b>	<b>\$ 0</b>	<b>\$ 300,000</b>	<b>\$ 100,000</b>	<b>\$ 100,000</b>	<b>\$ 200,000</b>	<b>\$ 98,600</b>	<b>\$ 198,600</b>	<b>\$ 398,600</b>																																																																																				
Círculo	Cantidad	Valor																																																																																													
Total certificados entre ORIPs		\$ 0																																																																																													
Banco	Vr Consignado	Vr Cheque	Vr Total																																																																																												
02 BANCO POPULAR	\$ 200,000	\$ 200,000	\$ 400,000																																																																																												
07 BANCOLOMBIA	\$ 0	\$ 1,400	\$ 1,400																																																																																												
<b>Total General:</b>	<b>\$ 200,000</b>	<b>\$ 201,400</b>	<b>\$ 401,400</b>																																																																																												
Ahora	<ul style="list-style-type: none"> <li>Se agrego sección Devoluciones para el detalle de cada usuario esta sección se identifica por una línea punteada.</li> <li>Se agrego sección Devoluciones para el Resumen General por conceptos esta sección se identifica por una línea punteada.</li> <li>Se modifco la columna cheque</li> <li>Se modifco la columna Consignación + Cheque</li> <li>Se agrego la Sección Resumen General por Bancos Devoluciones</li> <li>Los valores ya no se despliegan negativos</li> <li>Los únicos valores que se despliegan negativos son los de la liquidación ya que estos se almacenan así en la tabla SIR_OP_LIQUIDACION</li> </ul>																																																																																														

RESUMEN GENERAL POR CONCEPTOS												
Concepto	Cant. Turno	Cuantity	Derechos	Impuesto + Sanción	Liquidación	Efectivo	Consignación	Vr. Aplicado	Vr. Consignación	Total Aplicado	Consignación + Cheque + Nota Débito	Diferencia *
Certificados Anulados (3)	3	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0
Doc. Radiados (2)	2	\$42,000,000	\$200,000	\$0	\$200,000	\$100,000	\$100,000	\$200,000	\$0	\$0	\$200,000	\$100,000
Documentos Anulados (1)	1	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0	\$0
<b>Total Conceptos:</b>	<b>USUARIO PRUEBAS</b>	<b>\$200,000</b>	<b>\$0</b>	<b>\$200,000</b>	<b>\$100,000</b>	<b>\$0</b>	<b>\$100,000</b>	<b>\$200,000</b>	<b>\$0</b>	<b>\$200,000</b>	<b>\$100,000</b>	<b>\$0</b>
<b>Total Devoluciones:</b>	<b>USUARIO PRUEBAS</b>	<b>\$0</b>	<b>\$0</b>	<b>\$0</b>	<b>\$0</b>	<b>\$0</b>	<b>\$0</b>	<b>\$0</b>	<b>\$0</b>	<b>\$0</b>	<b>\$0</b>	<b>\$0</b>

\* (Vr. Consignación + vr. Cheque) - (Vr. Aplicado Consignación + Vr. Aplicado Cheque)

**CERTIFICADOS OTRAS ORIPS**

Círculo	Cantidad	Valor
Total certificados entre ORIPs		\$0

**RESUMEN GENERAL POR BANCOS**

Banco	Vr Consignado	Vr Cheque	Vr Total
02 BANCO POPULAR	\$200,000	\$0	\$200,000
<b>Total General:</b>	<b>\$200,000</b>	<b>\$0</b>	<b>\$200,000</b>

**RESUMEN GENERAL POR BANCOS DEVOLUCIONES**

Banco	Vr Consignado	Vr Cheque/Nota Débito	Vr Total
25 RANCAFF	\$0	\$308,800,000	\$308,800,000

- Se modifco el procedimiento del reporte, en cada INSERT a la tabla SIR\_TMP\_REPORTER\_51\_1 se agrego el tipo de pago 10 – Nota Debito

```

/*17-09-2009 AHERRENO
Se agrega el tipo de pago 10 - Nota Debito*/
INSERT
INTO SIR_TMP_REPORTER_51_1
(
    trno_id_workflow,
    doc_pago_id,
    id_banco,
    consignacion,
    cheque,
    vr_doc_pago,
    seq_proceso,
    id_usuario
)
VALUES
(
    V_ID_WORK_FLOW,
    ap(pp).id_documento_pago,
    doc(dd).id_banco,
    DECODE(doc(dd).id_tipo_doc_pago,2,doc(dd).dcpg_valor_documento, 0),
    DECODE(doc(dd).id_tipo_doc_pago,3,doc(dd).dcpg_valor_documento,
    4,doc(dd).dcpg_valor_documento,5,doc(dd).dcpg_valor_documento,10,doc(dd).dcpg_valor_documento,0),
    doc(dd).dcpg_valor_documento,
    sec_proceso,
    m(d).id_usuario
);
 Se modifco script subquery
SUBREPORTER_51_CONCEPTO.jrxml
*****
```

```

/*16/09/2009 - AHERRENO
Se agrega condición para que no despliegue turnos con el
proceso 4 - Devoluciones; ya que se agrego una sección
para los turnos de este proceso*/
AND concepto != 'Devoluciones Reg'
*****
```

- Se modifco el procedimiento del reporte, en cada INSERT a la tabla SIR\_TMP\_REPORTER\_51\_1 se agrego el tipo de pago 10 – Nota Debito

```

/*17-09-2009 AHERRENO
Se agrega el tipo de pago 10 - Nota Debito*/
INSERT
INTO SIR_TMP_Reporte_51_1
(
    trno_id_workflow,
    doc_pago_id,
    id_banco,
    consignacion,
    cheque,
    vr_doc_pago,
    seq_proceso,
    id_usuario
)
VALUES
(
    V_ID_WORK_FLOW,
    ap(pp).id_documento_pago,
    doc(dd).id_banco,
    DECODE(doc(dd).id_tipo_doc_pago,2,doc(dd).dcpg_valor_documento, 0),
    DECODE(doc(dd).id_tipo_doc_pago,3,doc(dd).dcpg_valor_documento,
    4,doc(dd).dcpg_valor_documento,5,doc(dd).dcpg_valor_documento,10,doc(dd).dcpg_valor_documento,0),
    doc(dd).dcpg_valor_documento,
    sec_proceso,
    m(d).id_usuario
);

```

- Se modifco script subquery  
**SUBREPORTE\_51\_CONCEPTO.jrxml**

```

/*****************/
/*16/09/2009 - AHERRENO
Se agrega condición para que no despliegue turnos con el
proceso 4 – Devoluciones; ya que se agrego una sección
para los turnos de este proceso*/
AND concepto != 'Devoluciones Reg'
/*****************/

```

## 104 Reporte folios masivos

### **REQUERIMIENTO**

Reporte folios masivos: la columna usuario se debe eliminar, las información se debe mostrar agrupada por usuarios, en la cual se debe mostrar un total de matriculas abiertas por usuario y al final un gran total de las matriculas creadas en el rango de fechas. El titulo de reporte se debe corregir debe ser: “RADICADOR MATRICULAS MASIVAS ANTIGUO SISTEMA”

### **DESARROLLO**

Se realizan pruebas en la aplicación de calidad con:

- El circulo: PRUEBAS
- Usuario: USUARIO.PRUEBAS y DESARROLLO.PRUEBAS
- Se radicaron los folios 999-6779, 999-6780, 999-681, 999-682 y 999-683

Los cambios son los siguientes:

<b>Reporte Radicador Matriculas Masivas Antiguo Sistema</b>	
Columna Usuario	
Eliminada	Grupo Usuario
Nuevo.	Variable \$V{TOTAL_MAT_USUARIO}
Totaliza los folios radicados por usuario	Variable \$V{TOTAL_MATRICULAS}

Nueva. Totaliza los folios radicados en el rango de fechas.	
<b>ADMINREPORTES_REPORT_FOLIOS_MAS_PARAMFORM.jsp</b>	
Antes	<td background="<%request.getContextPath()%>/jsp/images/tabla_central_bgn001.gif" class="titulotcentral">Reporte Folios Masivos</td>
Ahora	<td background="<%request.getContextPath()%>/jsp/images/tabla_central_bgn001.gif" class="titulotcentral">Radicador Matriculas Masivas Antiguo Sistema</td>
<b>Modificación Nombre Reporte</b>	
<pre>UPDATE SIR_OP_PANTALLA_ADMINISTRATIVA SET ptdm_nombre = 'Radicador Matriculas Masivas Antiguo Sistema' WHERE id_pantalla_administrativa = 1112 AND id_tipo_pantalla = 13 AND ptdm_vista = 'admin.reportes.report-FOLIOS-MAS.param-form.view'</pre>	

#### OFICINA DE REGISTRO DE INSTRUMENTOS PÚBLICOS CIRCULO\_PRUEBAS - 999

**RADICADOR MATRICULAS MASIVAS ANTIGUO SISTEMA**  
**FECHA CREACION DESDE 2009-10-26 HASTA 2009-10-27**

Impreso el 27 de Octubre de 2009 a las 08:25:55 AM

Página 1

##### USUARIO PRUEBAS

Matricula	Dirección	Municipio	Nombre Propietario	Fecha Creación
999-6779	AUTOPISTA N TRANSVERSAL N COMPLEMENTACION	VILLA DE LEYVA		26-10-2009 15:19:04
999-6780		MOTAVITA		26-10-2009 15:29:30
999-6781		MOTAVITA		26-10-2009 16:05:34
999-6782	AUTOPISTA NORTE CALLE 90 CASA	COMBITA		26-10-2009 17:10:54

**Total Matriculas Radicadas Por USUARIO PRUEBAS : 4**

##### DESARROLLO.PRUEBAS

Matricula	Dirección	Municipio	Nombre Propietario	Fecha Creación
999-6783	DIAGONAL 16 B # 106-65 CENTENARIO	BOYACA	PRUEBA PRUEBA PRUEBA	27-10-2009 08:22:32

**Total Matriculas Radicadas Por DESARROLLO.PRUEBAS : 1**

**Total Matriculas Radicadas En el Rango de Fecha: 5**

## 107 Reporte Genérico de Relaciones

### REQUERIMIENTO

Se requiere que todos los imprimibles generados al utilizar el genérico de relaciones tengan la misma configuración, del imprimible inicial; ya que al imprimir el genérico muestra una configuración diferente.

### DESARROLLO

Los archivos que se ejecutan para las relaciones son los siguientes:

- Reporte-38.jasper
- Reporte-97.jasper
- Reporte-41.jasper
- Reporte-35.jasper -> Archivo que tiene configuración
- Reporte-155.jasper

Las condiciones que se tuvieron en cuenta para la generación de estos reportes:

- No debe repetir el turno
- Ordenamiento por Fecha de Radicación y turno

Se modificaron los reportes quedando de la siguiente forma:

- Reporte-38.jasper
- o Para el campo "turno" se quito la configuración de Imprimir Valores Repetidos

o Se modifco el script

```
***** AHERRENO 08/09/2009 *****
/** Se modifco ordenamiento **/
ORDER BY turno.trno_fecha_inicio, turno.trno_id_workflow
*****
```

- Reporte-97.jasper

o Para el campo “turno” se quito la configuración de Imprimir Valores Repetidos  
o Se modifco el script

```
***** AHERRENO 08/09/2009 *****
/** Se modifco ordenamiento **/
ORDER BY 2, 1
*****
```

- Reporte-41.jasper

o Para el campo “turno” se quito la configuración de Imprimir Valores Repetidos  
o Se modifco el script

```
***** AHERRENO 08/09/2009 *****
/** Se modifco ordenamiento **/
order by calificador,fecha_hora_radicacion, turno
*****
```

- Reporte-155.jasper

o Para el campo “turno” se quito la configuración de Imprimir Valores Repetidos  
o Se modifco el script

```
***** AHERRENO 08/09/2009 *****
/** Se modifco ordenamiento **/
ORDER BY usuario.id_usuario||"||UPPER(usuario.srio_nombre_de_usuario), turno.trno_fecha_inicio, turno.trno_id_workflow
*****
```

Este cambio será reflejado en el próximo despliegue.

## 108 Rol Pago de Devoluciones - Nota Debito

### REQUERIMIENTO

Rol pago Devolución Nota Debito

En el ítem de forma de pago debe agregarse la opción “Nota Debito”, al seleccionar esta opción (Nota Debito) el aplicativo debe desplegar para ingresar los siguientes datos: Número, fecha, Código del Banco desplegando el listado de los Bancos y valor de la Nota Debito.

Nota: en el reporte 25 Informe Recaudos del día Tiracaja la información del ítem de devoluciones de dinero debe reflejarse en una sección aparte de la información de ingresos. En esta sección se debe reflejar todos los datos que se encuentran actualmente en la sección de ingresos adicionado en la forma de pago una columna para Nota Debito.

Adicionalmente se debe agregar un total de Nota Debito.

Radicación	Recibo	Vr. Aplicado			Banco - No	Vr.Docu.Pago	Diferencia	Vr. Docu. Pago	Vr. Liquidado	Hora Pago	Proc.
		Efectivo	Consignación	Cheque							
2009-3144-15	101	\$ 0	\$ 0	\$ 300.000	01-3990189 /	- \$ 300.000	- \$ 600.000	- \$ 300.000	11:36:26 AM	DEV	
<b>TOTALES</b>											
Total Efectivo:				\$ 0							
Total Consignación:				\$ 0							
Total Cheque:				\$ 300.000							
Total Oficina Vr:				\$ 300.000							
Total Otras ORIPS Vr:				\$ 0							
Vr. total aplicado:				\$ 300.000							

En el reporte numero 18, informe de devoluciones pagadas en cheque debe agregarse y nota debito].

Se debe adicionar en la columna forma de pago el pago con Nota Debito.

En el Reporte 21 informe de Devoluciones ingresadas del día xx al día yy, en la columna numero debe mostrar el titulo completo que es “Numero de Consignación / Cheque” al cual se le debe agregar Nota Debito.

En la columna de Solicitante se debe Justificar a la Izquierda.

Para el Reporte 79 Informe de Devoluciones pagadas de día xx al día yy, en la columna “Numero Consignación / Cheque” se debe agregar Nota Debito. En la Forma de pago se debe adicionar con Nota Debito.

Se solicita que la columna numero de resolución se concatene con la fecha de la Resolución para dar espacio a la Columna Nota Debito que se Adiciona.

### **DESARROLLO**

Se insertó un registro en la base de datos con el script:

```
INSERT INTO sir.sir_op_tipo_doc_pago(id_tipo_doc_pago, tpdp_nombre)
VALUES(10,'NOTA DEBITO');
```

Se crearon las clases:

gov.sir.core.negocio.modelo.DocPagoNotaDebito.java

gov.sir.core.negocio.modelo.jdogenie.DocPagoNotaDebitoEnhanced.java

gov.sir.core.negocio.modelo.transferutil.DocPagoNotaDebitoTransferCopier.java

Se modificaron los siguientes archivos:

#### **CLASE:**

**gov.sir.core.negocio.acciones.devolucion.ANDevolucion.java**

**Se hicieron las importaciones:**

```
import gov.sir.core.negocio.modelo.DocPagoNotaDebito;
import gov.sir.core.negocio.modelo.constantes.CAplicacionPago;
```

#### **Método:**

**private void efectuarPago(EvnDevolucion evento)**

**Código:**

```
if(tipoPago.equals("CHEQUE")) {
    valorPago = evento.getChequeValor();
} else if(tipoPago.equals("TRANSFERENCIA")) {
    valorPago = evento.getTransferenciaValor();
} else {
    valorPago = evento.getEfectivoValor();
}
```

**Por:**

```
if(tipoPago.equals(CAplicacionPago.CHEQUE)) {
    valorPago = evento.getChequeValor();
} else if(tipoPago.equals(CAplicacionPago.NOTA_DEBITO)) {
    valorPago = evento.getNotaDebitoValor();
} else if(tipoPago.equals(CAplicacionPago.TRANSFERENCIA)) {
    valorPago = evento.getTransferenciaValor();
} else {
    valorPago = evento.getEfectivoValor();
}
```

**Código:**

```
if(tipoPago.equals("CHEQUE")) {
    Banco banco = null;
    try{
        banco = hermod.getBancoByID(evento.getChequeBanco());
    }catch(HermodException h){
        throw new EventoException("No se Encontro el Banco");
    }catch(Throwable t){
        ExceptionPrinter printer = new ExceptionPrinter(t);
        Log.getInstance().error(ANDevolucion.class,"Ha ocurrido una excepcion
```

```

inesperada " + printer.toString());
}
throw new EventoException(t.getMessage(), t);
}

DocPagoCheque cheque = new DocPagoCheque(banco, "", "", evento.getChequeNumero(), evento.getChequeValor());

AplicacionPago ap = new AplicacionPago();
ap.setDocumentoPago(cheque);
ap.setValorAplicado(cheque.getValorDocumento());

pago.addAplicacionPago(ap);
}
//TIPO DE PAGO EN TRANSFERENCIA
if (tipoPago.equals("TRANSFERENCIA")) {
    Banco banco = null;
    try{
        banco = hermod.getBancoByID(evento.getTransferenciaBanco());
    }catch(HermodException h){
        throw new EventoException("No se Encontro el Banco");
    }catch(Throwable t){
        ExceptionPrinter printer = new ExceptionPrinter(t);
        Log.getInstance().error(ANDevolucion.class,"Ha ocurrido una excepcion
inesperada " + printer.toString());
        throw new EventoException(t.getMessage(), t);
    }
    DocPagoConsignacion cons = new DocPagoConsignacion(banco, "", "", evento.getTransferenciaNumero(), evento.getTransferenciaValor());
    AplicacionPago ap = new AplicacionPago();
    ap.setDocumentoPago(cons);
    ap.setValorAplicado(cons.getValorDocumento());
    pago.addAplicacionPago(ap);
}
//TIPO DE PAGO EN EFECTIVO:
if (tipoPago.equals("EFECTIVO")) {
    DocPagoEfectivo efectivo = new DocPagoEfectivo(evento.getEfectivoValor());
    AplicacionPago appEfectivo = new AplicacionPago();
    appEfectivo.setValorAplicado(evento.getEfectivoValor());
    appEfectivo.setDocumentoPago(efectivo);
    pago.addAplicacionPago(appEfectivo);
}

Por:
if (tipoPago.equals(CAplicacionPago.CHEQUE)) {
    Banco banco = null;
    try{
        banco = hermod.getBancoByID(evento.getChequeBanco());
    }catch(HermodException h){
        throw new EventoException("No se Encontro el Banco");
    }catch(Throwable t){
        ExceptionPrinter printer = new ExceptionPrinter(t);
        Log.getInstance().error(ANDevolucion.class,"Ha ocurrido una excepcion
inesperada " + printer.toString());
        throw new EventoException(t.getMessage(), t);
    }
    DocPagoCheque cheque = new DocPagoCheque(banco, "", "", evento.getChequeNumero(), evento.getChequeValor());
    AplicacionPago ap = new AplicacionPago();
    ap.setDocumentoPago(cheque);
    ap.setValorAplicado(cheque.getValorDocumento());
    pago.addAplicacionPago(ap);
}else if (tipoPago.equals(CAplicacionPago.NOTA_DEBITO)) {
    Banco banco = null;
    try{
        banco = hermod.getBancoByID(evento.getNotaDebitoBanco());
    }catch(HermodException h){
        throw new EventoException("No se Encontro el Banco");
    }catch(Throwable t){
        ExceptionPrinter printer = new ExceptionPrinter(t);
        Log.getInstance().error(ANDevolucion.class,"Ha ocurrido una excepcion
inesperada " + printer.toString());
        throw new EventoException(t.getMessage(), t);
    }
    DocPagoNotaDebito notaDebito = new DocPagoNotaDebito(banco, "", "", evento.getNotaDebitoNumero(), evento.getNotaDebitoValor());
    notaDebito.setTipoPago(tipoPago);
}

```

```

AplicacionPago ap = new AplicacionPago();
ap.setDocumentoPago(notaDebito);
ap.setValorAplicado(notaDebito.getValorDocumento());
pago.addAplicacionPago(ap);
}else if(tipoPago.equals("TRANSFERENCIA")) {//TIPO DE PAGO EN TRANSFERENCIA
Banco banco = null;
try{
    banco = hermod.getBancoByID(evento.getTransferenciaBanco());
}catch(HermodException h){
    throw new EventoException("No se Encontro el Banco");
}catch(Throwable t){
    ExceptionPrinter printer = new ExceptionPrinter(t);
    Log.getInstance().error(ANDevolucion.class,"Ha ocurrido una excepcion
inesperada " + printer.toString());
    throw new EventoException(t.getMessage(), t);
}
DocPagoConsignacion cons = new DocPagoConsignacion(banco, "", "", 
evento.getTransferenciaNumero(), evento.getTransferenciaValor());
AplicacionPago ap = new AplicacionPago();
ap.setDocumentoPago(cons);
ap.setValorAplicado(cons.getValorDocumento());
pago.addAplicacionPago(ap);
}else if(tipoPago.equals("EFECTIVO")) {//TIPO DE PAGO EN EFECTIVO:
DocPagoEfectivo efectivo = new DocPagoEfectivo(evento.getEfectivoValor());
AplicacionPago appEfectivo = new AplicacionPago();
appEfectivo.setValorAplicado(evento.getEfectivoValor());
appEfectivo.setDocumentoPago(efectivo);
pago.addAplicacionPago(appEfectivo); }
}

```

**CLASE:**

**gov.sir.core.negocio.modelo.transferutil. AplicacionPagoTransferCopier.java**

**Método:**

**public static gov.sir.core.negocio.modelo.jdogenie.Enhanced deepCopy(  
gov.sir.core.negocio.modelo.AplicacionPago transferObject, java.util.HashMap cache)**

**Código:**

```

if(documentoPago instanceof gov.sir.core.negocio.modelo.DocPagoCheque)
{
    gov.sir.core.negocio.modelo.jdogenie.DocPagoChequeEnhanced objEn =
(gov.sir.core.negocio.modelo.jdogenie.DocPagoChequeEnhanced)cache.get(documentoPago);
    if(objEn == null)
    {
        objEn =
(gov.sir.core.negocio.modelo.jdogenie.DocPagoChequeEnhanced)DocPagoChequeTransferCopier.deepCopy((gov.sir.c
ore.negocio.modelo.DocPagoCheque)documentoPago, cache);
    }
    enhancedObject.setDocumentoPago(objEn);
    assigned = true;
}

```

**Por:**

```

if(documentoPago instanceof gov.sir.core.negocio.modelo.DocPagoCheque)
{
    gov.sir.core.negocio.modelo.jdogenie.DocPagoChequeEnhanced objEn =
(gov.sir.core.negocio.modelo.jdogenie.DocPagoChequeEnhanced)cache.get(documentoPago);
    if(objEn == null)
    {
        objEn =
(gov.sir.core.negocio.modelo.jdogenie.DocPagoChequeEnhanced)DocPagoChequeTransferCopier.deepCopy((gov.sir.c
ore.negocio.modelo.DocPagoCheque)documentoPago, cache);
    }
    enhancedObject.setDocumentoPago(objEn);
    assigned = true;
    }else if(documentoPago instanceof gov.sir.core.negocio.modelo.DocPagoNotaDebito)
    {
        gov.sir.core.negocio.modelo.jdogenie.DocPagoNotaDebitoEnhanced objEn =
(gov.sir.core.negocio.modelo.jdogenie.DocPagoNotaDebitoEnhanced)cache.get(documentoPago);
        if(objEn == null)
        {
            objEn =
(gov.sir.core.negocio.modelo.jdogenie.DocPagoNotaDebitoEnhanced)DocPagoNotaDebitoTransferCopier.deepCopy((g
ov.sir.core.negocio.modelo.DocPagoNotaDebito)documentoPago, cache);
        }
        enhancedObject.setDocumentoPago(objEn);
        assigned = true;
    }
}

```

	<pre>         }  <b>CLASE:</b> <b>gov.sir.core.web.acciones.devolucion.AWDevolucion.java</b> </pre>	
	<p><b>Se crearon las constantes:</b></p> <pre> public static final String NOTA_DEBITO_NUMERO = "NOTA_DEBITO_NUMERO"; public static final String NOTA_DEBITO_VALOR = "NOTA_DEBITO_VALOR"; public static final String NOTA_DEBITO_BANCO = "NOTA_DEBITO_BANCO"; public static final String FECHA_NOTA_DEBITO = "FECHA_NOTA_DEBITO"; </pre>	
	<p><b>Método:</b></p> <pre> <b>public Evento perform(HttpServletRequest request)</b> </pre>	
	<p><b>Código:</b></p> <pre> else if (accion.equals(AWDevolucion.PAGO_CONFIRMAR)) {     PagoDevolucionException pde = new PagoDevolucionException();     EvnDevolucion evn = new EvnDevolucion(usuarioAuriga, EvnDevolucion.PAGO_CONFIRMAR, turno, fase, estacion,     EvnDevolucion.CONFIRMAR, usuario);     evn.setUID(session.getId());     evn.setRol(rol);     String forma_pago = "";     SolicitudDevolucion sol = (SolicitudDevolucion)turno.getSolicitud();     if(sol.isAprobada()){         forma_pago = request.getParameter(AWDevolucion.FORMA_PAGO);         String cheque_numero = "";         cheque_numero = request.getParameter(AWDevolucion.CHEQUE_NUMERO);         String cheque_banco = "";         cheque_banco = request.getParameter(AWDevolucion.CHEQUE_BANCO);         String cheque_valor = "";         cheque_valor = request.getParameter(AWDevolucion.CHEQUE_VALOR);         String cheque_fecha = "";         cheque_fecha = request.getParameter(AWDevolucion.FECHA_CHEQUE);         String trans_banco = "";         trans_banco = request.getParameter(AWDevolucion.TRANSFERENCIA_BANCO);         String trans_titular = "";         trans_titular = request.getParameter(AWDevolucion.TITULAR_TRANSFERENCIA);         String trans_numero = "";         trans_numero = request.getParameter(AWDevolucion.CUENTA_NUMERO);         String trans_valor = "";         trans_valor = request.getParameter(AWDevolucion.TRANSFERENCIA_VALOR);         String efectivo = "";         efectivo = request.getParameter(AWDevolucion.EFECTIVO_REGRESADO);         String orden_numero = "";         orden_numero = request.getParameter(AWDevolucion.NUMERO_ORDEN);         String orden_fecha = "";         orden_fecha = request.getParameter(AWDevolucion.FECHA_ORDEN);          request.getSession().setAttribute(AWDevolucion.FORMA_PAGO,forma_pago);         request.getSession().setAttribute(AWDevolucion.CHEQUE_NUMERO,cheque_numero);         request.getSession().setAttribute(AWDevolucion.CHEQUE_BANCO,cheque_banco);         request.getSession().setAttribute(AWDevolucion.CHEQUE_VALOR,cheque_valor);         request.getSession().setAttribute(AWDevolucion.FECHA_CHEQUE,cheque_fecha);         request.getSession().setAttribute(AWDevolucion.TRANSFERENCIA_BANCO,trans_banco);         request.getSession().setAttribute(AWDevolucion.TITULAR_TRANSFERENCIA,trans_titular);         request.getSession().setAttribute(AWDevolucion.CUENTA_NUMERO,trans_numero);         request.getSession().setAttribute(AWDevolucion.TRANSFERENCIA_VALOR,trans_valor);         request.getSession().setAttribute(AWDevolucion.EFECTIVO_REGRESADO,efectivo);         request.getSession().setAttribute(AWDevolucion.NUMERO_ORDEN,orden_numero);         request.getSession().setAttribute(AWDevolucion.FECHA_ORDEN,orden_fecha);         Date dateCheque = null;         //SI HAY DATOS EN UN CAMPO DE CHEQUE, GARANTIZAR QUE ESTEN         //LLENOS TODOS LOS DATOS DE CHEQUE.         if(forma_pago.equals(WebKeys.SIN_SELECCIONAR)){             pde.addError("Debe elegir una forma de pago válida");             throw pde;         }         if(forma_pago.equals(CAplicacionPago.CHEQUE)){             if(cheque_numero == null    cheque_numero.length() &lt;= 0){                 pde.addError("El número del cheque esta en blanco, por favor ingréselo.");             }             if(cheque_fecha == null    (cheque_fecha.equals(""))){                 pde.addError("Debe ingresar la fecha del del cheque");             } else {                 String sendI = null;                 try {                     dateCheque = DateFormatUtil.parse(cheque_fecha);                 }                 catch (Exception e){                     pde.addError("La fecha del cheque es inválida" + e.getMessage());                 }             }             if(cheque_banco == null    cheque_banco.length() &lt;= 0                cheque_banco.equals(WebKeys.SIN_SELECCIONAR)){                 pde.addError("Seleccione el banco.");             }         }     } } </pre>	

```

        }
        if(cheque_valor == null || cheque_valor.length() <= 0) {
            pde.addError("El valor del cheque esta en blanco, por favor ingréselo.");
        }else{
            try{
                //cheque_valor.replaceAll(".", "");
                cheque_valor=cheque_valor.replaceAll(".", "");
                int i = new Double(cheque_valor).intValue();
            }catch(Exception e){
                pde.addError("El valor del cheque debe ser númerico.");
            }
        }
    }
    if(forma_pago.equals(CAplicacionPago.TRANSFERENCIA)){
        if(trans_banco == null || trans_banco.length() <= 0 ||
trans_banco.equals(WebKeys.SIN_SELECCIONAR) ){
            pde.addError("Seleccione el banco.");
        }
        if(trans_titular == null || trans_titular.length() <= 0) {
            pde.addError("El titular de esta en blanco, por favor ingréselo.");
        }
        if(trans_numero == null || trans_numero.length() <= 0) {
            pde.addError("El número de la cuenta esta en blanco, por favor ingréselo.");
        }
        if(trans_valor == null || trans_valor.length() <= 0) {
            pde.addError("El valor de la transferencia esta en blanco, por favor ingréselo.");
        }else{
            try{
                //trans_valor.replaceAll(".", "");
                trans_valor=trans_valor.replaceAll(".", "");
                int i = new Double(trans_valor).intValue();
            }catch(Exception e){
                pde.addError("El valor de la transferencia debe ser númerico.");
            }
        }
    }
    if(forma_pago.equals(CAplicacionPago.EFECTIVO)){
        if(efectivo == null || efectivo.length() <= 0) {
            pde.addError("El valor en efectivo esta en blanco, por favor ingréselo.");
        }else{
            try{
                //efectivo.replaceAll(".", "");
                efectivo=efectivo.replaceAll(".", "");
                int i = new Double(efectivo).intValue();
            }catch(Exception e){
                pde.addError("El valor del efectivo debe ser numérico.");
            }
        }
    }
    if(orden_numero == null || orden_numero.length() <= 0) {
        pde.addError("El número de la orden de pago esta en blanco, por favor ingréselo.");
    }
    Date dateOrden = null;
    if( orden_fecha == null) || (orden_fecha.equals("")))) {
        pde.addError( "Debe ingresar la fecha de la orden de pago" );
    } else {
        try {
            dateOrden = DateFormatUtil.parse(orden_fecha);
        }
        catch (Exception e) {
            pde.addError("La fecha de orden de pago es inválida" + e.getMessage());
        }
    }
    evn.setNumeroOrden(orden_numero);
    evn.setFechaOrden(dateOrden);
    if(pde.getErrores().size()>0){
        throw pde;
    }
}
//NUMERO DEL CHEQUE.
if(forma_pago.equals(CAplicacionPago.CHEQUE) ){
    if(cheque_numero != null && cheque_numero.length() > 0) {
        evn.setChequeNumero(cheque_numero);
    }
    if(cheque_banco != null && cheque_banco.length() > 0) {
        evn.setChequeBanco(cheque_banco);
    }
    if(cheque_valor != null && cheque_valor.length() > 0) {
        evn.setChequeValor(new Double(cheque_valor).doubleValue());
    }
    if(cheque_fecha != null && cheque_fecha.length() > 0) {
        evn.setChequeFecha(dateCheque);
    }
}
//Transferencia.
if(forma_pago.equals(CAplicacionPago.TRANSFERENCIA) ){
    if(trans_numero != null && trans_numero.length() > 0) {

```

```

        evn.setTransferenciaNumero(trans_numero);
    }
    if(trans_titular != null && trans_titular.length() > 0) {
        evn.setTransferenciaTitular(trans_titular);
    }
    if(trans_valor != null && trans_valor.length() > 0) {
        evn.setTransferenciaValor(new Double(trans_valor).doubleValue());
    }
    if(trans_banco != null && trans_banco.length() > 0) {
        evn.setTransferenciaBanco(trans_banco);
    }
}
if(forma_pago.equals(CAplicacionPago.EFECTIVO )) {
    if(efectivo != null && efectivo.length() > 0) {
        evn.setEfectivoValor(new Double(efectivo).doubleValue());
    }
}
}
Por:
else if(accion.equals(AWDevolucion.PAGO_CONFIRMAR)) {
    PagoDevolucionException pde = new PagoDevolucionException();
    EvnDevolucion evn = new EvnDevolucion(usuarioAuriga, EvnDevolucion.PAGO_CONFIRMAR, turno, fase,
estacion, EvnDevolucion.CONFIRMAR, usuario);
    evn.setUID(session.getId());
    evn.setRol(rol);
    String forma_pago = "";
    SolicitudDevolucion sol = (SolicitudDevolucion)turno.getSolicitud();
    if(sol.isAprobada()){
        forma_pago = request.getParameter(AWDevolucion.FORMA_PAGO);
        String cheque_numero = "";
        cheque_numero = request.getParameter(AWDevolucion.CHEQUE_NUMERO);
        String cheque_banco = "";
        cheque_banco = request.getParameter(AWDevolucion.CHEQUE_BANCO);
        String cheque_valor = "";
        cheque_valor = request.getParameter(AWDevolucion.CHEQUE_VALOR);
        String cheque_fecha = "";
        cheque_fecha = request.getParameter(AWDevolucion.FECHA_CHEQUE);
        // Modificado por OSBERT LINERO - Iridium Telecomunicaciones e informática Ltda.
        // Cambio para agregar nota débito requerimiento 108 - Incidencia Mantis 02360.
        String notaDebitoNumero = "";
        notaDebitoNumero = request.getParameter(AWDevolucion.NOTA_DEBITO_NUMERO);
        String notaDebitoBanco = "";
        notaDebitoBanco = request.getParameter(AWDevolucion.NOTA_DEBITO_BANCO);
        String notaDebitoValor = "";
        notaDebitoValor = request.getParameter(AWDevolucion.NOTA_DEBITO_VALOR);
        String notaDebitoFecha = "";
        notaDebitoFecha = request.getParameter(AWDevolucion.FECHA_NOTA_DEBITO);
        String trans_banco = "";
        trans_banco = request.getParameter(AWDevolucion.TRANSFERENCIA_BANCO);
        String trans_titular = "";
        trans_titular = request.getParameter(AWDevolucion.TITULAR_TRANSFERENCIA);
        String trans_numero = "";
        trans_numero = request.getParameter(AWDevolucion.CUENTA_NUMERO);
        String trans_valor = "";
        trans_valor = request.getParameter(AWDevolucion.TRANSFERENCIA_VALOR);
        String efectivo = "";
        efectivo = request.getParameter(AWDevolucion.EFECTIVO_REGRESADO);
        String orden_numero = "";
        orden_numero = request.getParameter(AWDevolucion.NUMERO_ORDEN);
        String orden_fecha = "";
        orden_fecha = request.getParameter(AWDevolucion.FECHA_ORDEN);
        request.getSession().setAttribute(AWDevolucion.FORMA_PAGO,forma_pago);
        request.getSession().setAttribute(AWDevolucion.CHEQUE_NUMERO,cheque_numero);
        request.getSession().setAttribute(AWDevolucion.CHEQUE_BANCO,cheque_banco);
        request.getSession().setAttribute(AWDevolucion.CHEQUE_VALOR,cheque_valor);
        request.getSession().setAttribute(AWDevolucion.FECHA_CHEQUE,cheque_fecha);
        request.getSession().setAttribute(AWDevolucion.NOTA_DEBITO_NUMERO,notaDebitoNumero);
        request.getSession().setAttribute(AWDevolucion.NOTA_DEBITO_BANCO,notaDebitoBanco);
        request.getSession().setAttribute(AWDevolucion.NOTA_DEBITO_VALOR,notaDebitoValor);
        request.getSession().setAttribute(AWDevolucion.FECHA_NOTA_DEBITO,notaDebitoFecha);
        request.getSession().setAttribute(AWDevolucion.TRANSFERENCIA_BANCO,trans_banco);
        request.getSession().setAttribute(AWDevolucion.TITULAR_TRANSFERENCIA,trans_titular);
        request.getSession().setAttribute(AWDevolucion.CUENTA_NUMERO,trans_numero);
        request.getSession().setAttribute(AWDevolucion.TRANSFERENCIA_VALOR,trans_valor);
        request.getSession().setAttribute(AWDevolucion.EFECTIVO_REGRESADO,efectivo);
        request.getSession().setAttribute(AWDevolucion.NUMERO_ORDEN,orden_numero);
        request.getSession().setAttribute(AWDevolucion.FECHA_ORDEN,orden_fecha);
        Date dateCheque = null;
        Date dateNotaDebito = null;
        //SI HAY DATOS EN UN CAMPO DE CHEQUE, GARANTIZAR QUE ESTEN
        //LLENOS TODOS LOS DATOS DE CHEQUE.
        if(forma_pago.equals(WebKeys.SIN_SELECCIONAR )) {
            pde.addError("Debe elegir una forma de pago válida");
            throw pde;
        }else if(forma_pago.equals(CAplicacionPago.CHEQUE) ){
            if(cheque_numero == null || cheque_numero.length() <= 0) {

```

```

        pde.addError("El número del cheque está en blanco, por favor ingréselo.");
    }
    if( cheque_fecha == null || ("").equals(cheque_fecha)) {
        pde.addError( "Debe ingresar la fecha del cheque");
    } else {
        //String sendI = null;
        try {
            dateCheque = DateFormatUtil.parse(cheque_fecha);
        }
        catch (Exception e) {
            pde.addError("La fecha del cheque es inválida" + e.getMessage());
        }
    }

    if(cheque_banco == null || cheque_banco.length() <= 0 ||
cheque_banco.equals(WebKeys.SIN_SELECCIONAR) ) {
        pde.addError("Seleccione el banco.");
    }
    if(cheque_valor == null || cheque_valor.length() <= 0) {
        pde.addError("El valor del cheque está en blanco, por favor ingréselo.");
    }else{
        try{
            //cheque_valor.replaceAll(".", "");
            cheque_valor=cheque_valor.replaceAll(".", "");
            int i = new Double(cheque_valor).intValue();
        }catch(Exception e){
            pde.addError("El valor del cheque debe ser numérico.");
        }
    }

}else if(forma_pago.equals(CAplicacionPago.NOTA_DEBITO)){
    if( notaDebitoNumero == null || ("").equals(notaDebitoNumero) || notaDebitoNumero.length() <= 0) {
        if(notaDebitoNumero == null || ("").equals(notaDebitoNumero)){
            pde.addError("El número de la nota débito está en blanco, por favor ingréselo.");
        }else{
            pde.addError("El número de la nota débito no debe ser menor a 1, por favor ingréselo nuevamente.");
        }
    }
    if( notaDebitoFecha == null || ("").equals(notaDebitoFecha)) {
        pde.addError( "Debe ingresar la fecha de la nota débito");
    } else {
        //String sendI = null;
        try {
            dateNotaDebito = DateFormatUtil.parse(notaDebitoFecha);
        }
        catch (Exception e) {
            pde.addError("La fecha de la nota débito es inválida" + e.getMessage());
        }
    }
    if(notaDebitoBanco == null || notaDebitoBanco.length() <= 0 ||
notaDebitoBanco.equals(WebKeys.SIN_SELECCIONAR) ) {
        pde.addError("Seleccione el banco.");
    }
    if(notaDebitoValor == null || notaDebitoValor.length() <= 0) {
        pde.addError("El valor de la nota débito está en blanco, por favor ingréselo.");
    }else{
        try{
            //cheque_valor.replaceAll(".", "");
            notaDebitoValor = notaDebitoValor.replaceAll(".", "");
            int i = new Double(notaDebitoValor).intValue();
        }catch(Exception e){
            pde.addError("El valor de la nota débito debe ser numérico.");
        }
    }
}else if(forma_pago.equals(CAplicacionPago.TRANSFERENCIA)){
    if(trans_banco == null || trans_banco.length() <= 0 || trans_banco.equals(WebKeys.SIN_SELECCIONAR) ) {
        pde.addError("Seleccione el banco.");
    }
    if(trans_titular == null || trans_titular.length() <= 0) {
        pde.addError("El titular de está en blanco, por favor ingréselo.");
    }
    if(trans_numero == null || trans_numero.length() <= 0) {
        pde.addError("El número de la cuenta está en blanco, por favor ingréselo.");
    }
    if(trans_valor == null || trans_valor.length() <= 0) {
        pde.addError("El valor de la transferencia está en blanco, por favor ingréselo.");
    }else{
        try{
            //trans_valor.replaceAll(".", "");
            trans_valor=trans_valor.replaceAll(".", "");
            int i = new Double(trans_valor).intValue();
        }catch(Exception e){
            pde.addError("El valor de la transferencia debe ser numérico.");
        }
    }
}else if(forma_pago.equals(CAplicacionPago.EFECTIVO)){

```

```

        if (efectivo == null || efectivo.length() <= 0) {
            pde.addError("El valor en efectivo está en blanco, por favor ingréselo.");
        }else{
            try{
                //efectivo.replaceAll(",","");
                efectivo=efectivo.replaceAll(",","");
                int i = new Double(efectivo).intValue();
            }catch(Exception e){
                pde.addError("El valor del efectivo debe ser numérico.");
            }
        }
    }
    if(orden_numero == null || orden_numero.length() <= 0) {
        pde.addError("El número de la orden de pago está en blanco, por favor ingréselo.");
    }
    Date dateOrden = null;
    if( orden_fecha == null || ("").equals(orden_fecha)) {
        pde.addError( "Debe ingresar la fecha de la orden de pago" );
    } else {
        try {
            dateOrden = DateFormatUtil.parse(orden_fecha);
        } catch (Exception e) {
            pde.addError("La fecha de orden de pago es inválida" + e.getMessage());
        }
    }
    evn.setNumeroOrden(orden_numero);
    evn.setFechaOrden(dateOrden);
    if(pde.getErrores().size()>0){
        throw pde;
    }
//NUMERO DEL CHEQUE.
if(forma_pago.equals(CAplicacionPago.CHEQUE)) {
    if(cheque_numero != null && cheque_numero.length() > 0) {
        evn.setChequeNumero(cheque_numero);
    }
    if(cheque_banco != null && cheque_banco.length() > 0) {
        evn.setChequeBanco(cheque_banco);
    }
    if(cheque_valor != null && cheque_valor.length() > 0) {
        evn.setChequeValor(new Double(cheque_valor).doubleValue());
    }
    if(cheque_fecha != null && cheque_fecha.length() > 0) {
        evn.setChequeFecha(dateCheque);
    }
}else if(forma_pago.equals(CAplicacionPago.NOTA_DEBITO)){
    if(notaDebitoNumero != null && notaDebitoNumero.length() > 0) {
        evn.setNotaDebitoNumero(notaDebitoNumero);
    }
    if(notaDebitoBanco != null && notaDebitoBanco.length() > 0) {
        evn.setNotaDebitoBanco(notaDebitoBanco);
    }
    if(notaDebitoValor != null && notaDebitoValor.length() > 0) {
        evn.setNotaDebitoValor(new Double(notaDebitoValor).doubleValue());
    }
    if(notaDebitoFecha != null && notaDebitoFecha.length() > 0) {
        evn.setNotaDebitoFecha(dateNotaDebito);
    }
}else if(forma_pago.equals(CAplicacionPago.TRANSFERENCIA)){//Transferencia.
    if(trans_numero != null && trans_numero.length() > 0) {
        evn.setTransferenciaNumero(trans_numero);
    }
    if(trans_titular != null && trans_titular.length() > 0) {
        evn.setTransferenciaTitular(trans_titular);
    }
    if(trans_valor != null && trans_valor.length() > 0) {
        evn.setTransferenciaValor(new Double(trans_valor).doubleValue());
    }
    if(trans_banco != null && trans_banco.length() > 0) {
        evn.setTransferenciaBanco(trans_banco);
    }
}else if(forma_pago.equals(CAplicacionPago.EFECTIVO)){
    if(efectivo != null && efectivo.length() > 0) {
        evn.setEfectivoValor(new Double(efectivo).doubleValue());
    }
}
}

```

#### CLASE:

gov.sir.core.negocio.modelo.constantes.CAplicacionPago.java

#### Se creó la constante:

public static final String NOTA\_DEBITO = "NOTA DEBITO";

#### CLASE:

gov.sir.core.negocio.modelo.DocumentoPago.java

	<p><b>Se creó la constante:</b>  <code>public static final String PAGO_NOTA_DEBITO = "NOTA DEBITO";</code></p> <p style="text-align: center;"><b>CLASE:</b></p> <p style="text-align: center;"><b>gov.sir.core.negocio.modelo.jdogenie.DocumentoPagoEnhanced.java</b></p>	
	<p><b>Se creó la constante:</b>  <code>public static final String PAGO_NOTA_DEBITO = "NOTA DEBITO";</code></p> <p style="text-align: center;"><b>CLASE:</b></p> <p style="text-align: center;"><b>gov.sir.core.eventos.devolucion.EvnDevolucion.java</b></p>	
	<p><b>Se agregó el código:</b></p> <pre>private String notaDebitoBanco; private String notaDebitoNumero; private double notaDebitoValor; private Date notaDebitoFecha; /***  * Creado por OSBERT LINERO - Iridium Telecomunicaciones e informática Ltda.  * Cambio para agregar nota débito requerimiento 108 - Incidencia Mantis 02360.  * @return the notaDebitoBanco  */ public String getNotaDebitoBanco() {     return notaDebitoBanco; } /***  * Creado por OSBERT LINERO - Iridium Telecomunicaciones e informática Ltda.  * Cambio para agregar nota débito requerimiento 108 - Incidencia Mantis 02360.  * @param notaDebitoBanco the notaDebitoBanco to set  */ public void setNotaDebitoBanco(String notaDebitoBanco) {     this.notaDebitoBanco = notaDebitoBanco; } /***  * Creado por OSBERT LINERO - Iridium Telecomunicaciones e informática Ltda.  * Cambio para agregar nota débito requerimiento 108 - Incidencia Mantis 02360.  * @return the notaDebitoNumero  */ public String getNotaDebitoNumero() {     return notaDebitoNumero; } /***  * Creado por OSBERT LINERO - Iridium Telecomunicaciones e informática Ltda.  * Cambio para agregar nota débito requerimiento 108 - Incidencia Mantis 02360.  * @param notaDebitoNumero the notaDebitoNumero to set  */ public void setNotaDebitoNumero(String notaDebitoNumero) {     this.notaDebitoNumero = notaDebitoNumero; } /***  * Creado por OSBERT LINERO - Iridium Telecomunicaciones e informática Ltda.  * Cambio para agregar nota débito requerimiento 108 - Incidencia Mantis 02360.  * @return the notaDebitoValor  */ public double getNotaDebitoValor() {     return notaDebitoValor; } /***  * Creado por OSBERT LINERO - Iridium Telecomunicaciones e informática Ltda.  * Cambio para agregar nota débito requerimiento 108 - Incidencia Mantis 02360.  * @param notaDebitoValor the notaDebitoValor to set  */ public void setNotaDebitoValor(double notaDebitoValor) {     this.notaDebitoValor = notaDebitoValor; } /***  * Creado por OSBERT LINERO - Iridium Telecomunicaciones e informática Ltda.  * Cambio para agregar nota débito requerimiento 108 - Incidencia Mantis 02360.  * @return the notaDebitoFecha  */ public Date getNotaDebitoFecha() {     return notaDebitoFecha; } /***  * Creado por OSBERT LINERO - Iridium Telecomunicaciones e informática Ltda.  * Cambio para agregar nota débito requerimiento 108 - Incidencia Mantis 02360.  * @param notaDebitoFecha the notaDebitoFecha to set  */</pre>	

```

/*
public void setNotaDebitoFecha(Date notaDebitoFecha) {
    this.notaDebitoFecha = notaDebitoFecha;
}

```

### ARCHIVO:

gov.sir.impl.jdo.xml

#### Se agregó:

```

<class name="DocPagoNotaDebitoEnhanced" persistence-capable-superclass="DocumentoPagoEnhanced">
    <field name="codSucursalBanco">
        <extension vendor-name="versant" key="jdbc-column">
            <extension vendor-name="versant" key="jdbc-column-name" value="DCPG_COD_SUCURSAL_BANCO" />
        </extension>
    </field>
    <field name="noNotaDebito">
        <extension vendor-name="versant" key="jdbc-column">
            <extension vendor-name="versant" key="jdbc-column-name" value="DCPG_NO_NOTA_DEBITO" />
            <extension vendor-name="versant" key="jdbc-length" value="100" />
        </extension>
    </field>
    <field name="noCuenta">
        <extension vendor-name="versant" key="jdbc-column">
            <extension vendor-name="versant" key="jdbc-column-name" value="DCPG_NO CUENTA" />
        </extension>
    </field>
    <extension vendor-name="versant" key="jdbc-class-id" value="10" />
    <field name="fecha">
        <extension vendor-name="versant" key="jdbc-column">
            <extension vendor-name="versant" key="jdbc-column-name" value="DCPG_FECHA_DOCUMENTO" />
        </extension>
    </field>
    <field name="banco">
        <extension vendor-name="versant" key="jdbc-ref" value="idBanco">
            <extension vendor-name="versant" key="jdbc-column">
                <extension vendor-name="versant" key="jdbc-column-name" value="ID_BANCO" />
            </extension>
        </extension>
    </field>
</class>

```

### CLASE:

gov.sir.hermmod.dao.impl.jdogenie.JDOPagosDAO.java

#### Se hicieron las importaciones:

```

import gov.sir.core.negocio.modelo.DocPagoNotaDebito;
import gov.sir.core.negocio.modelo.jdogenie.DocPagoNotaDebitoEnhanced;

```

#### Y se comentaron las importaciones:

```

import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;

```

### Método:

**protected PagoEnhancedPk crearPago(PagoEnhanced p, Usuario user, PersistenceManager pm)**

#### Código:

```

if(dpr instanceof DocPagoConsignacionEnhanced) {
    Date fechaDate = dpr.getFechaCreacion();
    Calendar calendarDocPago = new GregorianCalendar();
    calendarDocPago.setTime(fechaDate);
    Calendar calendarHoy = new GregorianCalendar();
    calendarHoy.setTime(new Date());
    /*if(calendarDocPago.get(Calendar.YEAR) < calendarHoy.get(Calendar.YEAR)) ||
(calendarDocPago.get(Calendar.DAY_OF_YEAR) < calendarHoy.get(Calendar.DAY_OF_YEAR)) {
        throw new DAOException("El saldo del documento solo puede utilizarse el primer día que
ingresa al sistema");
    }*/
} else if(dpr instanceof DocPagoChequeEnhanced) {
    Date fechaDate = dpr.getFechaCreacion();
    Calendar calendarDocPago = new GregorianCalendar();
    calendarDocPago.setTime(fechaDate);
    Calendar calendarHoy = new GregorianCalendar();
    calendarHoy.setTime(new Date());
    /*if(calendarDocPago.get(Calendar.YEAR) < calendarHoy.get(Calendar.YEAR)) ||

```

	<pre>(calendarDocPago.get(Calendar.DAY_OF_YEAR) &lt; calendarHoy.get(Calendar.DAY_OF_YEAR)) {     throw new DAOException("El saldo del documento solo puede utilizarse el primer día que ingresa al sistema"); } }  <b>Por:</b> if(dpr instanceof DocPagoConsignacionEnhanced    dpr instanceof DocPagoChequeEnhanced    dpr instanceof DocPagoNotaDebitoEnhanced) {     Date fechaDate = dpr.getFechaCreacion();     Calendar calendarDocPago = new GregorianCalendar();     calendarDocPago.setTime(fechaDate);     Calendar calendarHoy = new GregorianCalendar();     calendarHoy.setTime(new Date());     /*if ((calendarDocPago.get(Calendar.YEAR) &lt; calendarHoy.get(Calendar.YEAR))    (calendarDocPago.get(Calendar.DAY_OF_YEAR) &lt; calendarHoy.get(Calendar.DAY_OF_YEAR)) {          throw new DAOException("El saldo del documento solo puede utilizarse el primer día que ingresa al sistema");     } }</pre>	
	<b>Método:</b>	
	<p><b>protected TurnoEnhancedPk procesarPago(PagoEnhanced p, String estacion, String impresora, Usuario user, PersistenceManager pm)</b></p>	
	<p><b>Código:</b></p> <pre>if(dpr instanceof DocPagoConsignacionEnhanced) {     Date fechaDate = dpr.getFechaCreacion();     Calendar calendarDocPago = new GregorianCalendar();     calendarDocPago.setTime(fechaDate);     Calendar calendarHoy = new GregorianCalendar();     calendarHoy.setTime(new Date());     /*if ((calendarDocPago.get(Calendar.YEAR) &lt; calendarHoy.get(Calendar.YEAR))    (calendarDocPago.get(Calendar.DAY_OF_YEAR) &lt; calendarHoy.get(Calendar.DAY_OF_YEAR)) {          throw new DAOException("El saldo del documento solo puede utilizarse el primer día que ingresa al sistema");     } } //          } catch (java.text.ParseException pe) { //              Log.getInstance().error(JDOPagosDAO.class,"Error de formato en fecha: " + fechaString + ":" + pe.getMessage(), pe); // } else if(dpr instanceof DocPagoChequeEnhanced) {     Date fechaDate = dpr.getFechaCreacion();     Calendar calendarDocPago = new GregorianCalendar();     calendarDocPago.setTime(fechaDate);     Calendar calendarHoy = new GregorianCalendar();     calendarHoy.setTime(new Date());     /*if ((calendarDocPago.get(Calendar.YEAR) &lt; calendarHoy.get(Calendar.YEAR))    (calendarDocPago.get(Calendar.DAY_OF_YEAR) &lt; calendarHoy.get(Calendar.DAY_OF_YEAR)) {          throw new DAOException("El saldo del documento solo puede utilizarse el primer día que ingresa al sistema");     } } //          } catch (java.text.ParseException pe) { //              Log.getInstance().error(JDOPagosDAO.class,"Error de formato en fecha: " + fechaString + ": " + pe.getMessage(), pe); // }</pre>	
	<p><b>Por:</b></p> <pre>if(dpr instanceof DocPagoConsignacionEnhanced    dpr instanceof DocPagoChequeEnhanced    dpr instanceof DocPagoNotaDebitoEnhanced) {     Date fechaDate = dpr.getFechaCreacion();     Calendar calendarDocPago = new GregorianCalendar();     calendarDocPago.setTime(fechaDate);     Calendar calendarHoy = new GregorianCalendar();     calendarHoy.setTime(new Date());     /*if ((calendarDocPago.get(Calendar.YEAR) &lt; calendarHoy.get(Calendar.YEAR))    (calendarDocPago.get(Calendar.DAY_OF_YEAR) &lt; calendarHoy.get(Calendar.DAY_OF_YEAR)) {          throw new DAOException("El saldo del documento solo puede utilizarse el primer día que ingresa al sistema");     } } //          } catch (java.text.ParseException pe) { //              Log.getInstance().error(JDOPagosDAO.class,"Error de formato en fecha: " + fechaString + ":" + pe.getMessage(), pe); // }</pre>	
	<b>Método:</b>	

	<pre>public Pago validarPago(Pago pago)</pre>	
	<p><b>Código:</b></p> <pre>if(dp instanceof DocPagoChequeEnhanced) {     DocPagoChequeEnhanced dpc = (DocPagoChequeEnhanced) dp;     dpc = this.getDocumentosPagoByCheque(dpc, pm);     dpr = dpc; } if(dp instanceof DocPagoConsignacionEnhanced) {     DocPagoConsignacionEnhanced dpo = (DocPagoConsignacionEnhanced) dp;     dpo = this.getDocumentosPagoByConsignacion(dpo, pm);     dpr = dpo; }</pre>	
	<p><b>Por:</b></p> <pre>if(dp instanceof DocPagoChequeEnhanced) {     DocPagoChequeEnhanced dpc = (DocPagoChequeEnhanced) dp;     dpc = this.getDocumentosPagoByCheque(dpc, pm);     dpr = dpc; }else if(dp instanceof DocPagoNotaDebitoEnhanced) {     DocPagoNotaDebitoEnhanced dpc = (DocPagoNotaDebitoEnhanced) dp;     dpc = this.getDocumentosPagoByNotaDebito(dpc, pm);     dpr = dpc; }else if(dp instanceof DocPagoConsignacionEnhanced) {     DocPagoConsignacionEnhanced dpo = (DocPagoConsignacionEnhanced) dp;     dpo = this.getDocumentosPagoByConsignacion(dpo, pm);     dpr = dpo; }</pre>	
	<b>Método:</b>	
	<pre>protected DocumentoPagoEnhanced crearDocumentoPago(DocumentoPagoEnhanced dp, PersistenceManager pm)</pre>	
	<p><b>Código:</b></p> <pre>if(dp instanceof DocPagoChequeEnhanced) {     DocPagoChequeEnhanced dpc = (DocPagoChequeEnhanced) dp;     bId.idBanco = dpc.getBanco().getIdBanco();     b = BancosDAO.getBancoByID(bId, pm);     dpc.setBanco(b); } //Documento de Pago Consignación if(dp instanceof DocPagoConsignacionEnhanced) {     DocPagoConsignacionEnhanced dpo = (DocPagoConsignacionEnhanced) dp;     bId.idBanco = dpo.getBanco().getIdBanco();     b = BancosDAO.getBancoByID(bId, pm);     dpo.setBanco(b); }</pre>	
	<p><b>Por:</b></p> <pre>if(dp instanceof DocPagoChequeEnhanced) {     DocPagoChequeEnhanced dpc = (DocPagoChequeEnhanced) dp;     bId.idBanco = dpc.getBanco().getIdBanco();     b = BancosDAO.getBancoByID(bId, pm);     dpc.setBanco(b); }else if(dp instanceof DocPagoNotaDebitoEnhanced) { //Documento de Pago Nota Débito     DocPagoNotaDebitoEnhanced dpc = (DocPagoNotaDebitoEnhanced) dp;     bId.idBanco = dpc.getBanco().getIdBanco();     b = BancosDAO.getBancoByID(bId, pm);     dpc.setBanco(b); }else if(dp instanceof DocPagoConsignacionEnhanced) //Documento de Pago Consignación     DocPagoConsignacionEnhanced dpo = (DocPagoConsignacionEnhanced) dp;     bId.idBanco = dpo.getBanco().getIdBanco();     b = BancosDAO.getBancoByID(bId, pm);     dpo.setBanco(b); }</pre>	
	<b>Nuevo Método:</b>	
	<pre>protected DocPagoNotaDebitoEnhanced getDocumentosPagoByNotaDebito(DocPagoNotaDebitoEnhanced dp, PersistenceManager pm) throws DAOException {     DocPagoNotaDebitoEnhanced rta = null;     if(dp.getNoNotaDebito() != null &amp;&amp; dp.getNoCuenta() != null) {         try {             Query query = pm.newQuery(DocPagoNotaDebitoEnhanced.class);             //String cuenta = dp.getNoCuenta();             String notaDebito = dp.getNoNotaDebito();             BancoEnhanced banco = dp.getBanco();             query.declareParameters("String notaDebito, String bancoId");             //query.declareVariables("BancoEnhanced banco");             query.setFilter("noNotaDebito == notaDebito &amp;&amp; banco.idBanco == bancoId");         }     } }</pre>	

```

Collection col = (Collection) query.execute(notaDebito, banco.getIdBanco());
if(col.size() == 0) {
    rta = null;
} else {
    for (Iterator iter = col.iterator(); iter.hasNext(); ) {
        rta = (DocPagoNotaDebitoEnhanced) iter.next();
    }
    query.closeAll();
}
} catch (JDOObjectNotFoundException e) {
    Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
    rta = null;
} catch (JDOException e) {
    Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
    throw new DAOException(e.getMessage(), e);
}
}
return rta;
}

```

#### Nuevo Método:

```

public DocPagoNotaDebito getDocPagoNotaDebito(DocumentoPago docPago) throws DAOException {
    DocumentoPagoEnhanced dpr = null;
    PersistenceManager pm = AdministradorPM.getPM();
    try {
        dpr = this.getDocPagoNotaDebito(docPago, pm);
        this.makeTransientDocumentoPago(dpr, pm);
    } catch (DAOException e) {
        if(pm.currentTransaction().isActive()) {
            pm.currentTransaction().rollback();
        }
        Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
        throw new DAOException(e.getMessage(), e);
    } catch (Throwable e) {
        if(pm.currentTransaction().isActive()) {
            pm.currentTransaction().rollback();
        }
        Log.getInstance().error(JDOPagosDAO.class,e.getMessage());
        throw new DAOException(e.getMessage(), e);
    } finally {
        pm.close();
    }
    if(dpr!=null){
        return (DocPagoNotaDebito) dpr.toTransferObject();
    }else{
        return null;
    }
}

```

#### Nuevo Método:

```

public DocPagoNotaDebito getDocPagoNotaDebito(String noNotaDebito) throws DAOException {
    DocumentoPagoEnhanced dpr = null;
    PersistenceManager pm = AdministradorPM.getPM();
    try {
        dpr = this.getDocPagoNotaDebito(noNotaDebito, pm);
        this.makeTransientDocumentoPago(dpr, pm);
    } catch (DAOException e) {
        if(pm.currentTransaction().isActive()) {
            pm.currentTransaction().rollback();
        }
        Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
        throw new DAOException(e.getMessage(), e);
    } catch (Throwable e) {
        if(pm.currentTransaction().isActive()) {
            pm.currentTransaction().rollback();
        }
        Log.getInstance().error(JDOPagosDAO.class,e.getMessage());
        throw new DAOException(e.getMessage(), e);
    } finally {
        pm.close();
    }
    return (DocPagoNotaDebito) dpr.toTransferObject();
}

```

#### Método:

```

public DocumentoPago getDocumentosPagoExistente(DocumentoPago doc)

```

**Código:**

```

if(doc instanceof DocPagoCheque){
    dpr = this.getDocumentosPagoByCheque((DocPagoChequeEnhanced)
DocPagoChequeEnhanced.enhance(doc), pm);
}
else if(doc instanceof DocPagoConsignacion){
    dpr = this.getDocumentosPagoByConsignacion((DocPagoConsignacionEnhanced)
DocPagoConsignacionEnhanced.enhance(doc), pm);
}

```

**Por:**

```

if(doc instanceof DocPagoCheque){
    dpr = this.getDocumentosPagoByCheque((DocPagoChequeEnhanced)
DocPagoChequeEnhanced.enhance(doc), pm);
}else if(doc instanceof DocPagoNotaDebito){
    dpr = this.getDocumentosPagoByNotaDebito((DocPagoNotaDebitoEnhanced)
DocPagoNotaDebitoEnhanced.enhance(doc), pm);
}
else if(doc instanceof DocPagoConsignacion){
    dpr = this.getDocumentosPagoByConsignacion((DocPagoConsignacionEnhanced)
DocPagoConsignacionEnhanced.enhance(doc), pm);
}

```

**Nuevo Método:**

```

protected DocPagoNotaDebitoEnhanced getDocPagoNotaDebito( DocumentoPago dp, PersistenceManager pm)
throws DAOException {
    DocPagoNotaDebitoEnhanced rta = null;
    if (dp.getIdDocumentoPago() != null) {
        try {
            Query query = pm.newQuery(DocPagoNotaDebitoEnhanced.class);
            String idDocPago = dp.getIdDocumentoPago();
            query.declareParameters("String idDocPago");
            query.setFilter("idDocumentoPago == idDocPago");
            Collection col = (Collection) query.execute(idDocPago);
            if (col.size() == 0) {
                rta = null;
            } else {
                for (Iterator iter = col.iterator(); iter.hasNext();) {
                    rta = (DocPagoNotaDebitoEnhanced) iter.next();
                }
                query.closeAll();
            }
        } catch (JDOObjectNotFoundException e) {
            Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
            rta = null;
        } catch (JDOException e) {
            Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
            throw new DAOException(e.getMessage(), e);
        }
    }
    return rta;
}

```

**Nuevo Método:**

```

protected DocPagoNotaDebitoEnhanced getDocPagoNotaDebito( String noNotaDebito, PersistenceManager pm)
throws DAOException {
    DocPagoNotaDebitoEnhanced rta = null;
    if (noNotaDebito != null) {
        try {
            Query query = pm.newQuery(DocPagoNotaDebitoEnhanced.class);
            query.declareParameters("String noNota");
            query.setFilter("noNotaDebito == noNota");
            Collection col = (Collection) query.execute(noNotaDebito);
            if (col.size() == 0) {
                rta = null;
            } else {
                for (Iterator iter = col.iterator(); iter.hasNext();) {
                    rta = (DocPagoNotaDebitoEnhanced) iter.next();
                }
                query.closeAll();
            }
        } catch (JDOObjectNotFoundException e) {
            Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
            rta = null;
        } catch (JDOException e) {
            Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
        }
    }
}

```

	<pre>         throw new DAOException(e.getMessage(), e);     } } return rta; } </pre>	
<b>Método:</b>		
<b>protected Vector ordenarDocumentos(List documentos)</b>		
<b>Se agregó La declaración:</b> <pre>Vector pagoNotaDebito = new Vector();</pre>		
<b>Código:</b> <pre> for(Iterator iter = documentos.iterator() ; iter.hasNext();){     DocumentoPagoEnhanced dp = (DocumentoPagoEnhanced)iter.next();     if(dp.getTipoPago().equals(DocumentoPagoEnhanced.PAGO_EFECTIVO)){         pagoEfectivo.add(dp);     }     if(dp.getTipoPago().equals(DocumentoPagoEnhanced.PAGO_CONSIGNACION)){         pagoConsignacion.add(dp);     }     if(dp.getTipoPago().equals(DocumentoPagoEnhanced.PAGO_CHEQUE)){         pagoCheque.add(dp);     }     if(dp.getTipoPago().equals(DocumentoPagoEnhanced.PAGO_CONVENIO)){         pagoConvenio.add(dp);     }     if(dp.getTipoPago().equals(DocumentoPagoEnhanced.TIMBRE_CONSTANCIA_LIQUIDACION)){         timbre.add(dp);     } } datosOrdenados.addAll(pagoEfectivo); datosOrdenados.addAll(pagoConsignacion); datosOrdenados.addAll(pagoCheque); datosOrdenados.addAll(pagoConvenio); datosOrdenados.addAll(timbre); return datosOrdenados; </pre>		
<b>Por:</b> <pre> for(Iterator iter = documentos.iterator() ; iter.hasNext();){     DocumentoPagoEnhanced dp = (DocumentoPagoEnhanced)iter.next();     if(dp.getTipoPago().equals(DocumentoPagoEnhanced.PAGO_EFECTIVO)){         pagoEfectivo.add(dp);     }     if(dp.getTipoPago().equals(DocumentoPagoEnhanced.PAGO_CONSIGNACION)){         pagoConsignacion.add(dp);     }     if(dp.getTipoPago().equals(DocumentoPagoEnhanced.PAGO_CHEQUE)){         pagoCheque.add(dp);     }     }else if(dp.getTipoPago().equals(DocumentoPagoEnhanced.PAGO_NOTA_DEBITO)){         pagoNotaDebito.add(dp);     }     if(dp.getTipoPago().equals(DocumentoPagoEnhanced.PAGO_CONVENIO)){         pagoConvenio.add(dp);     }     if(dp.getTipoPago().equals(DocumentoPagoEnhanced.TIMBRE_CONSTANCIA_LIQUIDACION)){         timbre.add(dp);     } } datosOrdenados.addAll(pagoEfectivo); datosOrdenados.addAll(pagoConsignacion); datosOrdenados.addAll(pagoCheque); datosOrdenados.addAll(pagoNotaDebito); datosOrdenados.addAll(pagoConvenio); datosOrdenados.addAll(timbre); return datosOrdenados; </pre>		
<b>Método:</b>		
<b>public boolean getDocsPagosDevolucion(DocumentoPago dp)</b>		
<b>Código:</b> <pre> else if(dp instanceof DocPagoCheque){     String cheque = ((DocPagoCheque)dp).getNoCheque();     if(((DocPagoCheque)dp).getBanco() != null) {         idBanco=((DocPagoCheque)dp).getBanco().getIdBanco();         try {             Query query = pm.newQuery(DocPagoChequeEnhanced.class);             query.declareParameters("String cheque, String bancoId");             query.setFilter("noCheque == cheque &amp;&amp; banco.idBanco == bancoId");         }     } } </pre>		

```

        query.setOrdering("idDocumentoPago ascending");
        Collection col = (Collection) query.execute(cheque,idBanco);
        if(col.size() == 0) {
            rta = null;
        } else {
            for (Iterator iter = col.iterator(); iter.hasNext();) {
                rta.add( ((DocPagoChequeEnhanced) iter.next()).getIdDocumentoPago() );
            }
            query.closeAll();
        }
    } catch (JDOObjectNotFoundException e) {
        Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
        rta = null;
    } catch (JDOException e) {
        Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
        throw new DAOException(e.getMessage(), e);
    }
}
}

Por:
else if(dp instanceof DocPagoCheque){
    String cheque = ((DocPagoCheque)dp).getNoCheque();
    if(((DocPagoCheque)dp).getBanco() != null) {
        idBanco=((DocPagoCheque)dp).getBanco().getIdBanco();
        try {
            Query query = pm.newQuery(DocPagoChequeEnhanced.class);
            query.declareParameters("String cheque, String bancoId");
            query.setFilter("noCheque == cheque && banco.idBanco == bancoId");
            query.setOrdering("idDocumentoPago ascending");
            Collection col = (Collection) query.execute(cheque,idBanco);
            if(col.size() == 0) {
                rta = null;
            } else {
                for (Iterator iter = col.iterator(); iter.hasNext();) {
                    rta.add( ((DocPagoChequeEnhanced) iter.next()).getIdDocumentoPago() );
                }
                query.closeAll();
            }
        } catch (JDOObjectNotFoundException e) {
            Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
            rta = null;
        } catch (JDOException e) {
            Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
            throw new DAOException(e.getMessage(), e);
        }
    }
}
}

else if(dp instanceof DocPagoNotaDebito){
    String notaDebito = ((DocPagoNotaDebito)dp).getNoNotaDebito();
    if(((DocPagoNotaDebito)dp).getBanco() != null) {
        idBanco=((DocPagoNotaDebito)dp).getBanco().getIdBanco();
        try {
            Query query = pm.newQuery(DocPagoNotaDebitoEnhanced.class);
            query.declareParameters("String notaDebito, String bancoid");
            query.setFilter("noNotaDebito == notaDebito && banco.idBanco == bancoid");
            query.setOrdering("idDocumentoPago ascending");
            Collection col = (Collection) query.execute(notaDebito,idBanco);
            if(col.size() == 0) {
                rta = null;
            } else {
                for (Iterator iter = col.iterator(); iter.hasNext();) {
                    rta.add( ((DocPagoNotaDebitoEnhanced) iter.next()).getIdDocumentoPago() );
                }
                query.closeAll();
            }
        } catch (JDOObjectNotFoundException e) {
            Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
            rta = null;
        } catch (JDOException e) {
            Log.getInstance().error(JDOPagosDAO.class,e.getMessage(), e);
            throw new DAOException(e.getMessage(), e);
        }
    }
}
}
}

```

PÁGINA JSP:

Se le dió formato a la página y se corrigieron las etiquetas que estaban mal cerradas.

**Código:**

```

function mostrarFormaPago(valor){
    if(valor=='<%=CAplicacionPago.CHEQUE%>'){
        document.all.cheque1.style.display = 'block'
        document.all.cheque2.style.display = 'block'
        document.all.transferencia1.style.display = 'none'
        document.all.transferencia2.style.display = 'none'
        document.all.efectivo1.style.display = 'none'
        document.all.efectivo2.style.display = 'none'
    }
    if(valor=='<%=CAplicacionPago.TRANSFERENCIA%>'){
        document.all.cheque1.style.display = 'none'
        document.all.cheque2.style.display = 'none'
        document.all.transferencia1.style.display = 'block'
        document.all.transferencia2.style.display = 'block'
        document.all.efectivo1.style.display = 'none'
        document.all.efectivo2.style.display = 'none'
    }
    if(valor=='<%=CAplicacionPago.EFECTIVO%>'){
        document.all.cheque1.style.display = 'none'
        document.all.cheque2.style.display = 'none'
        document.all.transferencia1.style.display = 'none'
        document.all.transferencia2.style.display = 'none'
        document.all.efectivo1.style.display = 'block'
        document.all.efectivo2.style.display = 'block'
    }
    if(valor=='<%=WebKeys.SIN_SELECCIONAR%>'){
        document.all.cheque1.style.display = 'none'
        document.all.cheque2.style.display = 'none'
        document.all.transferencia1.style.display = 'none'
        document.all.transferencia2.style.display = 'none'
        document.all.efectivo1.style.display = 'none'
        document.all.efectivo2.style.display = 'none'
    }
}

```

**Por:**

```

function mostrarFormaPago(valor){
    if(valor=='<%=CAplicacionPago.CHEQUE%>'){
        document.all.cheque1.style.display = 'block'
        document.all.cheque2.style.display = 'block'
        document.all.notaDebito1.style.display = 'none'
        document.all.notaDebito2.style.display = 'none'
        document.all.transferencia1.style.display = 'none'
        document.all.transferencia2.style.display = 'none'
        document.all.efectivo1.style.display = 'none'
        document.all.efectivo2.style.display = 'none'
    }else if(valor=='<%=CAplicacionPago.NOTA_DEBITO%>'){
        document.all.cheque1.style.display = 'none'
        document.all.cheque2.style.display = 'none'
        document.all.notaDebito1.style.display = 'block'
        document.all.notaDebito2.style.display = 'block'
        document.all.transferencia1.style.display = 'none'
        document.all.transferencia2.style.display = 'none'
        document.all.efectivo1.style.display = 'none'
        document.all.efectivo2.style.display = 'none'
    }else if(valor=='<%=CAplicacionPago.TRANSFERENCIA%>'){
        document.all.cheque1.style.display = 'none'
        document.all.cheque2.style.display = 'none'
        document.all.notaDebito1.style.display = 'none'
        document.all.notaDebito2.style.display = 'none'
        document.all.transferencia1.style.display = 'block'
        document.all.transferencia2.style.display = 'block'
        document.all.efectivo1.style.display = 'none'
        document.all.efectivo2.style.display = 'none'
    }else if(valor=='<%=CAplicacionPago.EFECTIVO%>'){
        document.all.cheque1.style.display = 'none'
        document.all.cheque2.style.display = 'none'
        document.all.notaDebito1.style.display = 'none'
        document.all.notaDebito2.style.display = 'none'
        document.all.transferencia1.style.display = 'none'
        document.all.transferencia2.style.display = 'none'
        document.all.efectivo1.style.display = 'block'
    }
}

```

```

        document.all.efectivo2.style.display = 'block'
    }else if(valor=='%WebKeys.SIN_SELECCIONAR%'){
        document.all.cheque1.style.display = 'none'
        document.all.cheque2.style.display = 'none'
        document.all.notaDebito1.style.display = 'none'
        document.all.notaDebito2.style.display = 'none'
        document.all.transferencia1.style.display = 'none'
        document.all.transferencia2.style.display = 'none'
        document.all.efectivo1.style.display = 'none'
        document.all.efectivo2.style.display = 'none'
    }
}

Se agrego um item AL select de tipo de pago:
elementosLista.add(new ElementoLista( CAplicacionPago.NOTA_DEBITO, CAplicacionPago.NOTA_DEBITO ) );
Se agregaron las tablas:
<table width="100%" border="0" cellpadding="0" cellspacing="0" id="notaDebito1" style='display:none'>
    <tr>
        <td width="12"></td>
        <td background="<%request.getContextPath()%>/jsp/images/sub_bgn001.gif" class="bgnsub"> Nota débito </td>
        <td width="16" class="bgnsub"></td>
        <td width="16" class="bgnsub"></td>
        <td width="15"></td>
    </tr>
</table>
<table width="100%" class="camposform" id="notaDebito2" style='display:none'>
    <tr>
        <td width="20"></td>
        <td>Introducir datos nota débito</td>
        <td>&ampnbsp</td>
    </tr>
    <tr>
        <td>&ampnbsp</td>
        <td>N&acute;mero:</td>
        <td>
            <% try {
                TextHelper textHelper = new TextHelper();
                textHelper.setNombre(AWDevolucion.NOTA_DEBITO_NUMERO);
                textHelper.setCssClase("camposformtext");
                textHelper.setSize("25");
                textHelper.setId(AWDevolucion.NOTA_DEBITO_NUMERO);
                textHelper.render(request,out);
            }
            catch(HelperException re){
                out.println("ERROR " + re.getMessage());
            }
            %>
        </td>
        <td width="146">Fecha</td>
        <td width="200">
            <%
            try {
                session.setAttribute(AWDevolucion.FECHA_NOTA_DEBITO, "" );
                TextHelper textHelper = new TextHelper();
                textHelper.setId(AWDevolucion.FECHA_NOTA_DEBITO);
                textHelper.setNombre(AWDevolucion.FECHA_NOTA_DEBITO);
                textHelper.setCssClase("camposformtext");
                textHelper.render(request,out);
            }
            catch(HelperException re){
                out.println("ERROR " + re.getMessage());
            }
            %>
            <a href="javascript:NewCal('<%=AWDevolucion.FECHA_NOTA_DEBITO%>','ddmmmyyyy',true,24)">')"/></a>
        </td>
    </tr>

```

```

<tr>
    <td>&nbsp;</td>
    <td>Código Banco:</td>
    <td width="211">
        <% try {
            List bancos = (List) session.getServletContext().getAttribute(WebKeys.LISTA_BANCOS);
            ListaElementoHelper bancosHelper = new ListaElementoHelper();
            if(bancos == null){
                bancos = new ArrayList();
            }
            List elemList = new ArrayList();
            ListIterator iter = bancos.listIterator();
            while (iter.hasNext()){
                Banco banco = ((Banco)iter.next());
                //elemList.add(new ElementoLista( banco.getNombre(), banco.getNombre() ) ); //provisional
                elemList.add(new ElementoLista( banco.getIdBanco(), banco.getNombre() ) );
            }
            bancosHelper.setTipos(elemList);
            bancosHelper.setNombre(AWDevolucion.NOTA_DEBITO_BANCO);
            bancosHelper.setCssClase("camposformtext");
            bancosHelper.setId(AWDevolucion.NOTA_DEBITO_BANCO);
            bancosHelper.render(request,out);
        }
        catch(HelperException re){
            out.println("ERROR " + re.getMessage());
        }
        %>
    </td>
    <td>Valor:</td>
    <td>
        <%
        String valorNotaDebito = (String)session.getAttribute(AWDevolucion.NOTA_DEBITO_VALOR);
        if(valorNotaDebito == null){
            session.setAttribute(AWDevolucion.NOTA_DEBITO_VALOR, ""
+format.format(absValorDevolver));
        }
        try {
            TextHelper textHelper = new TextHelper();
            textHelper.setNombre(AWDevolucion.NOTA_DEBITO_VALOR);
            textHelper.setCssClase("camposformtext");
            textHelper.setSize('25');
            textHelper.setId(AWDevolucion.NOTA_DEBITO_VALOR);
            textHelper.setReadonly(true);
            textHelper.render(request,out);
        }
        catch(HelperException re){
            out.println("ERROR " + re.getMessage());
        }
        %>
    </td>
</tr>

</table>

```

**Despues de la tabla de tipo de pago efectivo se agregó:**

```

<script>
    if(document.all.<%=AWDevolucion.FORMA_PAGO%>){
        var tipoFormaPago =
document.all.<%=AWDevolucion.FORMA_PAGO%>.options[document.all.<%=AWDevolucion.FORMA_PAGO%>.selectedIndex].value;
        if(tipoFormaPago != "SIN_SELECCIONAR"){
            mostrarFormaPago(tipoFormaPago);
        }
    }
</script>

```

### Reporte 25: Informe Recaudos del día

El cambio que se realizo en este reporte se registro en el requerimiento 102

### Reporte 18: Informe de Devoluciones pagadas en Cheque

#### Modificación Query reporte-18.jrxml

##### Antes

```
/*REPORTE-18.INFORME DE DEVOLUCIONES PAGADAS EN CHEQUE*/
/*EKA-2008 SE APLICA REGLA EN SUBQUERY*/
```

```

SELECT inner_1.turno_id AS Radicacion,
       inner_1.Cheque AS Cheque_No,
       inner_1.liquidacion_valor AS Valor,
       inner_1.pago_fecha AS Fecha_Pago,
       inner_1.pago_id_usuario AS Cajero, -- usuario liquidacion
       inner_1.proceso_nombre AS Proceso,
       inner_1.Valor_derechos AS Derechos,
       inner_1.Valor_Impuestos AS Impuestos,
       INITCAP(inner_1.proceso_nombre) AS CF_PROCESO,
       inner_1.vlr_certificados_a_pagar,
       inner_1.vlr_derechos_a_pagar,
       inner_1.id_resolucion,
       inner_1.fecha_resolucion,
       inner_1.solicitantes,
       inner_1.vr_efectivo,
       inner_1.vr_consignacion,
       inner_1.vr_cheque
  FROM (
    SELECT turno_asociado.id_solicitud sola,
           (SELECT fnc_ciudadanos_solicitantes(turno_devolucion.id_solicitud) FROM dual ) AS solicitantes,
           turno_devolucion.trno_id_workflow AS turno_id,
           documento_pago.dcpq_no_cheque AS Cheque,
           sir_op_pago.pago_num_recibo AS AS
      pago_recibo_id,
           sir_op_pago.pago_fecha_creacion AS pago_fecha,
           sir_op_liquidacion.lfdc_valor AS AS
     liquidacion_valor,
           turno_devolucion.trno_fecha_inicio AS
     turno_inicio_fecha,
           sir_op_pago.id_usuario AS pago_id_usuario,
           sir_op_proceso.prcs_nombre AS proceso_nombre,
           documento_pago.dcpq_tipo_pago AS tipo_pago,
           DECODE(sir_op_proceso.id_proceso,6,NVL (sir_op_liquidacion.lfdc_valor_derechos_rg, 0),NVL(sir_op_liquidacion.lfdc_valor,0))
     AS Valor_derechos,
           NVL (sir_op_liquidacion.lfdc_valor_impuestos_rg, 0)
             + NVL (sir_op_liquidacion.lfdc_valor_mora_rg, 0)
             + NVL (sir_op_liquidacion.lfdc_valor_otro_impuesto_rg, 0) AS
     Valor_Impuestos,
           sir_op_liquidacion.lfdc_fecha_creacion AS liquidacion_fecha,
           (CASE
             WHEN sir_op_pago.id_liquidacion IS NULL
             THEN sir_op_liquidacion.lfdc_valor
             ELSE NULL
           END
           ) AS sinpago_liquid,
           (CASE
             WHEN sir_op_pago.id_liquidacion IS NOT NULL
             THEN sir_op_liquidacion.lfdc_valor
             ELSE NULL
           END
           ) AS conpago_liquid,
           (SELECT sum(lfdc_valor)
          FROM sir_op_liquidacion
         WHERE id_solicitud=turno_asociado.id_solicitud
         AND sir_op_liquidacion.id_liquidacion>1
         AND sir_op_liquidacion.lfdc_valor>0
           ) AS vlr_certificados_a_pagar,
           (SELECT SUM(lfdc_valor)
          FROM sir_op_liquidacion
         WHERE id_solicitud=turno_asociado.id_solicitud
         AND sir_op_liquidacion.id_liquidacion=1
         AND sir_op_liquidacion.lfdc_valor>0
           ) AS vlr_derechos_a_pagar,
           (SELECT MAX(id_oficio)
          FROM sir_op_oficio
         WHERE solicitud_devolucion.id_turno_ant = sir_op_oficio.id_turno
         AND solicitud_devolucion.id_proceso_ant = sir_op_oficio.id_proceso
         AND solicitud_devolucion.id_circulo_ant = sir_op_oficio.id_circulo
         AND solicitud_devolucion.anio_ant = sir_op_oficio.anio
           ) AS id_resolucion,
           (SELECT MAX(fcio_fecha_creacion)
          FROM sir_op_oficio
         WHERE solicitud_devolucion.id_turno_ant = sir_op_oficio.id_turno
         AND solicitud_devolucion.id_proceso_ant = sir_op_oficio.id_proceso
         AND solicitud_devolucion.id_circulo_ant = sir_op_oficio.id_circulo
         AND solicitud_devolucion.anio_ant = sir_op_oficio.anio
           ) AS fecha_resolucion,
           (SELECT SUM(DECODE (sir_op_documento_pago.id_tipo_doc_pago,
                                /*3,sir_op_aplicacion_pago.plpg_valor_aplicado,
                                6,sir_op_aplicacion_pago.plpg_valor_aplicado,
                                7,sir_op_aplicacion_pago.plpg_valor_aplicado.*/
                                1, sir_op_documento_pago.dcpq_valor_documento,
                                6, sir_op_documento_pago.dcpq_valor_documento,
                                7, sir_op_documento_pago.dcpq_valor_documento,
                                null))
          FROM SIR_OP_APPLICACION_PAGO,sir_op_documento_pago
         WHERE SIR_OP_APPLICACION_PAGO.ID_SOLICITUD=turno_asociado.id_solicitud
         AND sir_op_aplicacion_pago.id_documento_pago=sir_op_documento_pago.id_documento_pago
           ) AS vr_efectivo,
           (SELECT SUM(DECODE (sir_op_documento_pago.id_tipo_doc_pago,
                                --2, sir_op_aplicacion_pago.plpg_valor_aplicado,
                                2, sir_op_documento_pago.dcpq_valor_documento,
                                null))
          FROM SIR_OP_APPLICACION_PAGO,sir_op_documento_pago
         WHERE SIR_OP_APPLICACION_PAGO.ID_SOLICITUD=turno_asociado.id_solicitud
         AND sir_op_aplicacion_pago.id_documento_pago=sir_op_documento_pago.id_documento_pago
           ) AS vr_consignacion,
           (SELECT SUM(DECODE (sir_op_documento_pago.id_tipo_doc_pago,
                                /*3,sir_op_aplicacion_pago.plpg_valor_aplicado,
                                4, sir_op_aplicacion_pago.plpg_valor_aplicado,
                                5, sir_op_aplicacion_pago.plpg_valor_aplicado,*/
                                3, sir_op_documento_pago.dcpq_valor_documento,
                                null))
          FROM SIR_OP_APPLICACION_PAGO,sir_op_documento_pago
         WHERE SIR_OP_APPLICACION_PAGO.ID_SOLICITUD=turno_asociado.id_solicitud
         AND sir_op_aplicacion_pago.id_documento_pago=sir_op_documento_pago.id_documento_pago
           )

```

```

        4, sir_op_documento_pago.dcpg_valor_documento,
        5, sir_op_documento_pago.dcpg_valor_documento,
        null))
FROM SIR_OP_APPLICACION_PAGO.sir_op_documento_pago
WHERE SIR_OP_APPLICACION_PAGO.ID_SOLICITUD=turno_asociado.id_solicitud
AND sir_op_aplicacion_pago.id_documento_pago=sir_op_documento_pago.id_documento_pago
) AS vr_cheque
FROM sir_op_turno_devolucion, sir_op_turno_asociado,
sir_op_solicitud_solicitud, devolucion, sir_op_solicitud_solicitud_asociada,
sir_op_liquidacion, sir_op_pago,
sir_op_proceso, sir_op_aplicacion_pago aplicacion_pago,
sir_op_documento_pago documento_pago, sir_op_tipo_doc_pago tipo_doc_pago
WHERE turno_devolucion.tmo_anulado = 'N'
AND turno_devolucion.id_solicitud = solicitud_devolucion.id_solicitud
AND turno_devolucion.id_circulo = solicitud_devolucion.id_circulo
AND turno_devolucion.id_proceso = solicitud_devolucion.id_proceso
AND turno_devolucion.id_circulo = $P{P_CIRCULO_ID}
AND solicitud_devolucion.id_turno_ant = turno_asociado.id_turno
AND solicitud_devolucion.id_proceso_ant = turno_asociado.id_proceso
AND solicitud_devolucion.id_circulo_ant = turno_asociado.id_circulo
AND solicitud_devolucion.anio_ant = turno_asociado.anio
AND solicitud_devolucion.id_proceso = 4
AND turno_asociado.id_solicitud = solicitud_asociada.id_solicitud
AND turno_asociado.id_circulo = solicitud_asociada.id_circulo
AND turno_asociado.id_proceso = solicitud_asociada.id_proceso
AND solicitud_asociada.id_proceso IN (1, 6)
AND solicitud_asociada.id_solicitud = sir_op_liquidacion.id_solicitud
-- @ filter="fecha liquidacion" -----
AND sir_op_pago.page_fecha BETWEEN TO_DATE($P{P_FECHA_INI}, 'YYYY-MM-DD')
AND TO_DATE($P{P_FECHA_FIN}, 'YYYY-MM-DD') + 1
-- @ filter="valor" -----
--AND sir_op_liquidacion.lqdc_valor < 0
--AND sir_op_liquidacion.id_liquidacion > 1
-----
AND sir_op_liquidacion.id_liquidacion = sir_op_pago.id_liquidacion
AND sir_op_liquidacion.id_solicitud = sir_op_pago.id_solicitud
AND sir_op_proceso.id_proceso = solicitud_asociada.id_proceso
AND sir_op_pago.id_liquidacion = aplicacion_pago.id_liquidacion
AND sir_op_pago.id_solicitud = aplicacion_pago.id_solicitud
AND aplicacion_pago.id_documento_pago = documento_pago.id_documento_pago
AND documento_pago.id_tipo_doc_pago = tipo_doc_pago.id_tipo_doc_pago
AND documento_pago.id_documento_pago = aplicacion_pago.id_documento_pago
AND aplicacion_pago.id_documento_pago=(SELECT MIN(id_documento_pago)
FROM sir_op_aplicacion_pago
WHERE id_solicitud=turno_asociado.id_solicitud)
AND tipo_doc_pago.id_tipo_doc_pago IN (3,4,5)
) inner_1
ORDER BY inner_1.proceso_nombre,inner_1.turno_id

```

Ahora

Para la modificación de este query se tuvo en cuenta el query del reporte 79

```

/*REPORTE-18,INFORME DE DEVOLUCIONES PAGADAS EN CHEQUE*/
/*EKA-2008 SE APlica REGLA EN SUBQUERY*/ 
SELECT      inner_1.turno_id          AS Radicacion,
            inner_1.n_chque        AS Cheque_No,
            inner_1.efectivo       AS vr_efectivo,
            inner_1.liquidacion_valor AS Valor,
            inner_1.pago_fecha     AS Fecha_Pago,
            inner_1.pago_usuario_id AS Cajero,
            inner_1.proceso_nombre   AS Proceso,
            case
                when inner_1.Valor_derechos<0 then inner_1.Valor_derechos *-1
                else inner_1.Valor_derechos
            end
            inner_1.Vvalor_Impuestos AS Derechos,
            inner_1.proceso_nombre   AS Impuestos,
            inner_1.proceso_nombre   AS CF_PROCESO,
            case
                when inner_1.vlr_certificados_a_pagar <0 then inner_1.vlr_certificados_a_pagar
                else inner_1.vlr_certificados_a_pagar
            end
            AS vlr_certificados_a_pagar,
            inner_1.vlr_derechos_a_pagar AS vlr_derechos_a_pagar,
            inner_1.id_resolucion       AS id_resolucion,
            to_char(inner_1.fecha_resolucion,'yyyy-dd-mm') AS fecha_resolucion,
            inner_1.solicitantes        AS solicitantes,
            case
                when inner_1.efectivo <0 then inner_1.efectivo * -1
                else inner_1.efectivo
            end
            AS vr_efectivo,
            case
                when inner_1.consignacion <0 then inner_1.consignacion * -1
                else inner_1.consignacion
            end
            AS vr_consignacion,
            case
                when inner_1.cheque <0 then inner_1.cheque * -1
                else inner_1.cheque
            end
            AS vr_cheque,
            case
                when inner_1.NOTADEBITO <0 then inner_1.NOTADEBITO * -1
                else inner_1.NOTADEBITO
            end
            AS vr_nota_debito,
            inner_1.turno_asociado      AS Turno_Asociado,
            inner_1.tipo_pago           AS tipo_pago,
            inner_1.saldo_favor         AS Saldo_Favor,
            inner_1.docs_pago           AS Docs_Pago,
            substr(inner_1.orden_pago,19,100) AS orden_pago
            SELECT (SELECT frc_ciudadanos_solicitudes(turno_devolucion_id_solicitud) FROM dual )
FROM (

```

<pre> turno_devolucion.trno_id_workflow sir_op_pago.pago_num_recibo sir_op_pago.pago_fecha_creacion turno_asociado.trno_id_workflow Turno_Asociado. CASE WHEN sir_op_proceso.prcs_nombre IS NULL THEN (SELECT SUM (DÉCODE (id_tipo_doc_pago,1,dcpq_valor_documento,0)) FROM sir_op_documento_pago,sir_op_sol_doc_pago WHERE sir_op_sol_doc_pago.id_solicitud=solicitud_devolucion.id_solicitud AND sir_op_documento_pago.id_documento_pago=sir_op_sol_doc_pago.id_documento_pago) ELSE CASE WHEN DECODE(tipo_doc_pago.id_tipo_doc_pago,1,sir_op_liquidacion.lqdc_valor,0) &lt;0 then DECODE(tipo_doc_pago.id_tipo_doc_pago,1,sir_op_liquidacion.lqdc_valor,0) * -1 ELSE DECODE(tipo_doc_pago.id_tipo_doc_pago,1,sir_op_liquidacion.lqdc_valor,0) END END efectivo, CASE WHEN sir_op_proceso.prcs_nombre IS NULL THEN (( SELECT SUM (DECODE (id_tipo_doc_pago, 3,dcpq_valor_documento, 4,dcpq_valor_documento, 5,dcpq_valor_documento,0)) FROM sir_op_documento_pago,sir_op_sol_doc_pago WHERE sir_op_sol_doc_pago.id_solicitud=solicitud_devolucion.id_solicitud AND sir_op_documento_pago.id_documento_pago=sir_op_sol_doc_pago.id_documento_pago) ELSE CASE WHEN DECODE(tipo_doc_pago.id_tipo_doc_pago, 3,sir_op_liquidacion.lqdc_valor, 4,sir_op_liquidacion.lqdc_valor, 5,sir_op_liquidacion.lqdc_valor)&lt;0 THEN DECODE(tipo_doc_pago.id_tipo_doc_pago, 3,sir_op_liquidacion.lqdc_valor, 4,sir_op_liquidacion.lqdc_valor, 5,sir_op_liquidacion.lqdc_valor,0) * -1 ELSE DECODE(tipo_doc_pago.id_tipo_doc_pago, 3,sir_op_liquidacion.lqdc_valor, 4,sir_op_liquidacion.lqdc_valor, 5,sir_op_liquidacion.lqdc_valor,0) * -1 END END cheque ,CASE WHEN sir_op_proceso.prcs_nombre IS NULL THEN (SELECT SUM (DECODE (id_tipo_doc_pago, 2,dcpq_valor_documento))) FROM sir_op_documento_pago,sir_op_sol_doc_pago WHERE sir_op_sol_doc_pago.id_solicitud=solicitud_devolucion.id_solicitud AND sir_op_documento_pago.id_documento_pago=sir_op_sol_doc_pago.id_documento_pago) ELSE CASE WHEN DECODE(tipo_doc_pago.id_tipo_doc_pago, 2,sir_op_liquidacion.lqdc_valor,0)&lt;0 THEN DECODE(tipo_doc_pago.id_tipo_doc_pago, 2,sir_op_liquidacion.lqdc_valor,0) ELSE DECODE(tipo_doc_pago.id_tipo_doc_pago, 2,sir_op_liquidacion.lqdc_valor,0) * -1 END END CASE WHEN sir_op_proceso.prcs_nombre IS NULL THEN (SELECT SUM (DECODE (id_tipo_doc_pago, 10,dcpq_valor_documento))) FROM sir_op_documento_pago,sir_op_sol_doc_pago WHERE sir_op_sol_doc_pago.id_solicitud=solicitud_devolucion.id_solicitud AND sir_op_documento_pago.id_documento_pago=sir_op_sol_doc_pago.id_documento_pago) ELSE CASE WHEN DECODE(tipo_doc_pago.id_tipo_doc_pago, 10,sir_op_liquidacion.lqdc_valor,0)&lt;0 THEN DECODE(tipo_doc_pago.id_tipo_doc_pago, 10,sir_op_liquidacion.lqdc_valor,0) ELSE DECODE(tipo_doc_pago.id_tipo_doc_pago, 10,sir_op_liquidacion.lqdc_valor,0) * -1 END END notas_debito, documento_pago.dcpq_no_cheque n_cheque, CASE WHEN sir_op_liquidacion.lqdc_valor &lt; 0 THEN sir_op_liquidacion.lqdc_valor * -1 ELSE sir_op_liquidacion.lqdc_valor END liquidacion_valor, DECODE(sir_op_proceso.id_proceso,6,NVL (sir_op_liquidacion.lqdc_valor_derechos_rg, 0), 2,0, 5,0, 10,0, NVL (sir_op_liquidacion.lqdc_valor, 0)) Valor_derechos, NVL (sir_op_liquidacion.lqdc_valor_impuestos_rg, 0) + NVL (sir_op_liquidacion.lqdc_valor_mora_rg, 0) + NVL (sir_op_liquidacion.lqdc_valor_otro_impuesto_rg, 0) Valor_Impuestos, sir_op_liquidacion.lqdc_fecha_creacion liquidacion_fecha, turno_devolucion.trno_fecha_inicio turno_inicio_fecha, sir_op_pago.id_usuario pago_usuario_id, sir_op_proceso.prcs_nombre proceso_nombre, (CASE WHEN sir_op_pago.id_liquidacion IS NULL THEN sir_op_liquidacion.lqdc_valor ELSE NULL END ) (CASE WHEN sir_op_pago.id_liquidacion IS NOT NULL THEN sir_op_liquidacion.lqdc_valor ELSE NULL END ) </pre>	turno_id, pago_recibo_id, pago_fecha,
--	---------------------------------------

```

END)      compago_liquid,
(decode(sir_op_proceso.id_proceso,
6, (SELECT sum(lqdc_valor + nvl(lqdc_valor_mv,0))
     FROM sir_op_liquidacion
     WHERE id_solicitud=turno_devolucion.id_solicitud
     ,0)) vlr_certificados_a_pagar
     ,(SELECT sum(lqdc_valor)
     FROM sir_op_liquidacion
     WHERE id_solicitud=turno_asociado.id_solicitud
     AND sir_op_liquidacion.id_liquidacion=1
     AND sir_op_liquidacion.lqdc_valor>0
     ) vlr_derechos_a_pagar,
documento_pago.dcpq_tipo_pago tipo_pago,
(SELECT MAX(fcio_numero)
FROM sir_op_oficio
WHERE turno_devolucion.id_turno = sir_op_oficio.id_turno
AND turno_devolucion.id_proceso = sir_op_oficio.id_proceso
AND turno_devolucion.id_circulo = sir_op_oficio.id_circulo
AND turno_devolucion.anio = sir_op_oficio.anio) id_resolucion,
(SELECT MAX(fcio_fecha_creacion)
FROM sir_op_oficio
WHERE turno_devolucion.id_turno = sir_op_oficio.id_turno
AND turno_devolucion.id_proceso = sir_op_oficio.id_proceso
AND turno_devolucion.id_circulo = sir_op_oficio.id_circulo
AND turno_devolucion.anio = sir_op_oficio.anio) fecha_resolucion,
CASE
    WHEN sir_op_liquidacion.lqdc_valor_saldo_favor < 0 THEN
        --documento_pago.DCPG_SALDO_DOCUMENTO * -1
        sir_op_liquidacion.lqdc_valor_saldo_favor * -1
    ELSE
        --documento_pago.DCPG_SALDO_DOCUMENTO
        sir_op_liquidacion.lqdc_valor_saldo_favor
END      saldo_favor,
fn_NumDocsPago_Devolucion(turno_devolucion.id_solicitud) docs_pago,
sir_op_pago.pago_concepto orden_pago
FROM
sir_op_turno turno_devolucion,
sir_op_turno turno_asociado,
sir_op_solicitud solicitud_devolucion,
sir_op_solicitud solicitud_asociada,
sir_op_liquidacion,
sir_op_pago,
sir_op_proceso,
sir_op_aplicacion_pago aplicacion_pago,
sir_op_documento_pago documento_pago,
sir_op_tipo_doc_pago tipo_doc_pago
WHERE turno_devolucion.trmo_anulado = 'N'
AND turno_devolucion.trmo_id_fase= 'FINALIZADO'
AND turno_devolucion.id_solicitud = solicitud_devolucion.id_solicitud
AND turno_devolucion.id_circulo = solicitud_devolucion.id_circulo
AND turno_devolucion.id_proceso = solicitud_devolucion.id_proceso
AND turno_devolucion.id_circulo = $P{P_CIRCULO_ID}
AND solicitud_devolucion.id_turno_ant = turno_asociado.id_turno  (+)
AND solicitud_devolucion.id_proceso_ant = turno_asociado.id_proceso (+)
AND solicitud_devolucion.id_circulo_ant = turno_asociado.id_circulo (+)
AND solicitud_devolucion.anio_ant = turno_asociado.anio  (+)
AND solicitud_devolucion.id_proceso = 4
AND turno_asociado.id_solicitud = solicitud_asociada.id_solicitud (+)
AND turno_asociado.id_circulo= solicitud_asociada.id_circulo  (+)
AND turno_asociado.id_proceso = solicitud_asociada.id_proceso  (+)
--AND solicitud_asociada.id_proceso IN (1, 6)
AND solicitud_devolucion.id_solicitud = sir_op_liquidacion.id_solicitud
-- @ filter="fecha_liquidacion"
-- @ filter="valor"
-- BETWEEN TO_DATE( $P{P_FECHA_INI}, 'YYYY-MM-DD')
-- AND TO_DATE( $P{P_FECHA_FIN}, 'YYYY-MM-DD') + 1
-- @ filter="valor"
-- AND sir_op_liquidacion.lqdc_valor < 0
-- AND sir_op_liquidacion.id_liquidacion > 1
-- AND sir_op_liquidacion.id_liquidacion = sir_op_pago.id_liquidacion
-- AND sir_op_liquidacion.id_solicitud = sir_op_pago.id_solicitud
-- AND sir_op_proceso.id_proceso(+) = solicitud_asociada.id_proceso
-- AND sir_op_pago.id_liquidacion = aplicacion_pago.id_liquidacion
-- AND sir_op_pago.id_solicitud = aplicacion_pago.id_solicitud
-- AND aplicacion_pago.id_documento_pago = documento_pago.id_documento_pago
-- AND documento_pago.id_tipo_doc_pago = tipo_doc_pago.id_tipo_doc_pago
-- AND documento_pago.id_documento_pago = aplicacion_pago.id_documento_pago
-- AND solicitud_devolucion.sltc_aprobacion=1
AND tipo_doc_pago.id_tipo_doc_pago IN (3,4,5,10)
AND aplicacion_pago.id_documento_pago=
        SELECT MAX(id_documento_pago)
        FROM sir_op_aplicacion_pago
        WHERE id_solicitud=turno_devolucion.id_solicitud )
) inner_1
ORDER BY inner_1.proceso_nombre,inner_1.turno_id

Concepto: CERTIFICADOS


| Radicación            | Fecha de Pago    | Turno Asociado | Valor       | Solicitantes       | Derechos    | Impuestos | Certificados | Número Resolución | Fecha      | Efectivo | Forma de pago | Cheque | Consignación | Nota Débito |
|-----------------------|------------------|----------------|-------------|--------------------|-------------|-----------|--------------|-------------------|------------|----------|---------------|--------|--------------|-------------|
| 2009-999-4-59         | 10/10/09 9:19 AM | CERTIFICADOS   | \$11,840.00 | 85281513, PRUEBAS: | \$11,840.00 | \$0.00    | \$0.00       | 1234              | 2009-15-10 | 0.00     | 0.00          | 0.00   | 0.00         | 11,840.00   |
| Totales CERTIFICADOS: |                  |                | \$11,840.00 |                    | \$11,840.00 | \$0.00    | \$0.00       |                   |            |          |               |        |              |             |


```

**Reporte 21: Informe de Devoluciones Ingresadas  
Modificación Query reporte-21.jrxml**

/\* AHERRENO 19/10/2009  
Se modifica forma de traer el número relacionado al pago ya sea Consignación, Cheque o Nota Débito\*/  
DECODE (documento\_pago.ID\_TIPO\_DOC\_PAGO,  
3,documento\_pago.DCPG\_NO\_CHEQUE||' - CH',

```

4, documento_pago.DCPG_NO_CHEQUE||' - CH',
5, documento_pago.DCPG_NO_CHEQUE||' - CH',
2, documento_pago.DCPG_NO_CONSIGNACION||' - CO',
10,documento_pago.DCPG_NO_NOTA_DEBITO||' - ND',") docs_pago,
--fnc_NumDocsPago_Devolucion(turbo_devolucion.id_solicitud) docs_pago,
/*********************************************************/

```

Modificación Columna Número Consignación/Cheque: Se agrego a la columna el tipo de pago.

Concepto: REGISTRO

Radicación	Fecha	Turbo asociado	Número Consignacion/ Cheque/Nota Debito	Solicitantes
2009-999-4-50	14/10/2009	2009-999-6-318	987654 - ND	85281153, PRUEBAS;
2009-999-4-60	16/10/2009	2009-999-6-318		85281573, PRUEBAS;
2009-999-4-62	16/10/2009	2009-999-6-285	95136842 - CH	85281633, PRUEBAS;

Al final del reporte se desplegará los tipos de pagos:

ND - Nota Débito  
CH - Cheque  
CO - Consignación

## Reporte 79: Informe de Devoluciones Pagadas

### Modificación Query reporte-79.jrxml

```

/*AHERRENO 19/10/2009
Se agrega el tipo de pago 10: Nota Débito */
case
when inner_1.NOTA_DEBITO <0 then inner_1.NOTA_DEBITO * -1
else inner_1.NOTA_DEBITO
end
NOTA_DEBITO,
/*********************************************************/
/*AHERRENO 19/10/2009
Se agrega el tipo de pago 10: Nota Débito */
CASE WHEN sir_op_proceso.prcs_nombre IS NULL THEN
(SELECT SUM (DECODE (id_tipo_doc_pago,
10,dcpq_valor_documento) )
FROM sir_op_documento_pago,sir_op_sol_doc_pago
WHERE sir_op_sol_doc_pago.id_solicitud=solicitud_devolucion.id_solicitud
AND sir_op_documento_pago.id_documento_pago=sir_op_sol_doc_pago.id_documento_pago)
ELSE
CASE WHEN DECODE(tipo_doc_pago.id_tipo_doc_pago,
10,sir_op_liquidacion.lqdc_valor,0)<0 THEN
DECODE(tipo_doc_pago.id_tipo_doc_pago,
10,sir_op_liquidacion.lqdc_valor,0)
ELSE
DECODE(tipo_doc_pago.id_tipo_doc_pago,
10,sir_op_liquidacion.lqdc_valor,0)*-1
END
END
nota_debito,
/*********************************************************/
/* AHERRENO 19/10/2009
Se modifica forma de traer el número relacionado al pago ya
sea Consignación, Cheque o Nota Débito*/
DECODE (documento_pago.ID_TIPO_DOC_PAGO,
3, documento_pago.DCPG_NO_CHEQUE,
4, documento_pago.DCPG_NO_CHEQUE,
5, documento_pago.DCPG_NO_CHEQUE,
2, documento_pago.DCPG_NO_CONSIGNACION,
10,documento_pago.DCPG_NO_NOTA_DEBITO,") docs_pago,
--fnc_NumDocsPago_Devolucion(turbo_devolucion.id_solicitud) docs_pago,
/*********************************************************/

```

Concepto: CERTIFICADOS

Radicación	Fecha de Pago	Turbo Asociado	Num Consignacion/ Nota Débito	Vr. Derechos	Vr. Impuestos	Vr	Pago Exceso	Valor Total	Número / Fecha Resolución	Efectivo	Forma de pago	Orden - Fecha pago	Solicitantes
2009-999-4-59	16/10/2009 09:10:22	2009-999-1-522	4101354	\$ 11,840	\$ 0	\$ 0	\$ 0	\$ 11,840	1234 / 2009-15-10	\$ 0	\$ 0	\$ 11,840	456 DEL 16/10/2009
<b>Totales CERTIFICADOS:</b>				\$ 11,840	\$ 0	\$ 0	\$ 0	\$ 11,840					

## 111 Liquidacion Impuesto Hipoteca SL de Cuantia

### REQUERIMIENTO

Se debe corregir la funcionalidad de Radicación de documentos, en asociar actos y tarifas, dado que al liquidar el acto Hipoteca sin límite de cuantía, identificado con el código 15 por concepto de impuesto de registro no esta liquidando de manera correcta.

Actualmente, para este tipo de acto si se selecciona el impuesto con tarifa normal, el impuesto se esta liquidando como acto sin cuantía, es decir, 4 SMDL, y se debe liquidar como un acto con cuantía, es decir, el 1%, sobre el valor del acto a registrar.

## **DESARROLLO**

Esto se debe modificar por pantallas administrativas, opción "Administración de Tipos de Actos"; pero la pantalla al guardar genera una excepción.

Se modifica el método que guarda esta pantalla para que no genere la excepción.

clase:C:\WORK\SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDOVariablesOperativasDAO.java

método: public TipoActoPk updateTipoActo(TipoActo acto, Usuario usuario)

línea modificada: 2146

```
/**  
 * @author Fernando Padilla  
 * @change caso mantis 2859, Se hace comentario sobre este código para que no  
 * borre los datos del tipo de acto en la base de datos.  
 */  
//se eliminan los datos del tipo de acto en la base de datos.  
//asociados con el tipo de acto.  
/*Iterator it2 = actor.getTiposDerechosRegistrales().iterator();  
while (it2.hasNext()) {  
    TipoActoDerechoRegistralEnhanced tipoActDerToDel =  
        (TipoActoDerechoRegistralEnhanced) it2.next();  
    pm.deletePersistent(tipoActDerToDel);  
}  
*/  
Línea modificada: 2167  
/**@author Fernando Padilla  
 * @change caso mantis 2859, Se hace comentario sobre este código para que no  
 * cree nuevos los datos del tipos de actos en la base de datos.*/  
 //List derRegs = newData.getTiposDerechosRegistrales();  
 //Iterator it = derRegs.iterator();  
//Se intentan obtener cada uno de los tipos de derechos registrales,  
//asociados con el tipo de acto.  
/*while (it.hasNext()) {  
    TipoActoDerechoRegistralEnhanced tipoActDer =  
        (TipoActoDerechoRegistralEnhanced) it.next();  
    TipoDerechoRegEnhanced tipoDer = tipoActDer.getTipoDerechoRegistral();  
    TipoDerechoRegEnhancedPk tipoDerId = new TipoDerechoRegEnhancedPk();  
    tipoDerId.idTipoDerechoReg = tipoDer.getIdTipoDerechoReg();  
    TipoDerechoRegEnhanced tipoDerr = this.getTipoDerechoRegByID(tipoDerId, pm);  
  
    // Si no existe un tipo de derecho regstral con el identificador del objeto  
    // pasado como parámetro, se genera una excepción.  
    if (tipoDerr == null) {  
        throw new DAOException(  
            "No encontró el tipo de Derecho regstral con el ID: "  
            + tipoDer.getIdTipoDerechoReg());  
    }  
    tipoActDer.setTipoDerechoRegistral(tipoDerr);  
    tipoActDer.setTipoActo(actor);  
    pm.makePersistent(tipoActDer);  
}  
*/
```

## **112 Relación de turnos en desorden**

### **REQUERIMIENTO**

Relación de turnos en desorden. Turno 2009-350-1347. La usuaria nos informa que en la relación 2009-350-1347 los turnos están saliendo en desorden. La usuaria nos informa que en la relación 2009-350-1347 los turnos están saliendo en desorden como se puede verificar, se adjunta relación. Ibagué.

## **DESARROLLO**

El archivo que se está ejecutando para generar este reporte es el archivo reporte-35.jasper el cual tiene configurado el ordenamiento por el campo turno.trno\_fecha\_creacion el cual se modificó por turno.trno\_fecha\_inicio "FECHA RADICACION" y se agregó el ordenamiento también del campo turno.trno\_id\_workflow "TURNO". Se adjunta reporte; verificar si el usuario lo necesita para enviarlo; y se les aclara que el cambio será reflejado en el próximo despliegue en producción. El archivo adjunto es reporte-Ibagué-2009-350-1347.pdf

## 114 Reporte turnos anulados

### **REQUERIMIENTO**

Nos informa la funcionaria Venovit Mejia e informa que al generar el reporte de anulados del día 7-09-09 el sistema no asocia el turno 2009-350-1-70865. Se genera el reporte diario anulados del día 7-09-09 y se puede apreciar que el turno 2009-350- 1-70865 el cual lo anularon en la oficina de registro el día 7-09-09 no aparece relacionado.

Por favor verificar el reporte, se adjunta el reporte generado. Ibagué.

### **DESARROLLO**

Se modifica script Reporte 156 - Diario Anulados del Día:

```
/*Esta validación se agrego ya que cuando se consulta en la tabla  
SIR_OP_TURNO_HISTORIA la búsqueda se debe realizar por 4 campos:  
ID_TURNO, ID_CIRCULO, ANIO Y ID_PROCESO*/  
*****  
AND h.anio = sir_op_turno_historia.anio  
*****  
*****/
```

Se adjunta pdf reporte-156-Ibague-07-09-2009.pdf; este cambio será reflejado en el próximo despliegue.

## 117 Vista anotaciones en calificación

### **REQUERIMIENTO**

Vista anotaciones en calificación. Actualmente en algunos turnos cuando llegan a la fase calificación, al dar clic en inscripción el sistema muestra un número de anotaciones demasiado grande cuando realmente el folio solo tiene un número relativamente pequeño, como ejemplo el sistema muestra que el folio tiene 600 anotaciones pero realmente tiene 10, como referencia está el caso 12124.

Se requiere corregir y revisar este problema a nivel general para poder solucionar el problema de vista que tiene el folio en la fase calificación. Es necesario tener en cuenta que al tratar de hacer alguna modificación al folio el sistema queda cargando la información hasta que se sale de la sesión, al salir y volver a entrar al folio ya muestra la información correcta.

Por favor escalar este caso a desarrollo y crear acta de requerimiento formal para este caso general.

### **DESARROLLO**

El número de anotaciones estaba siendo guardado en una variable estática por lo cual el valor era compartido entre los que ejecuten la aplicación en el mismo servidor.

Solución: Se guardó el número de anotaciones en la session y como atributo del evento "EvnPaginadorAnotaciones"

Se modificaron los siguientes archivos:

CLASE:

gov.sir.core.negocio.acciones.comun.ANPaginadorAnotaciones.java

Se comentó la importación:

import gov.sir.core.negocio.modelo.AnotacionesFolio;

**Método:**

```
private EventoRespuesta consultarAnotacionesDefinitivasFolio(EvnPaginadorAnotaciones evento)
```

**Código:**

```
if(AnotacionesFolio.getNumAnotacionesTotalV()>0){
    ev= new EvnRespPaginadorAnotaciones(anotacionesDefinitivas, null,
    AnotacionesFolio.getNumAnotacionesTotalV()-AnotacionesFolio.getNumAnotacionesFolio(),
    EvnRespPaginadorAnotaciones.CONSULTAR_ANOTACIONES_FOLIO, evento.getNombrePaginador(),
    evento.getNombreResultado());
}
```

**Por:**

```
if(evento.getNumAnotacionesTotalV()>0){
    ev= new EvnRespPaginadorAnotaciones(anotacionesDefinitivas, null, evento.getNumAnotacionesTotalV()-
    evento.getNumAnotacionesDesdeV(), EvnRespPaginadorAnotaciones.CONSULTAR_ANOTACIONES_FOLIO,
    evento.getNombrePaginador(), evento.getNombreResultado());
}
```

**CLASE:**

```
gov.sir.core.web.acciones.registro.AWCalificacion.java
```

**Se comentó la importación:**

```
import gov.sir.core.negocio.modelo.AnotacionesFolio;
```

**Método:**

```
public Evento perform(HttpServletRequest request)
```

**Código:**

```
if(numAnotacionesDesde==null){
    AnotacionesFolio.setNumAnotacionesFolio((new Integer(0).intValue()));
}
else{
    try{
        int tNumAnotaciones =
        Integer.parseInt(request.getParameter(WebKeys.CAMPO_NUM_ANOTACIONES_TOTAL));
        if(tNumAnotaciones>0){
            AnotacionesFolio.setNumAnotacionesTotalV(tNumAnotaciones);
        }
        else{
            AnotacionesFolio.setNumAnotacionesTotalV(0);
        }
        AnotacionesFolio.setNumAnotacionesFolio(Integer.parseInt(numAnotacionesDesde));
        if(AnotacionesFolio.getNumAnotacionesFolio()<1 ||
        AnotacionesFolio.getNumAnotacionesFolio()>tNumAnotaciones){
            AnotacionesFolio.setNumAnotacionesFolio(0);
        }
        else{
            AnotacionesFolio.setNumAnotacionesFolio(AnotacionesFolio.getNumAnotacionesFolio()-1);
        }
    }catch(NumberFormatException eNum){
        AnotacionesFolio.setNumAnotacionesFolio(0);
        AnotacionesFolio.setNumAnotacionesTotalV(0);
    }
}

```

**Por:**

```
if(numAnotacionesDesde==null){
    request.getSession().setAttribute(WebKeys.CAMPO_NUM_ANOTACIONES DESDE, 0);
    request.getSession().setAttribute(WebKeys.CAMPO_NUM_ANOTACIONES_TOTAL, 0);
}
else{
    try{
        int tNumAnotaciones =
        Integer.parseInt(request.getParameter(WebKeys.CAMPO_NUM_ANOTACIONES_TOTAL));
        if(tNumAnotaciones>0){
            request.getSession().setAttribute(WebKeys.CAMPO_NUM_ANOTACIONES_TOTAL,
            tNumAnotaciones);
        }
        else{
            request.getSession().setAttribute(WebKeys.CAMPO_NUM_ANOTACIONES_TOTAL, 0);
        }
        int numAnotacionesDesdeI = Integer.parseInt(numAnotacionesDesde);
        if(numAnotacionesDesdeI<1 || numAnotacionesDesdeI>tNumAnotaciones){
            request.getSession().setAttribute(WebKeys.CAMPO_NUM_ANOTACIONES DESDE, 0);
        }
        else{
            request.getSession().setAttribute(WebKeys.CAMPO_NUM_ANOTACIONES DESDE,
            numAnotacionesDesdeI - 1);
        }
    }catch(NumberFormatException eNum){
        request.getSession().setAttribute(WebKeys.CAMPO_NUM_ANOTACIONES DESDE, 0);
        request.getSession().setAttribute(WebKeys.CAMPO_NUM_ANOTACIONES_TOTAL, 0);
    }
}
```

<b>CLASE:</b>
<b>gov.sir.core.web.acciones.comun.EvnPAGINADORANOTACIONES.java</b>
<b>Se comentó la importación:</b>
<i>import gov.sir.core.negocio.modelo.AnotacionesFolio;</i>
<b>Método:</b>
<i>private Evento consultarAnotacionesFolio(HttpServletRequest request)</i>
<b>Código:</b>
<i>if(AnotacionesFolio.getNumAnotacionesFolio()&gt;0 &amp;&amp; AnotacionesFolio.getNumAnotacionesTotalV()&gt;0){ e = new EvnPAGINADORANOTACIONES(usuarioAuriga, folioID, AnotacionesFolio.getNumAnotacionesFolio(), AnotacionesFolio.getNumAnotacionesTotalV(), EvnPAGINADORANOTACIONES.CONSULTAR_ANOTACIONES_FOLIO, usuarioNeg, consultarAnotacionesDefinitivas); } <b>Por:</b> Integer numAnotaDesdeV = (Integer)request.getSession().getAttribute(WebKeys.CAMPO_NUM_ANOTACIONES DESDE); Integer numAnotaTotalV = (Integer)request.getSession().getAttribute(WebKeys.CAMPO_NUM_ANOTACIONES TOTAL); if(numAnotaDesdeV != null &amp;&amp; numAnotaTotalV != null &amp;&amp; numAnotaDesdeV&gt;0 &amp;&amp; numAnotaTotalV&gt;0){ e = new EvnPAGINADORANOTACIONES(usuarioAuriga, folioID, numAnotaDesdeV, numAnotaTotalV, EvnPAGINADORANOTACIONES.CONSULTAR_ANOTACIONES_FOLIO, usuarioNeg, consultarAnotacionesDefinitivas); e.setAnotacionesDesdeV(numAnotaDesdeV); e.setNumAnotacionesTotalV(numAnotaTotalV); }</i>

<b>CLASE:</b>
<b>gov.sir.core.eventos.comun.EvnPAGINADORANOTACIONES.java</b>
<b>Se agregaron los atributos:</b>
<i>private Integer anotacionesDesdeV = 0; private Integer numAnotacionesTotalV = 0;</i>
<b>Y los métodos:</b>
<i>public Integer getAnotacionesDesdeV() public void setAnotacionesDesdeV(Integer anotacionesDesdeV) public IntegergetNumAnotacionesTotalV() public void setNumAnotacionesTotalV(Integer numAnotacionesTotalV)</i>

<b>CLASE:</b>
<b>gov.sir.core.web.helpers.comun.MostrarFolioHelper.java</b>
<b>Se comentó la importación:</b>
<i>import gov.sir.core.negocio.modelo.AnotacionesFolio;</i>
<b>Método:</b>
<i>private Evento consultarAnotacionesFolio(HttpServletRequest request)</i>
<b>Código:</b>
<i>if(AnotacionesFolio.getNumAnotacionesTotalV()&gt;0){ if(AnotacionesFolio.getNumAnotacionesTotalV()==1){ textoPostTotalAnotaciones="Anotaci&amp;oacute;n"; } out.println(" &lt;td class=""'+this.eTitulos+" width="43%"&gt;&amp;nbsp;Visualizando&amp;nbsp;" + (AnotacionesFolio.getNumAnotacionesTotalV()) AnotacionesFolio.getNumAnotacionesFolio()) +"&amp;nbsp;de&amp;nbsp;" + AnotacionesFolio.getNumAnotacionesTotalV() + "&amp;nbsp;" + textoPostTotalAnotaciones + "&lt;/td&gt;"); } <b>Por:</b> Integer numAnotacionesTotalV = (Integer)request1.getSession().getAttribute(WebKeys.CAMPO_NUM_ANOTACIONES_TOTAL); Integer numAnotacionesDesdeV = (Integer)request1.getSession().getAttribute(WebKeys.CAMPO_NUM_ANOTACIONES DESDE); if(numAnotacionesTotalV&gt;0){ if(numAnotacionesTotalV==1){ textoPostTotalAnotaciones="Anotaci&amp;oacute;n"; } out.println(" &lt;td class=""'+this.eTitulos+" width="43%"&gt;&amp;nbsp;Visualizando&amp;nbsp;" + (numAnotacionesTotalV - numAnotacionesDesdeV) + "&amp;nbsp;de&amp;nbsp;" + numAnotacionesTotalV + "&amp;nbsp;" + textoPostTotalAnotaciones + "&lt;/td&gt;"); }</i>

<b>CLASE:</b>
<b>gov.sir.core.web.helpers.comun.PaginadorAvanzadoHelper.java</b>
<b>Se comentó la importación:</b>
<i>import gov.sir.core.negocio.modelo.AnotacionesFolio;</i>
<b>Y se agregó la importación:</b>
<i>import gov.sir.core.web.WebKeys;</i>
<b>Método:</b>
<b>public Evento perform(HttpServletRequest request)</b>
<b>Código:</b>
<i>if(AnotacionesFolio.getNumAnotacionesTotalV()&gt;0){     out.write("    var vTempAnotacion = document."+nombreForm+".ORDEN_ANOTACION_ACTUAL.value     - " + AnotacionesFolio.getNumAnotacionesFolio() + ";" +r+n);     out.write("        if(vTempAnotacion&lt;1){r+n};         out.write("            alert("Anotacion fuera del rango.");r+n);         out.write("            document."+nombreForm+".ORDEN_ANOTACION_ACTUAL.value=\\\";" +r+n);         out.write("        else{r+n};         out.write("            document."+nombreForm+".ORDEN_ANOTACION_ACTUAL.value = vTempAnotacion;r+n);     } }  <b>Por:</b> Integer numAnotacionesTotalV = (Integer)request.getSession().getAttribute(WebKeys.CAMPO_NUM_ANOTACIONES_TOTAL); Integer numAnotacionesDesdeV = (Integer)request.getSession().getAttribute(WebKeys.CAMPO_NUM_ANOTACIONES DESDE); if(numAnotacionesTotalV&gt;0){     out.write("        var vTempAnotacion = document."+nombreForm+".ORDEN_ANOTACION_ACTUAL.value     - " + numAnotacionesDesdeV + ";" +r+n);     out.write("        if(vTempAnotacion&lt;1){r+n};         out.write("            alert("Anotacion fuera del rango.");r+n);         out.write("            document."+nombreForm+".ORDEN_ANOTACION_ACTUAL.value=\\\";" +r+n);         out.write("        else{r+n};         out.write("            document."+nombreForm+".ORDEN_ANOTACION_ACTUAL.value = vTempAnotacion;r+n); }  <b>Código:</b> if(AnotacionesFolio.getNumAnotacionesTotalV()&gt;0){     out.write("        /r+n); }  <b>Por:</b> if(numAnotacionesTotalV&gt;0){     out.write("        /r+n); }</i>

## 129 Oficina Internacional Turnos Registro

### **REQUERIMIENTO**

Oficina Internacional Turnos Registro. Al radicar un turno de registro como oficina internacional, se despliega un campo editable para manualmente escribir el nombre de dicha oficina internacional. Cuando llega el turno a calificación y se crean anotaciones en los folios asociados, al imprimir el formulario de calificación la oficina de origen internacional no aparece en el imprimible. Se debe corregir este dato que corresponde al encabezado de los turnos y que da origen a los datos de las anotaciones para que se reflejen en el imprimible del formulario de calificación. Se debe escalar el caso a desarrollo para su solución general, por favor crear acta de requerimiento a desarrollo para este caso. Id Glpi 12806.

### **DESARROLLO**

Análisis: No se estaba imprimiendo el nombre de la oficina internacional.

Solución: se imprime el nombre de la oficina internacional:

Clase:C:\WORK\SIR-APP\src\gov\sir\core\negocio\modelo\imprimibles\base\ImprimibleBaseFolio.java  
método:protected int imprimirAnotacion(Anotacion anota, boolean modo)  
Línea código 790:

```

    /**
     * @author: Fernando Padilla
     * @change: Caso mantis 2653, se imprime el nombre de la oficina internacional.
     */
    else{
        if(documento.getOficinaInternacional() != null){
            nombreOficDoc += documento.getOficinaInternacional();
        }
    }
}

```

### 130 reporte 99 ingreso diario por documentos

#### REQUERIMIENTO

Se presentan dificultades para el cuadre del día 18 de septiembre 2009 por la siguiente razón el turno de documentos 2009-280-6-17396, con recibo de caja 39743668, se refiere a una nueva entrada, tal como lo refleja dicho recibo con valor total de -0-.

Al cargar del reporte los derechos del recibo os presenta como cifras negativas de -34.650 -85.000.- ID GLPI 12796 GRUPO ARMENIA

#### DESARROLLO

Se revisa el procedimiento sp\_reporte\_99 del reporte 99 ya que se están generando valores inconsistentes cuando hay turnos anteriores con actos del mismo tipo.

#### SOLUCIÓN:

Se modificó el sp\_reporte\_99

```

/*
    GCABRERA 06/10/2009
    SE ADICIONA UN CURSOR PARA CARGAR LOS ACTOS DEL TURNO ANTERIOR.
*/
CURSOR c_act_ant (p_sol VARCHAR2) IS
    SELECT SUM(acto_valor) acto_valor, SUM(acto_cuantia) acto_cuantia, id_tipo_derecho_reg, id_tipo_acto
    FROM SIR_OP_ACTO
    WHERE id_solicitud = p_sol
    AND id_liquidacion = 1
    GROUP BY id_tipo_acto, id_tipo_derecho_reg;
/*
    GCABRERA 06/10/2009
    SE ACREDITA VARIABLE TIPO CURSOR DE LOS ACTOS DEL TURNO ANTERIOR
*/
CDATOS c_act_ant%ROWTYPE;
/*
    GCABRERA 06/10/2009
    SE MODIFICA LA FUNCIÓN PARA OBTENER LOS VALORES DE ACTOS RECORRIENDO EL CURSOR DE LOS
    ACTOS
    DEL TURNO ANTERIOR PASANDO POR PARAMETRO LA SOLICITUD.
    SE ADICIONA EL PARAMETRO p_id_tipo_derecho_reg
*/
FUNCTION fnc_valores_acto (p_sol VARCHAR2,p_id_tipo_derecho_reg VARCHAR2, p_tipo_acto NUMBER, p_val
NUMBER) RETURN NUMBER IS
    cur SYS_REFCURSOR;
    v_valor NUMBER(20,3);
    acto_valor NUMBER(20,3);
    acto_cuantia NUMBER(20,3);

```

### 133 Reporte 99 (Informativo)

#### REQUERIMIENTO

Turno 2009-140-6-11166. Informa Javier Dickson que en el Reporte 99: Ingresos diarios por documentos del 16 de octubre del presente año; aparece el turno 2009- 140-6-11166, CALIFICACION X-BETTY.ANAYA, aparece con el número de recibo 40822651, NO coincidiendo este con el recibo de caja original impreso: 40822656. Se generó el reporte 99 con los siguientes parámetros:

Fecha Inicial: 16/10/2009  
 Fecha Final: 16/10/2009  
 Círculo Registral: Montería

Aparece el turno 2009-140-6-11166 con el numero de recibo 40822651, al consultarlo por ver detalles del turno coincide con el numero de recibo, pero no con el numero de recibo impreso: 40822656, favor verificar que pudo haber sucedido y tomar las medidas correctivas, gracias. Montería. Id Glpi 13123.

### **DESARROLLO**

Se corrige la información inconsistente en la base de datos, se revisa el reporte 99 y el turno se visualiza con la información correcta. Scripts:

```
select * from sir_op_pago
where id_solicitud = '2009-140-6-11671'
```

```
update sir_op_pago
set pago_num_recibo = pago_last_num_recibo
where id_solicitud = '2009-140-6-11671'
```

### **DESPLIEGUE 07/10/2009 – 2.2.0**

#### **INCIDENCIAS RESUELTA**

ID GLPI	OBSERVACIONES
Pantalla administración – Reportes – Listado de Reportes	
<b>Reporte Rel01 – Reporte Genérico de Relaciones</b>	
10131	<p><b>REQUERIMIENTO (INFORMATIVO)</b>                  Informa Sandra Ramírez que al generar las relaciones 2009-070-992, 993, 994 por Reporte Rel 01: Reporte genérico de relaciones no se visualizan los turnos. Favor verificar el campo para que se visualicen los turnos. Tunja.</p>
9945	<p><b>REQUERIMIENTO (INFORMATIVO)</b>                  Amparo informa que los reportes de relaciones no muestran el número de turno. Se le indica a Amparo que nos de el número de relación para generarla, y efectivamente no sale con el número de turno. 2009-340-1072. Por favor verificar</p>
9834	<p><b>REQUERIMIENTO (INFORMATIVO)</b>                  Se comunica Sandra Ramírez e informa que al generar los reportes genéricos de las relaciones:                  2009-070-926.                  2009-070-927.                  2009-070-928.                  Aparecen incompletas, el turno no lo muestra completo. En el campo turno los turnos aparecen completos pero no aparece el consecutivo. Favor revisar debido a que puede ser un problema de vistas en los reportes. Tunja.</p>
9842	<p><b>REQUERIMIENTO (INFORMATIVO)</b>                  Informa Alberto García que al generar el Reporte de relaciones: 2009-314-670 y 2009-314-671 no se visualizan completos los turnos. Piedecuesta.</p>
<b>DESARROLLO</b>	

	<p>Se modifico el tamaño del campo TURNO y RELACION.</p> <p><b>OBSERVACIONES ETB:</b></p> <p>Estos casos fueron reportados en la mesa de ayuda en el tiempo en que se estuvieron realizando pruebas y el tiempo que se tomó para el despliegue; este requerimiento se soluciono con el despliegue (2.1.0) 08/07/2009</p>	
11190	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Se comunica la funcionaria Mónica Robledo y nos informa que al imprimir la relación de los turnos que pasan de la fase reparto de registro a calificación no está saliendo correctamente. Se tiene la relación de reparto de registro 2009-290-1912 la cual la imprimieron el día 31 de Julio del presente año, en ella salen los turnos que avanzaron de la fase reparto de registro a la fase de calificación y especifica el calificador para cada turno, al verificar la impresión del día 31 de Julio de 2009 hay turnos los cuales tienen asociados varios folios y en el reporte impreso sale el respectivo turno asociando los folios en frente, pero al momento de generar el reporte el día de hoy 04 de Agosto de 2009 de la misma relación 2009-290-1912 el sistema toma el turno de registro que tiene varios folios asociados y lo repite para cada uno de ellos, haciendo esto ver a los calificadores que tienen más turnos, lo cual no debería de ser ya que debe salir relacionado el turno a los folios que tenga asociado y no tomar el turno por cada folio.</p> <p>Anexo relación impresa del 31 de Julio de 2009 y relación de hoy 4 de Agosto de 2009. Pereira.</p> <p><b>DESARROLLO.</b></p> <p>El Reporte de Relaciones se genera con el rol Jefe Operativo y se genera por pantallas administrativas - Reportes:</p> <ul style="list-style-type: none"> <li>* Reporte 155 - Reparto a Calificación</li> <li>* Reporte REI 01 - Reporte Genérico de Relaciones</li> </ul> <p>Es de aclarar que el archivo (reporte-155.jasper) que se ejecuta desde el rol Jefe Operativo es el mismo que se ejecuta en el Reporte 155 - Reparto a Calificación y el Reporte REI 01 - Reporte Genérico de Relaciones se genera a través del archivo (reporte-97.jasper); archivos que cumplen el objetivo de desplegar la información de las relaciones, pero sus diseños son totalmente diferentes. Por tal motivo es necesario solicitarle al usuario el PDF que se esta generando con el Rol Jefe Operativo y el PDF que se esta generando desde pantallas administrativas - Reportes (del mes anterior); si identificamos que el reporte que se genero el 31 de julio del presente año (Jefe Operativo) no cumple con la configuración del reporte (Los turnos no se deben repetir); esa comparación también se realizará con el reporte generado desde pantallas administrativas. Con esta revisión se puede identificar si los archivos (.jasper) se han modificado de los servidores de</p>	

	<p>producción.</p> <p><b>MESA DE AYUDA</b></p> <p>Se indica a la funcionaria Mónica Robledo enviar el imprimible que se genera del reparto de registro con el rol de jefe operativo, nos informa que en el momento no tiene turnos a repartir por lo tanto no puede enviar el imprimible, la funcionaria informa que al momento de tomar los turnos para repartirlos y verificar el reparto no lo imprime, ella verifica la información y solo le da aceptar, por lo tanto quedó en enviar el reparto impreso para confrontar información y así poder tramitar el caso.</p> <p>Se cierra el caso y cuando se reciba la información que se pidió a la funcionaria se abrirá uno nuevo.</p>	
<b>Reporte G01 – Informe de Operación de turnos por circulo y por proceso</b>		
10015	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>José Luis se comunica e informa que no ha podido generar el reporte porque la página se queda en blanco y finalmente se cae.</p> <p>Se realizan pruebas desde la mesa de ayuda y se detecta que efectivamente el reporte no se genera.</p> <p>Se hacen pruebas por varios nodos, y con varias fechas. Por favor verificar. Bucaramanga.</p> <p><b>DESARROLLO</b></p> <p>El código fuente del reporte G-01 que entrego el operador ADA no tiene concordancia por favor solicitar a los usuarios de la SNR la descripción de lo que debe mostrar este reporte - Informe de Operación de turnos por circulo y por proceso; esta descripción la deben realizar a través de un formato de requerimientos.</p>	
<b>RUPTA 01 – Informe Estadístico RUPTA</b>		
10609	<p><b>REQUERIMIENTO</b></p> <p>Informa María Ortiz que al generar el reporte RUPTA01: Informe estadístico RUPTA no le esta arrojando ninguna información, por ejemplo, al consultar el turno 2009-372-6-481 vemos que tiene Acto: REGISTRO UNICO DE PREDIOS Y TERRITORIOS ABANDONADOS-RUPTA, pero al generar el reporte con los siguientes parámetros: Fecha Desanotación: 10/02/2009, Círculo Registral: Buenaventura, no aparece ningún dato. Favor verificar. Relacionado con el caso 10482. Buenaventura.</p>	
11187	<p><b>REQUERIMIENTO</b></p> <p>Informa María Ortiz que al generar el reporte RUPTA01: Informe estadístico RUPTA no le esta arrojando ninguna información, por ejemplo, al consultar los turnos 2009-372-6-2248 y 2009-372-6-2249 vemos que tiene Acto: REGISTRO UNICO DE PREDIOS Y TERRITORIOS ABANDONADOS-RUPTA, pero al generar el reporte con los siguientes parámetros: Fecha Desanotación: 19/08/2008, Círculo Registral: Buenaventura, no aparece ningún dato. Por favor tener en cuenta el caso 10609. Buenaventura.</p>	

	<p><b>DESARROLLO</b></p> <p>El reporte RUPTA01 no muestra un detalle de los documentos asociados al proceso RUPTA - REGISTRO UNICO DE PREDIOS Y TERRITORIOS ABANDONADOS su objetivo es desplegar un total teniendo en cuenta el tipo (Devueltos, Inscritos y Radicados); analizando el reporte se detecto que la validación que se estaba teniendo al consultar la fecha no era correcta; Se modifico la condición quedando de la siguiente forma:</p> <pre> /*REPORTE-RUPTA.INFORME ESTADISTICO RUPTA - TIPO DEVUELTO*/ ANTES WHERE TO_CHAR(sir_op_nota.nota_fecha_creacion, 'mmyyyy') = \$P{P_FECHA}  AHORA WHERE TO_CHAR(sir_op_nota.nota_fecha_creacion, 'YYYY-MM-DD') = \$P{P_FECHA}  /*REPORTE-RUPTA.INFORME ESTADISTICO RUPTA - TIPO INSCRITOS*/ ANTES and TO_CHAR(ntcn_fecha_creacion, 'mmyyyy') = \$P{P_FECHA}  AHORA AND TO_CHAR(ntcn_fecha_creacion, 'YYYY-MM-DD') = \$P{P_FECHA}  /*REPORTE-RUPTA.INFORME ESTADISTICO RUPTA - TIPO RADICADOS*/ ANTES where TO_CHAR(slct_fecha_solicitud, 'mmyyyy') = \$P{P_FECHA}  AHORA WHERE TO_CHAR(slct_fecha_solicitud, 'YYYY-MM-DD') = \$P{P_FECHA} </pre> <p>Se adjunta reporte; verificar si el usuario lo necesita para enviarlo; y se les aclara que el cambio será reflejado en el próximo despliegue en producción. El archivo adjunto es reporte- RUPTA01_Buenaventura.pdf</p>
10850	<p><b>REQUERIMIENTO</b></p> <p>En la pantalla de Reporte RUPTA01: Informe estadístico RUPTA aparece en el titulo: "Reporte 04 diario desanotador de documentos del día". Favor corregir el titulo ya que no describe el Reporte, gracias. Mesa de Ayuda</p> <p><b>DESARROLLO</b></p> <p>Se modificaron las siguientes líneas del archivo \WEB\WEB-INF\jspseguro\administracion\ADMINREPORTES_REPORTRUPTA01_PARAMFORM.jsp</p> <pre> &lt;!-- AHERREN --&gt; &lt;!-- Se actualizo esta línea ya que cuando se hacia el llamado del reporte RUPTA01 se desplegaba en el encabezado Reporte 04 --&gt; &lt;!--td background="<!---%-request.getContextPath()%&gt;/jsp/images/tabla_central_bgn001.gif" class="titulotbcentral"&gt;Reporte 04&lt;/td--&gt; &lt;td background="<%-request.getContextPath()%&gt;/jsp/images/tabla_central_bgn001.gif" </pre--> </pre>

	<p>class="titulotbcentral"&gt;RUPTA01&lt;/td&gt;</p> <p>&lt;!-- AHERRENO --&gt; &lt;!-- Se actualizo esta línea ya que cuando se hacia el llamado del reporte RUPTA01 se desplegaba en el encabezado Diario desanotador de documentos del día --&gt; &lt;!--td class="bgnsup"&gt;Diario desanotador de documentos del día&lt;/td--&gt; &lt;td class="bgnsup"&gt;Informe estadístico RUPTA01&lt;/td&gt;</p>	
<b>Reporte 25 – Informe Recaudo del día (TIRACAJA)</b>		
9787	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>En el reporte 25 del día 17/06/2009 no está apareciendo el recibo número 40446024 por el valor de 20,500 con el turno 2009-004-6- 1278, solo aparece el del mayor valor que equivale al recibo número 0446057 por 28,000, por lo tanto no coincide el valor físico pues faltan los 20.500 pesos, se adjunta imagen del reporte. Andes.</p> <p><b>DESARROLLO</b></p> <p>Se modifico el script del reporte. Se adjunta reporte; verificar si el usuario lo necesita para enviarlo; y se les aclara que el cambio será reflejado en el próximo despliegue en producción. El archivo adjunto es reporte_25_Andes_17_06_2009_ANGELA_VERGARA.pdf</p>	
10539	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Informa Paola Salazar que al generar el reporte 25 con los siguientes parámetros: Fecha de Pago: 13/07/09, Círculo Registral: Pereira, Usuario: Lucero Arenas Zapata, aparecen en el campo Proceso (Proc.) los turnos de mayor valor: 2009-290-6-10332, 2009-290-6-12136, 2009-290-6-1141 como Registro (Reg) y no como mayor valor (Mav). Favor verificar ya que todos los turnos de mayor valor deben aparecer como Mav en el campo Proceso. Pereira.</p> <p><b>DESARROLLO</b></p> <p>En el reporte 25 - Informe Recaudo del Día (Tiracaja) el sistema realiza una validación que tiene como prioridad validar el proceso del turno en este caso el proceso (6); por tal motivo si se desea tener como prioridad el pago de Mayor Valor la SNR deberá diligenciar el formato de Requerimientos en la cual se realice la descripción del cambio.</p>	
9096	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Reportes 51, 25, 6, 8, 9,19 y 20. Problemas con los valores de los recaudos que refleja el Sistema ya que las 3 cantidades son inconsistentes. Cúcuta.</p> <p><b>DESARROLLO</b></p> <p>Se modifico el script en los reportes 25 y 51 por que estaban sumando los turnos anulados. Se adjuntan reportes; verificar si el usuario los necesita para enviarlos; también los pueden generar por la aplicación. Los archivos adjuntos son</p> <p>Reporte_25_Cucuta_Alexandra_Espinoza_15_04_2009.pdf  Reporte_25_Cucuta_Astrid_Villar_15_04_2009.pdf  Reporte_25_Cucuta_Gloria_Suarez_15_04_2009.pdf  Reporte_25_Cucuta_Gloria_Torres_15_04_2009.pdf  Reporte_25_Cucuta_Shirley_Acevedo_15_04_2009.pdf  Reporte_25_Cucuta_Yulieth_Beltran_15_04_2009.pdf</p>	

	Reporte_51_Cucuta_15_04_2009.pdf	
10405	<p><b>REQUERIMIENTO</b></p> <p>José Luis García envía mail:</p> <p>En el día de ayer 09/07/2009 los valores aplicados de los tiracajas cuando existen ingresos por concepto de certificados de otras ORIPS no coinciden con los valores aplicados del reporte 51. Al parecer el sistema los suma en los valores de esta oficina más los ingresados en otras ORIPS. Este caso ocurrió con los usuarios HUGO.PRADA MAXIMO.FERNANDEZ Y LADYS.PALACIO.</p> <p>El Reporte de estos usuarios es inconsistente, pues el tiracaja saca el total de las radicaciones teniendo en cuenta los certificados de otras orips, y luego al Total Vr oficina le vuelve a sumar los certificados de otras ORIPS. Adicional al Reporte del usuario Maximo Fernandez, no le descuenta el turno anulado, debe tener 6, y le suma 7. Bucaramanga.</p> <p><b>DESARROLLO</b></p> <p>Se modifico el script que consulta los pagos realizados a otras ORIPS con esta modificación no sumara los turnos anulados de otras ORIPS. A demás se modifico el campo Total Oficina Vr para que no tuviera en cuenta el valor de otras ORIPS. Se adjuntan reportes; verificar si el usuario los necesita para enviarlos. Los archivos adjuntos son:</p> <p>Reporte_25_HUGO_PRADA.pdf  Reporte_25_MAXIMO_FERNANDEZ.pdf  Reporte_25_LADYS_PALACIO.pdf</p>	
8695	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Patricia se comunica con la Mesa de Ayuda e informa que cuando radican un turno de devoluciones se saltan los recibos 2 números. Verificar el turno 2009-062-4-4, se salto el recibo 39536318 y 39536319. Se realizaran las verificaciones y pruebas. Patricia también indica que el turno le afectó el reporte 51, pues no refleja 40 sino 80. Por favor revisar el caso, dado que se crearon dos turnos 2009-062-4- 4 y 2009-062-4-5 de devolución el día 6 de mayo, por el cajero Darío Yépez, y los dos turnos suman \$120252. Pero en el Reporte 51, muestra un valor de devoluciones por \$200252. De manera que se tiene inconsistencia. Carmen de Bolívar.</p> <p><b>DESARROLLO</b></p> <p>El error se presentaba por datos de la solicitud 2009-062-4-4 del turno 2009-062-4-4 registraba 3 pagos de 40.000. Por favor volver a generar el reporte.</p>	
10223	<p><b>REQUERIMIENTO</b></p> <p>Lady Clavijo se comunica y menciona que no se puede generar el reporte 25 de los usuarios jose.caviedes y francy.quintero, pues la pantalla se queda en blanco. Por favor revisar que pasa con el reporte, pues se realizan pruebas desde la mesa de ayuda, por varios nodos, y no sale el reporte para dichos usuarios del día 07/07/2009. Aguachica.</p>	
10224	<p><b>REQUERIMIENTO</b></p>	

	<p>Nos informa la funcionaria Paola Salazar que al tratar de generar el reporte 25 de la funcionaria Violeta Rojas del día 7-07-09 el sistema no lo genera y queda la pantalla en blanco. Se confirma con la funcionaria si esto pasa con el resto de los funcionarios pero nos informa que es únicamente con la funcionaria Violeta Rojas, por lo tanto se trató de generar el reporte pero no sale y queda la pantalla en blanco. Se anexa la imagen del reporte al momento de generarlo. Pereira.</p> <p><b>REQUERIMIENTO</b></p> <p>Al tratar de generar el reporte 25 del día 07 de Julio de 2009 del usuario JHON JAIRO ROJAS TORO la pantalla se queda en blanco, se intenta generar aquí en mesa de ayuda por todos los nodos pero presenta el mismo inconveniente. Manizales.</p> <p><b>DESARROLLO</b></p> <p>El reporte estaba generando una excepción el cual no permitía su despliegue; este caso ya fue solucionado. Se le informa al usuario que puede ingresar al sistema en cualquier nodo y generar el reporte.</p>	
<b>Reporte 51 – Resumen Diario de Ingresos</b>		
9904	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Se comunica Ligia Estella Suárez e informa una inconsistencia en el reporte 51 de la fecha de 18/06/09 y del 19/06/09, debido a que en el total de resumen general por bancos el valor no coincide con el campo TOTAL en el reporte esto se debe a que en el resumen general por bancos no esta tomando los turnos anulados y los suma. Piedecuesta.</p> <p><b>DESARROLLO</b></p> <p>Se modifico el script para que no sumara los valores de turnos anulados. Se adjunta reporte; verificar si el usuario lo necesita para enviarlo; y se les aclara que el cambio será reflejado en el próximo despliegue en producción. Los archivos adjuntos son Reporte_51_Piedecuesta_18_06_2009.pdf y Reporte_51_Piedecuesta_19_06_2009.pdf</p>	
10643	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Ligia Estella se comunica indicando que tiene una inconsistencia con el Reporte 51 del día 14/07/09. El reporte muestra una diferencia de \$6850, cuando debería ser \$7150. El usuario indica que ese día solo los cajeros Yolly y Edinson tienen movimiento, y efectivamente los valores totales de cada uno coinciden el tiracaja de cada uno es igual a lo referenciado en el Rep 51. Pero en el tiracaja de Yolly hay una diferencia de \$7150, y en el Rep 51 la diferencia es de \$6850. Por favor verificar. Piedecuesta.</p> <p><b>DESARROLLO</b></p> <p>El reporte 51 estaba generando una diferencia de -6.850 por que en el turno 2009-314-6-5333 tenia como valor documento pago (0) el cual se actualizo teniendo en cuenta el valor de consignación (14.000). Por favor indicarle al usuario de Piedecuesta que genere de nuevo el reporte 51 del día 17/07/2009.</p>	

10153	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Informa Mery Tamayo que al generar el reporte 51: Ingresos y egresos, del día 03/07/09 salio con valores negativos en la columna de nuevas entradas y el total de derechos e impuesto. Favor verificar este reporte ya que en las columnas de Derechos y de Impuesto esta restando los valores negativos. Villavicencio.</p> <p><b>DESARROLLO</b></p> <p>El turno (2009-230-6-11949 ) es la nueva entrada para el turno (2009-230-6-7732) y en su liquidación el valor de la cuantía es de 156.000.000 diferente a la cuantía del turno de la nueva entrada que fue de 72.000.000 la cual pudo ser modificada en la pantalla de Liquidación al momento de traer la información radicada para el turno anterior.</p> <p>Teniendo en cuenta la diferencia de valores en la liquidación esta influye en la información que se muestra en el reporte 51 en las columnas CUANTIA, DERECHOS, IMPUESTO + SANCION Y LIQUIDACION.</p>	
9940	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Informa Eusebio Sánchez que al generar el Reporte 51: Resumen diario de ingresos y egresos no aparece el total general, ver anexo. Carmen de Bolívar</p> <p><b>MESA DE AYUDA</b></p> <p>Nos comunicamos con Eusebio y nos confirma que en la oficina reciben Efectivo y no Consignaciones, se le informa que el campo Total General al que se refiere es para Resumen General por Bancos, el usuario informa que en reportes anteriores salían datos en ese campo, se le pidió enviar un reporte 51 anterior donde podamos ver que aparezcan valores en ese campo. Nos comunicamos de nuevo con Eusebio informa que verifico y estaba confundiendo los reportes y que el reporte 51 del día 25/06/09 se generó satisfactoriamente, el caso se puede cerrar ya que no presenta inconvenientes.</p>	
10893	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Informa Ligia Stella Suárez que en el Reporte 51 generado el día 23/07/09 aparece en Resumen General por Banco en Valor cheque por \$300.000 y que no debería sumarse al total ya que es de una devolución de dinero. Favor verificar. Piedecuesta.</p> <p><b>DESARROLLO</b></p> <p>Teniendo en cuenta que el reporte 51 - Resumen Diario de Ingresos y Egresos, tiene como objetivo en la sección de RESUMEN GENERAL POR BANCOS desplegar los valores de pagos que se han realizado por concepto de consignaciones y cheques en el cual no se valida el tipo de proceso que tiene cada recaudo y por eso es que despliega el valor de 300.000 en cheque. Se solicita que el usuario de la ORIP se dirija a la SNR para que exponga este caso y así mismo se haga el respectivo análisis del reporte 51 en la sección RESUMEN GENERAL POR BANCOS.</p>	

10278	<p><b>REQUERIMIENTO</b></p> <p>Informa Mery Tamayo que al generar el reporte 51: Resumen diario de ingresos y egresos con los siguientes parámetros: Fecha de pago: 01/07/2009, Círculo: Villavicencio, en el usuario Alondra.Rojas en Documentos Radicados, al sumar Derechos + Impuesto da un valor de 20.878.030 y en liquidación aparece 29.348.300 y hay una diferencia de 11.840. Favor verificar este reporte ya que la usuaria quiere saber a que corresponde ese valor de 11.840 pesos. Villavicencio.</p> <p><b>DESARROLLO</b></p> <p>Se modifico la configuración del campo EFECTIVO. Se adjunta reporte; verificar si el usuario lo necesita para enviarlo; y se les aclara que el cambio será reflejado en el próximo despliegue en producción. El archivo adjunto es Reporte_51_Villavicencio.pdf</p>	
11152	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Se comunica con la mesa de ayuda Paola Salazar y nos comenta que cuando sacan el reporte 51 y el tira caja los turnos anulados no aparecen relacionados el mismo día sino al día siguiente. Los turnos anulados del día de hoy no salen en el reporte 51 del mismo día, se ve reflejado al día siguiente. Y al día siguiente cuando se imprimen los tira caja relaciona otros turnos ya anulados. Pereira.</p> <p><b>DESARROLLO</b></p> <p>Por favor comunicarle al usuario de la ORIP que reporto el caso que el Reporte 25 - Informe de Recaudos del día por Usuario; Realiza la búsqueda de los turnos anulados por su fecha de anulación.</p> <p>Turnos:</p> <ul style="list-style-type: none"> <li>*2009-290-1-63088 <ul style="list-style-type: none"> <li>-Fecha Pago = 31/07/2009</li> <li>-Fecha Anulación = 03/08/2009</li> </ul> </li>   <li>*2009-290-6-13964 <ul style="list-style-type: none"> <li>-Fecha Pago = 30/07/2009</li> <li>-Fecha Anulación = 31/07/2009</li> </ul> </li>   <li>*2009-290-6-14028 <ul style="list-style-type: none"> <li>-Fecha Pago = 30/07/2009</li> <li>-Fecha Anulación = 31/07/2009</li> </ul> </li>   <li>*2009-290-6-13586 <ul style="list-style-type: none"> <li>-Fecha Pago = 27/07/2009</li> <li>-Fecha Anulación = 28/07/2009</li> </ul> </li> </ul>	
11178	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Jose Luis García se comunica indicando que el reporte 51 del 22/07/09 tiene inconveniente con el usuario hugo mantilla, ya que se habilitó el turno 2009-300-1-149872 que había sido anulado, y ahora el reporte está cobrando \$11840 adicionales. El usuario envía reporte generado antes de habilitar el turno 2009-300-1-149872, y el valor total de hugo.mantilla era \$10754940 y ahora sale \$10778620. Es</p>	

	<p>decir se incremento en \$23680 y no en \$11840 como debía ser. Por favor verificar el inconveniente. Bucaramanga.</p> <p><b>DESARROLLO</b></p> <p>El reporte 25 y 51 manejan los Documentos y Certificados Asociados de la misma forma lo cual se puede observar en el total que genera cada uno de ellos: \$10,778,620. El Reporte 25 - Informe de recaudos del día por usuario; presenta las siguientes diferencias:</p> <p>1. Con el turno 2009-300-6-31500 anteriormente el valor del documento era de 14.000 (No tenia en cuenta el valor del Certificado Asociado 11.840) y la diferencia era de 0; y como se puede observar no se desplegaba el turno 2009-300-1-149872</p> <p>Reporte Anterior</p> <p>Radicación 2009-300-6-31500 Recibo 40525905 Efectivo \$ 14,000 Consignación \$ 0 Cheque \$ 0 Banco - No Vr.Docu.Pago \$ 14,000 Diferencia \$ 0 Vr. Liquidado \$ 14,000 Hora Pago 4.06.19 PM Proc. REG</p> <p>2. Con el turno 2009-300-6-31500 actualmente el valor del documento es de 25.840 (Tiene en cuenta el valor del Certificado Asociado 11.840) y la diferencia es de 11.840; y como se puede observar ahora despliega el turno 2009-300-1-149872</p> <p>Reporte Actual</p> <p>Radicación 2009-300-6-31500 Recibo 40525905 Efectivo \$ 14,000 Consignación \$ 0 Cheque \$ 0 Banco - No Vr.Docu.Pago \$ 25,840 Diferencia \$ 11,840 Vr. Liquidado \$ 14,000 Hora Pago 4.06.19 PM</p>	
--	---	--

	<p>Proc. REG</p> <p>3. En el reporte actual se suma el valor del certificado asociado</p> <p>Radicación 2009-300-1-149872      Recibo 40525906      Efectivo \$ 11,840      Consignación \$ 0      Cheque \$ 0      Banco - No      Vr.Docu.Pago \$ 0      Diferencia -\$ 11,840      Vr. Liquidado \$ 11,840      Hora Pago 4.06.19 PM CER-ASC      Proc. CER-ASC</p> <p>El certificado asociado NO se suma dos veces eso se puede observar en el campo Vr.Docu.Pago \$ 0</p>	
<b>Reporte 57 – Diario Radicador de Matriculas</b>		
9697	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Nur se comunica y menciona que el reporte 57, de los días 10 y 11 sale en blanco. Se verifica el reporte con el usuario generándolo desde la Mesa de Ayuda, y sale en blanco, para los días 10 y 11 de junio. Se realizan pruebas de otros días y se generan con normalidad, al revisar otras ciudades están correctas también. Cerete.</p> <p><b>DESARROLLO</b></p> <p>El reporte 57 - Diario Radicador de Matriculas se genera en blanco en los días 10/06/2009 y 11/06/2009 ya que en este día no se radicaron matriculas (nuevas) en el circulo de Cerete; si el usuario desea ver los certificados que se radicaron para la oficina indicarle que puede ver esta información en el Reporte 06 - Diario Radicador de Certificados del día.</p>	
<b>Reporte 74 – Informe de Ingresos por Certificados</b>		
10656	<p><b>REQUERIMIENTO</b></p> <p>Mery Tamayo se comunica y reporta que el Reporte 74 del día 15 de Julio no muestra los totales. Se revisa y en Reporte 74 del día 15, no aparece el valor enfrente de Derechos certificados locales, ni total de certificados, por favor verificar, pues al generar reportes de otras fechas si se mostraban estos campos. Villavicencio.</p> <p><b>DESARROLLO</b></p> <p>Se amplio el tamaño y se configuro la propiedad Estirar con desbordamiento de los campos Derechos certificados locales y Total certificados. Se adjunta reporte; verificar si el usuario lo necesita para enviarlo; y se les aclara que el cambio será reflejado en el próximo despliegue en producción. El archivo adjunto es Reporte_74_15_07_2009_villavicencio.pdf</p>	
<b>Reporte 99 – Ingresos Diarios por Documentos</b>		
9611	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p>	

	<p>Javier se comunica y menciona que el reporte 99 del día 02-06-09 , tiene inconsistencia debido a que los turnos 2009-140-6-5763, 2009- 140-6-5764 presentan valor, cuando fueron nuevas entradas y el reporte debería referenciarlos como \$0. Por favor verificar el reporte 99, pues al generar el reporte 25 se evidencia que los turnos salieron con valor cero. Se adjuntan reportes. Montería.</p> <p><b>DESARROLLO</b></p> <p>El turno 2009-140-6-5764 estaba desplegando valores en Cuantia/Cant. por que no tenia relación con el turno anterior 2005- 140-6-10878 ya que este fue cargado a través de migración lo mismo sucedió con el turno 2009-140-6-5763 no tenia relación con el turno anterior 2005-140-6-10875 se agrego la relación y en el reporte 99 adjunto se puede observar el cambio. Para el turno 2009-140-6-5763 en el campo recaudo se despliega un valor de \$40 ya que en el turno anterior 2005-140-6-1087 se liquido con un valor de 1.014.500. El archivo adjunto es Reporte_99_Monteria.pdf.</p>
<b>Reporte 156 – Diario anulados del Día</b>	
11186	<p><b>REQUERIMIENTO</b></p> <p>Señores mesa de ayuda sir snr</p> <p>El reporte de anulados de el día 03/08/2009 esta relacionando turnos de el año 2008 los cuales no fueron anulados en el día de ayer. Por favor revisar que sucede con este reporte...agradezco su pronta colaboración. Se anexo vía fax dicho reporte.</p> <p>Se genera el reporte y se revisa, y efectivamente referencia turnos del 2008, que no fueron anulados el día de ayer.. 2008-300-1-1, 2008- 300-1-2 y 2008-300-1-3.... 2008-300-6-1; 2008-300-6-2; 2008-300-6-3 Por favor verificar. Bucaramanga.</p> <p>Se modifco script del reporte 156 - Diario anulados del Día; ya que no estaba validando el proceso de la radicación. Se adjunta reporte; verificar si el usuario lo necesita para enviarlo; y se les aclara que el cambio será reflejado en el próximo despliegue en producción. El archivo adjunto es: Reporte 156 Bucaramanga 03_08_2009.pdf</p> <p><b>DESARROLLO</b></p> <pre>/*Esta validación se agrego ya que cuando se consulta en la tabla SIR_OP_TURNO_HISTORIA la búsqueda se debe realizar por 4 campos: ID_TURNO, ID_CIRCULO, ANIO Y ID_PROCESO*/ ***** AND sir_op_turno_historia.id_proceso = sir_op_turno.id_proceso *****</pre>
11223	<p><b>REQUERIMIENTO</b></p> <p>Al generar el reporte 156 DIARIO ANULADOS DEL DIA muestra el turno 2007-200-1-76890 que fue anulado el 21/08/2007. Al generar el reporte 156 DIARIO ANULADOS DEL DIA muestra el turno 2007-</p>

	<p>200-1-76890 que fue anulado el 21/08/2007, al verificar en el sistema se ve que el turno aparece con varias fases de anulados en diferentes fechas tal como se muestra en la imagen anexa y una de las fechas son 16/06/2009 07:48:22 AM y 17/07/2009 08:45:14 AM por esta razón el turno aparece en el reporte 156 generado de estas fechas, es necesario revisar el porque quedó con todas estas fases de anulado al parecer en el momento de reasignarlo se produjo esto. Neiva.</p> <p><b>DESARROLLO</b></p> <p>Se modifico el script del reporte 156 - DIARIO ANULADOS DEL DIA. Se adjunta reporte; verificar si el usuario lo necesita para enviarlo; y se les aclara que el cambio será reflejado en el próximo despliegue en producción. El archivo adjunto es Reporte_156_Neiva_16_06_2009.pdf</p> <pre>***** /* Se agrega esta condición ya que la aplicación al momento de reasignar turnos a diferentes usuarios se están reasignando los turnos anulados y este historial esta siendo registrado en la tabla histórica del turno. Para este se tuvo en cuenta el caso 11223 de GLPI. El reporte siempre tomara la primera fecha de anulación*/ AND sir_op_turno_historia.TRHS_FECHA_REGISTRO IN (SELECT MIN(h.TRHS_FECHA_REGISTRO) FROM sir_op_turno_historia h WHERE h.id_circulo = sir_op_turno_historia.id_circulo AND h.id_proceso = sir_op_turno_historia.id_proceso AND h.TRHS_ID_FASE = 'ANULADO' AND h.id_turno = sir_op_turno_historia.id_turno ) *****</pre>
<b>APLICACION</b>	
<b>Acta de Reparto Notarial</b>	
9159	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>En la oficina de Bucaramanga imprimen un acta de Reparto Notarial y en las observaciones no aparece el nombre del funcionario en cargado de Reparto.</p> <p>De la oficina envían fax de los Repartos Notariales del día 19 de Mayo, y en este en donde se encuentra el error. También nos envían un Reparto del día 7 de mayo de 2009 y este es le que se encuentra bien. Se anexan los documentos. Bucaramanga.</p> <p><b>MESA DE AYUDA</b></p> <p>Nos comunicamos con Jose Luis Garcia para preguntarle si este error se ha vuelto a presentar, él nos comenta que no que solo se presento esa vez y no mas, el siempre esta pendiente por que él esta encargado de este proceso. Este caso solo se presentó con este círculo, ninguna otra oficina presentó este problema. Quedamos pendientes si se presenta de nuevo este caso para revisarlo y hacer su seguimiento.</p> <p><b>DESARROLLO</b></p> <p>Teniendo en cuenta la observación de Nayibe Baron (Mesa de Ayuda) se cierra este caso.</p>
<b>Prefabricados</b>	
9589	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Nos informan de la oficina de registro que algunos certificados están</p>

	<p>saliendo con la información incompleta. Nos informan de la oficina de registro que algunos certificados están saliendo con la información incompleta, por lo tanto se indica a la funcionaria Ivonne Ospina indicarnos los turnos y folios correspondientes a los que no les está saliendo la información completa y enviar los certificados que salieron incompletos, el folio es el siguiente 230-123165 tiene 2 anotaciones y solo salió impresa en el certificado la 1.</p> <p>Por favor tener en cuenta el caso 8870. Villavicencio.</p>	
10162	<p><b>REQUERIMIENTO</b></p> <p>Informa Mery Tamayo que algunos de los certificados de libertad continúan saliendo con las anotaciones incompletas:</p> <ul style="list-style-type: none"> <li>-El turno 2009-230-1-55873, matricula 118627, falto la anotación 8, se envía la reimpresión y sale bien.</li> <li>-Los turnos 2009-230-1-56295 y 2009-230-1-57077, matricula 111309, falto las anotaciones 18 y 19, se envía la reimpresión y sale bien.</li> <li>-El turno 2009-230-3-719, matricula 230-17180, en la anotación 7 se corrige anotación a cancelar y en la anotación 9 se agrega el nombre,</li> <li>-Se expide el certificado con turno 2009-230-1-54641 y salio sin las correcciones. Favor verificar por que no salen bien las impresiones y por reimpresión si. Relacionado con el caso 9595. Villavicencio.</li> </ul> <p><b>DESARROLLO</b></p> <p>Análisis: Se verificó que al realizar la impresión de los certificados salen incompletos en los círculos de Pruebas, La Ceja, Buenaventura y Villavicencio, ya que, existe una validación en la creación de este certificado que verifica cuales son los círculos que podrán generarse a partir de la tabla de SIR_OP_PREFABRICADO, que contiene un imprimible ya prefabricado, pero este imprimible no se actualiza cuando hay cambios en las anotaciones de un folio, por esto el imprimible contiene información desactualizada. Actualmente los círculos que generan estos certificados a partir de prefabricados son los arriba mencionado, ya que están parametrizados en el archivo gov.sir.forseti.properties en mapas.jar en el campo gov.sir.forseti.prfb.circulo=999,017,372,230 donde están los códigos de los círculos ya mencionados.</p> <p>Solución: Se modifco la propiedad así: gov.sir.forseti.prfb.circulo = 999 del archivo gov.sir.forseti.propertiesel retirando los círculos de La Ceja, Buenaventura y Villavicencio para que no impriman los certificados de desde la tabla SIR_OP_PREFABRICADO y haga el proceso como el resto de círculos en Colombia. Este cambio actualmente esta realizado en ambiente de desarollo.</p> <p><b>OBSERVACION ETB:</b></p> <p>Este requerimiento se soluciono con el despliegue (2.1.3) 15/07/2009</p>	
<b>Certificados Incompletos</b>		
9923	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>El usuario Miguel Galvis reporta que desde el momento en que se</p>	

	<p>implementó el sello de fondo de la SNR en los imprimibles de certificados, estos certificados aparecen incompletos en algunas ocasiones. Por favor revisar este tema para poder encontrar el problema y ejecutar su corrección si así corresponde. Pasto</p> <p><b>DESARROLLO</b></p> <p>Se revisó el caso y no se encontró ningún certificado incompleto, por favor enviar caso puntual para dar solución a este inconveniente.</p> <p><b>MESA DE AYUDA</b></p> <p>Se hizo revisión por parte de desarrollo para evidenciar el problema reportado por el administrador de centro de cómputo Miguel Galvis, se espera algún soporte o evidencia para poder investigar más a fondo el problema y poder tomar los correctivos necesarios si los hay. Se da cierre a este caso.</p>	
10248	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>El usuario Miguel Galvis reporta que desde el momento en que se implementó el sello de fondo de la SNR en los imprimibles de certificados, estos certificados aparecen incompletos en algunas ocasiones. Por favor revisar este tema ya que se había reportado con el caso 9923. En esta ocasión el usuario indica que un turno en el cual se imprimió el certificado incompleto es 2009-240-1-53322, por favor revisar el ID Imprimible de este turno y revisar el problema que reporta el usuario. Pasto.</p> <p><b>DESARROLLO</b></p> <p>Análisis: este caso ya se había reportado con el caso 9923 y también se le había dado respuesta y era que no había información suficiente para solucionar el caso.</p> <p>Al igual que este caso, se dice que el certificado esta incompleto, pero que le falta? no se que le falta ya que reimprimo el certificado y lo observo bien, con todas sus anotaciones.</p> <p>Adjunto envío el imprimible generado por el usuario.</p> <p><b>MESA DE AYUDA</b></p> <p>Se hace revisión del caso y efectivamente el certificado se imprime completo, se pedirá al usuario soportes físicos de la impresión del certificado para seguir en la búsqueda del problema y poder tomar los correctivos correspondientes. Se envía mensaje a la oficina de registro de Pasto para solicitar los soportes correspondientes donde muestre los certificados incompletos, el caso se cerrara, si la oficina envía los soportes del turno relacionado en este caso se reabrirá este mismo para su revisión de lo contrario se abrirá una nueva incidencia para su revisión.</p>	
<b>Fases Invertidas</b>		
9996	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>En el sistema con frecuencia se presentan fases invertidas en los diferentes turnos de todos los procesos, el sistema ejecuta</p>	

	<p>correctamente el flujo de fases para los turnos pero algunas veces el sistema invierte las fases de los turnos tomando la fecha y hora como referencia, esto hace que el flujo del turno no sea coherente y en ocasiones conflicto en el avance de los turnos.</p> <p>Se requiere escalar este caso a desarrollo para que sea analizado el problema ya que el sistema en algunos turnos invierte las fases ordenándolas por fecha y hora de ejecución. Como ejemplo, el turno 2009-017-1-8464 que tiene las fases invertidas. Este problema se presenta con frecuencia en todos los círculos registrales de SIR. La ceja.</p> <p><b>DESARROLLO</b></p> <p>Análisis: Todos los procesos de SIR poseen un turno, este turno esta dividido en fases, estas fases son registradas en la tabla sir_op_turno_historia guarda el histórico por fases del turno; la aplicación en estos momentos guarda en esta tabla también la fecha cuando se ha registrado la fase y lo hace haciendo new Date(), tomando la fecha y hora del servidor; El problema es que existen 8 servidores en producción y no están sincronizados en tiempo y cada fase puede estar responsabilizada a diferentes personas; estas personas pueden estar ingresando a la aplicación en cualquiera de los servidores, al estar desincronizados los servidores pueden ingresar fechas menores de una fase a otra, ocurriendo el problema de fases invertidas.</p> <p>Solución: La solución es sincronizar el reloj de los servidores de aplicación.</p> <p><b>OBSERVACIONES ETB:</b></p> <p>Todos los servidores de producción y Base de Datos que se encuentran en Bogotá y Cali se sincronizan cada 5 minutos con un servidor central en Bogotá con el servidor 10 que es el de aplicación y en Cali con el servidor 9 de aplicación estos a su vez se sincronizan cada hora con el servidor de la hora legal colombiana de la Superintendencia de industria y comercio que se identifica como horalegal.sic.gov.co.</p>	
<b>Recibos Certificados Masivos</b>		
9836	<p><b>REQUERIMIENTO (INFORMATIVO)</b></p> <p>Informa Mónica Robledo que al imprimir certificados masivos no se están imprimiendo todos los recibos de los turnos individuales, por ejemplo el turno 2009-290-20-3455: solo se imprimo el recibo del turno 2009-290-1-50479 y no del turnos 2009-290-1-50480, la usuaria solicito el id de impresión porque por reimpresión tampoco salio, favor verificar ya que la usuaria esta informando que esto ocurre para todos los certificados masivos. Relacionado con el caso 9829. Pereira.</p> <p><b>DESARROLLO</b></p> <p>Análisis: Se trató de replicar este error y no fue posible.</p>	

	<p>Base de Datos Id de recibo = 16947050 y si se necesita el Id del certificado = 16947092 para el turno 2009-290-1-50480. Adjunto se envío en formato PDF los imprimibles solicitados.</p>	
<b>Error al Segregar</b>		
10594	<p><b>REQUERIMIENTO (INFORMATIVO)</b> Luz Perla se comunica informando que intenta hacer una segregación en el folio 314-18956, del turno 2009-314-6-5337 y el sistema le muestra un error de ORA-01722. Por favor revisar el inconveniente. Piedecuesta.</p> <p><b>DESARROLLO</b> Análisis: El problema se encontró en la última publicación a producción (15/07/2009), al modificarse la clase java JDOGenieFolioDAO.java, este archivo es persistente con objetos de la base de datos (el procedimiento SP_SG y el paquete PCK_SIR_VALIDACIONES.SP_DELEGAR_BLOQUEO_FOLIOS) esto llevo a que al subir nuevamente la aplicación en los servidores ocurriera un problema de en la persistencia a estos objetos causando los problemas antes mencionados por los usuarios de la aplicación SIR. Es de anotar que el objeto SP_SG (procedimiento de DB) fue el único que tuvo este problema.</p> <p>Solución: Este problema se solucionó (16/07/2009) solo con reiniciar nuevamente los servidores de aplicación para que hiciera la persistencia correcta sobre los objetos de base de datos.</p>	
<b>Orden de Impresión Certificados</b>		
8734	<p><b>REQUERIMIENTO (INFORMATIVO)</b> Cuando se expiden certificados asociados, al imprimir los recibos correspondientes por cada turno de certificados individuales, invierte el orden de impresión de los recibos y no existe coincidencia del número del recibo preimpreso con el consecutivo asignado por el sistema. Como ejemplo, cito: Turno 2009-028-1-1304 Recibo físico 39530765 Recibo Consecutivo Sistema 39530770. Sonson.</p> <p><b>DESARROLLO</b> Este error no se replicó ni en desarrollo (usuarios.usuarios.pruebas y HUGO.MANTILLA), ni en producción (usuario.usuarios.pruebas); aun así se revisó el flujo de la aplicación y no se encontró en el código fuente algo que cause este problema.</p> <p><b>OBSERVACION ETB:</b> Este requerimiento se revisó con las pruebas del despliegue (2.1.0) 08/07/2009.</p>	
<b>Formulario Calificación</b>		
10846	<p><b>REQUERIMIENTO</b> Se comunica Amparo Pastor de la oficina de registro para informar que no sale el formulario de calificación del turno 2009-340-6-5214. Por favor adjuntar PDF del formulario de calificación para el turno 2009-340-6-5214 ya que en la oficina no salió impreso y el turno lo</p>	

	<p>avanzaron a la fase mesa de control. Sincelejo.</p> <p><b>DESARROLLO</b></p> <p>Análisis: El error se presenta cuando se intenta realizar un ordenamiento de los ids de anotaciones canceladoras, se intentaba convertir un valor null en numérico, por esto generaba una excepción no controlada por la aplicación (FormatNumberException) y por esto no se le presentaba ninguna clase de mensaje al usuario de la no impresión de este formulario.</p> <p>Solución: Se valida antes de la conversión que el valor no sea null.</p> <p>En la clase ImprimibleBaseFolio.java se modifco el método public static void ordenarIdsCancelacion(String a[ ])</p> <pre>//Se valida que el valor no sea null. if(a[k]!= null){     numerosArray [k] = Integer.valueOf(a[k]).intValue(); } }  Arrays.sort(numerosArray);  //Se redefine el array en caso de existir valores nulos dentro a[]. a = new String[numerosArray.length];</pre>	
<b>REQUERIMIENTOS RESUELTOS</b>		
	<p><b>002 Historial de Turnos y Fases</b></p> <p><b>REQUERIMIENTO</b></p> <p>Historial de turnos, fases.</p> <p>SIR-316.</p> <p>En algunos casos el sistema deja las fases de los folios invertidas en el historial, este problema no se presenta frecuentemente y las causas por las cuales se invierten las fases es desconocida. Se necesita que el sistema siempre mantenga el orden de las fases de acuerdo al orden cronológico del flujo del turno.</p> <p><b>SEGUIMIENTO ADA:</b></p> <p>Se requiere la definición del escenario donde se produjo el error, indicando los siguiente: Fecha (tiempo en el que se genero) Proceso (Proceso del sistema donde se evidencio el error) Turno (turno que presenta la inconsistencia)</p> <p><b>OBSERVACIONES SNR:</b></p> <p>Las fases se deben visualizar en estricto orden cronológico Como lo indica el requerimiento y el historial de los casos colocados en GLPI es frecuente el reporte de casos de bloqueos de los turnos los cuales al verificar la ruta del documento se encuentran con las fases invertidas, se debe efectuar por parte del operador del SIR un diagnostico de los casos reportados y encontrar la solución para este tipo de casos.</p> <p><b>DESARROLLO</b></p> <p><b>Análisis de Requerimiento:</b></p> <p>Todos los procesos de SIR poseen un turno, este turno esta dividido en fases, estas fases son registradas en la tabla sir_op_turno_historia guarda el histórico por fases del turno; la aplicación en estos momentos guarda en esta tabla también la fecha cuando se ha registrado la</p>	

fase y lo hace haciendo new Date(), tomando la fecha y hora del servidor, pudiendo guardar horas exactamente iguales para varios registros, al consultar estas fases el sistema las ordena por fecha de creación y por esto puede traer fases invertidas.

#### **Observaciones:**

Se cambian los ordenamientos de estas consultas por id\_turno\_historia.

Clase:

C:\WORK\SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie\JDOTurnosDAO.java

#### 1.) Método:

```
protected TurnoHistoriaEnhanced getMaxTurnoHistoria(TurnoHistoriaPk tID,  
PersistenceManager pm) throws DAOException
```

Código línea 4328:

```
q.setOrdering("idTurnoHistoria descending");
```

#### 2.) Método:

```
protected TurnoHistoriaEnhanced updateNumCopiasLastTurnoHistoriaByFase(TurnoEnhancedPk turnoID, String nombreFase,  
Integer numCopias, PersistenceManager pm) throws DAOException
```

Código línea 7477:

```
query.setOrdering("idTurnoHistoria descending");
```

#### 3.) Método:

```
protected TurnoHistoriaEnhanced updateUsuarioAtiendeLastTurnoHistoriaByFase(TurnoEnhancedPk turnoID, String  
nombreFase, UsuarioEnhanced usuarioAtiende, PersistenceManager pm) throws DAOException
```

Código linea 7508:

```
query.setOrdering("idTurnoHistoria descending");
```

## 011 Calificación Folios Extensos

### **REQUERIMIENTO**

Calificación: Folios extensos.

SIR-260

La calificación de folios extensos se torna muy demorada, debido entre otras razones a que solo el cargue de la información contenida en el folio es sumamente lento

### SEGUIMIENTO ADA:

Especificar el proceso de mejora, documentando como se debe de mejorar, que se debe eliminar, que se debe agregar, etc.

Se sugiere cargar las últimas anotaciones (por ejemplo las últimas 100), con el objeto de mejorar el desempeño en la calificación de estos folios. O modificar la aplicación para la carga de dichos folios basado en demanda.

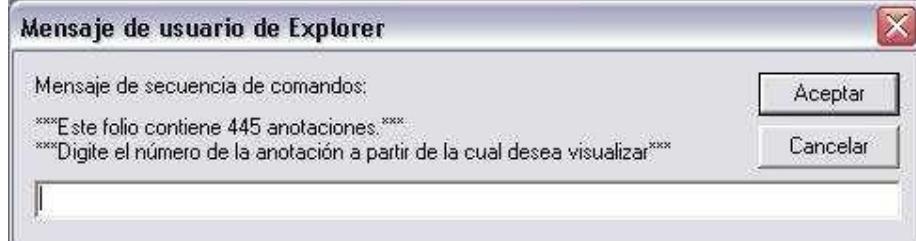
### OBSERVACIONES SNR:

En el aplicativo folio, se muestra en un campo la cantidad de anotaciones que tiene el folio consultado, luego, cuando se van a visualizar las anotaciones y si hay mas de 100 anotaciones, el aplicativo pregunta a partir de que anotación desea visualizar el usuario, se puede tener la misma funcionalidad en SIR. Como la inconsistencia se presenta es al oprimir el botón consultar el folio, una vez se ejecute esta acción el aplicativo debe hacer la pregunta y luego si iniciar el cargue de la información con esa validación es posible que sea más ágil el cargue de la misma.

### **DESARROLLO**

**Análisis:** La aplicación cargaba todas las anotaciones del folio por lo que en algunos casos el proceso se tornaba lento.

**Solución:** Al presionar el botón  se despliega la ventana



en la cual se le muestra al usuario el número de anotaciones que posee el folio y el usuario debe digitar el número de la anotación a partir de la cual desea visualizar. Si el usuario digita un número negativo, un número mayor al número de anotaciones del folio o letras, se visualizaran todas las anotaciones.

#### Cambios en el código:

Se crearon los archivos *anotacionFolioXMLHTTP.js* y *conXMLHTTP.js* en la ruta *web|jsp|plantillas|js*, y el archivo *AnotacionesFolioXMLHTTPServlet.java* en la ruta *gov|sir|core|web*. Estos archivos son para realizar la conexión XMLHTTP para traer el número de anotaciones del folio sin recargar la página.

Se modificaron los siguientes archivos:

Nombre: *gov|sir|core|negocio|acciones|comun|ANPaginadorAnotaciones.java*

Cambio:

Se agregó “*import gov.sir.core.negocio.modelo.AnotacionesFolio;*”

En el método “*private EventoRespuesta consultarAnotacionesFolio(EvnPaginadorAnotaciones evento)*” se comentó la línea “*List anotacionesTemporales = null;*” porque la variable nunca es utilizada.

En el método “*private EventoRespuesta consultarAnotacionesDefinitivasFolio(EvnPaginadorAnotaciones evento)*” se comentó la línea “*List anotacionesTemporales = null;*” porque la variable nunca es utilizada y se cambió la línea:

```
EvnRespPaginadorAnotaciones ev= new EvnRespPaginadorAnotaciones(anotacionesDefinitivas, null, cantidadRegistros,  
EvnRespPaginadorAnotaciones.CONSULTAR_ANOTACIONES_FOLIO, evento.getNombrePaginador(),  
evento.getNombreResultado());
```

por:

```
EvnRespPaginadorAnotaciones ev;  
if(AnotacionesFolio.getNumAnotacionesTotalV()>0){  
    ev= new EvnRespPaginadorAnotaciones(anotacionesDefinitivas, null, AnotacionesFolio.getNumAnotacionesTotalV()-  
    AnotacionesFolio.getNumAnotacionesFolio(), EvnRespPaginadorAnotaciones.CONSULTAR_ANOTACIONES_FOLIO,  
    evento.getNombrePaginador(), evento.getNombreResultado());  
}else{  
    ev= new EvnRespPaginadorAnotaciones(anotacionesDefinitivas, null, cantidadRegistros,  
    EvnRespPaginadorAnotaciones.CONSULTAR_ANOTACIONES_FOLIO, evento.getNombrePaginador(),  
    evento.getNombreResultado());  
}.
```

En el método “*private EventoRespuesta cargarAnotacionesPaginador(EvnPaginadorAnotaciones evento)*” se comentó la línea “*List anotacionesTemporales = null;*” porque la variable nunca es utilizada.

En el método “*private EventoRespuesta cargarAnotacionesDefinitivasPaginador(EvnPaginadorAnotaciones evento)*” se comentó la línea “*List anotacionesTemporales = null;*” porque la variable nunca es utilizada.

	<p>Nombre: <i>gov\sir\core\web\acciones\comun\AWPaginadorAnotaciones.java</i></p> <p>Cambio:</p> <p>Se agregó la línea “<i>import gov.sir.core.negocio.modelo.AnotacionesFolio;</i>”</p> <p>En el método “<i>private Evento consultarAnotacionesFolio(HttpServletRequest request)</i>” se cambió la línea:  <i>EvnPaginadorAnotaciones e=new EvnPaginadorAnotaciones(usuarioAuriga, folioID, new Integer(inicio).intValue(), new Integer(cantidad).intValue(), EvnPaginadorAnotaciones.CONSULTAR_ANOTACIONES_FOLIO, usuarioNeg, consultarAnotacionesDefinitivas);</i></p> <p>por:</p> <pre> EvnPaginadorAnotaciones e; if(AnotacionesFolio.getNumAnotacionesFolio()&gt;0 &amp;&amp; AnotacionesFolio.getNumAnotacionesTotalV()&gt;0){     e = new EvnPaginadorAnotaciones(usuarioAuriga, folioID, AnotacionesFolio.getNumAnotacionesFolio(),     AnotacionesFolio.getNumAnotacionesTotalV(), EvnPaginadorAnotaciones.CONSULTAR_ANOTACIONES_FOLIO,     usuarioNeg, consultarAnotacionesDefinitivas); } else{     e = new EvnPaginadorAnotaciones(usuarioAuriga, folioID, new Integer(inicio).intValue(), new     Integer(cantidad).intValue(), EvnPaginadorAnotaciones.CONSULTAR_ANOTACIONES_FOLIO, usuarioNeg,     consultarAnotacionesDefinitivas); } </pre> <p>En el método “<i>private Evento cambiarPagina(HttpServletRequest request)</i>” se cambió la línea:</p> <pre> if(numPagina!=null &amp;&amp; numPagina!=""){     por:     if(numPagina!=null &amp;&amp; !"".equals(numPagina)){ </pre> <p>En el método “<i>private void filter_PaginadorMultiselect_Enabled(HttpServletRequest request)</i>” se cambió el código:</p> <pre> Boolean checkBoxController_MultiSelectEnabled; checkBoxController_MultiSelectEnabled = null;  if( ( null == ( checkBoxController_MultiSelectEnabled = (Boolean)session.getAttribute( PREFIX + CHECKBOXCONTROLLER_MULTISELECTENABLED ) ) )     //   ( ! Boolean.TRUE.equals( checkBoxController_MultiSelectEnabled ) ) ) {     return; }  } // if por: if(null == session.getAttribute( PREFIX + CHECKBOXCONTROLLER_MULTISELECTENABLED ) ){     return; } </pre>
	<p>Nombre: <i>gov\sir\core\web\acciones\registro\AWCalificacion.java</i></p> <p>Cambio:</p> <p>Se quitó la línea “<i>import org.auriga.core.modelo.transferObjects.Estacion;</i>”.</p> <p>Se agregó la línea “<i>import gov.sir.core.negocio.modelo.AnotacionesFolio;</i>”.</p> <p>En el método “<i>public Evento perform(HttpServletRequest request)</i>” se agregó el código:</p> <pre> String numAnotacionesDesde = request.getParameter(WebKeys.CAMPO_NUM_ANOTACIONES DESDE); if(numAnotacionesDesde==null){     AnotacionesFolio.setNumAnotacionesFolio((new Integer(0).intValue())); } else{     try{         int tNumAnotaciones = Integer.parseInt(request.getParameter(WebKeys.CAMPO_NUM_ANOTACIONES_TOTAL));         if(tNumAnotaciones&gt;0){             AnotacionesFolio.setNumAnotacionesTotalV(tNumAnotaciones);         }     } } </pre>

```

        }else{
            AnotacionesFolio.setNumAnotacionesTotalV(0);
        }
        AnotacionesFolio.setNumAnotacionesFolio(Integer.parseInt(numAnotacionesDesde));
        if(AnotacionesFolio.getNumAnotacionesFolio()<1 ||
        AnotacionesFolio.getNumAnotacionesFolio()>tNumAnotaciones){
            AnotacionesFolio.setNumAnotacionesFolio(0);
        }else{
            AnotacionesFolio.setNumAnotacionesFolio(AnotacionesFolio.getNumAnotacionesFolio()-1);
        }
    }catch(NumberFormatException eNum){
        AnotacionesFolio.setNumAnotacionesFolio(0);
        AnotacionesFolio.setNumAnotacionesTotalV(0);
    }
}

```

En el método “*private Evento doCalificacion\_ReimpresionSellos\_PrintSelected\_Action( HttpServletRequest request )*” se quitó la línea “*String sessionId = session.getId();*”

En el método “*private Evento doCalificacion\_ReimpresionSellos\_Search\_Action( HttpServletRequest request )*” Se eliminaron las líneas “*gov.sir.core.negocio.modelo.Usuario usuarioSIR = (gov.sir.core.negocio.modelo.Usuario) session.getAttribute( WebKeys.USUARIO );*” y “*String sessionId = session.getId();*”

En el método “*private Evento actualizarFolioDirecciones(HttpServletRequest request)*” se cambio el código:

```

for (java.util.Iterator iteratorT1 = t1_folio_direcciones.iterator(); iteratorT1.hasNext(); ) {
    Direccion t1_element = (Direccion) iteratorT1.next();
    String t1_element_id = t1_element.getIdDireccion();
    boolean existsInT0 = false;
    // for (java.util.Iterator iteratorT0 = t0_folio_direcciones.iterator();
    //      iteratorT0.hasNext(); ) {
    // Direccion t0_element = (Direccion) iteratorT0.next();
    // // tener en cuenta null's en id;
    // String t0_element_id = t0_element.getIdDireccion();
    //
    // if ((null != t0_element_id)
    //     && (null != t1_element_id)
    //     && (t0_element_id.equalsIgnoreCase(t1_element_id))) {
    //     existsInT0 = true;
    //     break;
    // } // end if
    // } // end for

    if (!existsInT0) {
        // ver si tiene flag set to delete en true; si lo tiene, no es necesario agregarla
        // otra condicion es que no puede tener el id, porque hasta ahora, no se ha creado

        if (t1_element.isToDelete()) {
            continue;
        }
        if (null != t1_element.getIdDireccion()) {
            continue;
        }

        // se añade a la lista de cambios
        t2_folio_direcciones.add(t1_element);
        // se remueve de la otra lista ?
        // por el momento, no, para no afectar la iteracion; luego se podrian dejar en un arreglo
    } // end if
} // end for
por:
int numDirFolios = t1_folio_direcciones.size();
Direccion t1_element;
for (int itDF = 0; itDF < numDirFolios; itDF++) {
    t1_element = new Direccion();
    t1_element = (Direccion) t1_folio_direcciones.get(itDF);
    // ver si tiene flag set to delete en true; si lo tiene, no es necesario agregarla

```

```

    // otra condicion es que no puede tener el id, porque hasta ahora, no se ha creado
    if (t1_element.isToDelete()) {
        continue;
    }
    if (null != t1_element.getIdDireccion()) {
        continue;
    }
    // se añade a la lista de cambios
    t2_folio_direcciones.add(t1_element);
}

```

Y el código:

```

for (java.util.Iterator iteratorT1 = t1_folio_direcciones.iterator();
     iteratorT1.hasNext(); ) {
    Direccion t1_element = (Direccion) iteratorT1.next();

    // si tiene flag setToDelete = true, se agrega en la lista de modificaciones
    if( null == t1_element ) {
        continue;
    }

    // debe tener id
    if( null == t1_element.getIdDireccion() ) {
        continue;
    }

    if( t1_element.isToDelete() ) {
        t2_folio_direcciones.add( t1_element );
    }
}

```

por:

```

numDirFolios = t1_folio_direcciones.size();
for (int itDF = 0; itDF < numDirFolios; itDF++) {
    t1_element = new Direccion();
    t1_element = (Direccion) t1_folio_direcciones.get(itDF);

    // si tiene flag setToDelete = true, se agrega en la lista de modificaciones
    if( null == t1_element ) {
        continue;
    }

    // debe tener id
    if( null == t1_element.getIdDireccion() ) {
        continue;
    }

    if( t1_element.isToDelete() ) {
        t2_folio_direcciones.add( t1_element );
    }
}

```

En el método “*private Evento aceptarCorreccion(HttpServletRequest request)*” se eliminaron las líneas:

```

String tipo_oficina="";
String numero_oficina="";
tipo_oficina = request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_TIPO);
numero_oficina = request.getParameter(CSolicitudRegistro.ANOTACION_OFICINA_DOCUMENTO_NUM);

```

En el método “*private Evento crearFolioEnglobe(HttpServletRequest request)*” se quitó el código:

```

Enumeration e=tabla.elements();
List escogidas=new ArrayList();
while(e.hasMoreElements()){
    List anotacionesEscogidas=(List)e.nextElement();
    escogidas.addAll(anotaciones);
}

```

En el método “*private Evento avanzar(HttpServletRequest request)*” se eliminaron las líneas:

```

Estacion estacion = (Estacion)session.getAttribute(WebKeys.ESTACION);
String devolver = request.getParameter(DEVOLUCION);

```

En el método “*private Evento devolver(HttpServletRequest request)*” se eliminaron las líneas

```

Estacion estacion = (Estacion)session.getAttribute(WebKeys.ESTACION);
Usuario usuario = (Usuario)session.getAttribute(WebKeys.USUARIO);
String camino = request.getParameter(RESPUESTAWF);

```

En el método “*private Evento inscripcionParcial(HttpServletRequest request)*” se eliminaron las líneas:

```

Estacion estacion = (Estacion)session.getAttribute(WebKeys.ESTACION);
Usuario usuario = (Usuario)session.getAttribute(WebKeys.USUARIO);
String camino = request.getParameter(RESPUESTAWF);

```

En el método “*private Evento cancelar(HttpServletRequest request)*” se cambió el código:

```
Iterator it = RPag.getAnotacionesActual().iterator();
```

```

int i= 0;
while(it.hasNext()){
    Object obj = (Object)it.next();
    list.add(obj);
}

```

por:

```
list.addAll(RPag.getAnotacionesActual());
```

En el método “*private Evento cancelarAnotacion(HttpServletRequest request)*” se eliminó la línea  
“*Iterator itC=anotaciones.iterator();*”

Nombre: *gov\sir\core\web\helpers\comun\MostrarFolioHelper.java*

Cambio:

Se insertó la línea “*import gov.sir.core.negocio.modelo.AnotacionesFolio;*”

En el método “*public void drawGravamenesCautelares(HttpServletRequest requestI, JspWriter out)*” se cambió el código:

```

String textoPostTotalAnotaciones="Anotaci&oacute;n";
if(numeroAnotaciones>1){
    textoPostTotalAnotaciones="Anotaciones";
}
por:
String textoPostTotalAnotaciones="Anotaciones";
if(AnotacionesFolio.getNumAnotacionesTotalV()>0){
    if(AnotacionesFolio.getNumAnotacionesTotalV()==1){
        textoPostTotalAnotaciones='Anotaci&oacute;n';
    }
    out.println("          <td class='''+this.eTitulos+'|' width='43%'>&nbsp;Visualizando&nbsp;'+
(AnotacionesFolio.getNumAnotacionesTotalV())-AnotacionesFolio.getNumAnotacionesFolio()) +&nbsp;de&nbsp;'+
AnotacionesFolio.getNumAnotacionesTotalV() + '&nbsp;' +
}else{
    if(numeroAnotaciones==1){
        textoPostTotalAnotaciones="Anotaci&oacute;n";
    }
    out.println("          <td class='''+this.eTitulos+'|' width='43%'>&nbsp;El folio
tiene&nbsp;'+textoTotalAnotaciones+'&nbsp;'+textoPostTotalAnotaciones+'>      ");
}

```

Nombre: *gov\sir\core\web\helpers\comun\ PaginadorAvanzadoHelper.java*

Cambio:

Se insertó la línea “*import gov.sir.core.negocio.modelo.AnotacionesFolio;*”

En el método “*public void drawGUI(HttpServletRequest request, JspWriter out)*” después de la línea

“*out.write(" function cambiarAnotacionActual() {\r\n");*” se agregó el código:

```

if(AnotacionesFolio.getNumAnotacionesTotalV()>0{
    out.write("    var vTempAnotacion = document."+nombreForm+".ORDEN_ANOTACION_ACTUAL.value - " +
AnotacionesFolio.getNumAnotacionesFolio() +';\r\n');
    out.write("    if(vTempAnotacion<1){\r\n");

```

```

        out.write("    alert('Anotacion fuera del rango.\r\n');");
        out.write("    document."+nombreForm+".ORDEN_ANOTACION_ACTUAL.value='';\r\n");
        out.write("    }else{\r\n");
        out.write("        document."+nombreForm+".ORDEN_ANOTACION_ACTUAL.value = vTempAnotacion;\r\n");
    }
}

```

y después de las líneas:

```

out.write("    document."+nombreForm+".ORDEN_ANOTACION_ACTUAL.value='';\r\n");
out.write("    }\r\n");

```

se agregó:

```

if(AnotacionesFolio.getNumAnotacionesTotalV()>0){
    out.write("    }\r\n");
}

```

Nombre: *gov|sir|core|web|helpers|registro|MatriculaCalifHelper.java*

Cambio:

En el método “*public void drawGUI(HttpServletRequest request, JspWriter out)*” se cambió la línea:

```

out.println("<td></td><td>Cerrar<br><input type='checkbox' name='cbAprobar'\nonclick='AprobarRechazarSeleccionar_todo(this)'></td>");

```

por:

```

out.println("<td></td><td>Cerrar<br><input type='checkbox' name='cbAprobar'\nonclick='AprobarRechazarSeleccionar_todo(this)'><input type='hidden'\n id='"+WebKeys.CAMPO_NUM_ANOTACIONES_DESDE+"'\n name='"+WebKeys.CAMPO_NUM_ANOTACIONES_DESDE+"' value='0'><input type='hidden'\n id='"+WebKeys.CAMPO_NUM_ANOTACIONES_TOTAL+"'\n name='"+WebKeys.CAMPO_NUM_ANOTACIONES_TOTAL+"' value='0'></td>");

```

y la línea:

```

out.println("<td width='150'><a href='javascript:calificar(\""+matricula+"")'><img\nname='btn_consultar_folio.gif' src='"+request.getContextPath() + "/jsp/images/btn_consultar_folio.gif'\nwidth='180' height='21' style='cursor:hand' border='0'></a></td>");

```

por:

```

out.println("<td width='150'><a href='javascript:getNumAnotacionesFolio(\""+matricula+"", document.all." +\nWebKeys.CAMPO_NUM_ANOTACIONES_DESDE + ", document.all." +\nWebKeys.CAMPO_NUM_ANOTACIONES_TOTAL + ")'><img name='btn_consultar_folio.gif' src=''" +\nrequest.getContextPath() + "/jsp/images/btn_consultar_folio.gif' width='180' height='21' style='cursor:hand'\nborder='0'></a></td>");

```

Nombre: *web|WEB-INF|jspSeguro|registro|CALIFICACION\_INSCRIPCION.jsp*

Cambio:

Se eliminaron las líneas:

```

<%@page import="gov.sir.core.web.*"%>
<%@page import="java.util.*" %>

```

Y se agregaron las líneas:

```

<%@page import="org.auriga.core.web.HelperException"%>
<%@page import="gov.sir.core.web.WebKeys"%>
<%@page import="java.util.Map" %>
<%@page import="java.util.Iterator" %>
<%@page import="java.util.List" %>
<%@page import="java.util.ArrayList" %>
<script language="javascript" src="<%request.getContextPath()%>/jsp/plantillas/js/conXMLHTTP.js"></script>
<script language="javascript"
src="<%request.getContextPath()%>/jsp/plantillas/js/anotacionFolioXMLHTTP.js"></script>

```

Nombre: *web|WEB-INF|jspSeguro|registro|VER\_ANOTACION.jsp*

Cambio:

Se eliminó la línea:

```

<%@page import ="gov.sir.core.web.helpers.registro.*,org.auriga.core.web.*" %>

```

y se agregaron las líneas:

```
<%@page import ="gov.sir.core.web.helpers.registro.VerAnotacionHelper"%>
<%@page import ="org.auriga.core.web.HelperException"%>
```

Nombre: `gov|sir|core|web| WebKeys.java`

Cambio:

Se agregaron las líneas:

```
/***
 * Constante que representa el nombre del campo del formulario que contiene el número de anotaciones desde el cual el
usuario quiere visualizar.
 */
public static String CAMPO_NUM_ANOTACIONES_DESDE = "CAMPO_NUM_ANOTACIONES_DESDE";

/***
 * Constante que representa el nombre del campo del formulario que contiene el número de anotaciones total que posee
el folio.
 */
public static String CAMPO_NUM_ANOTACIONES_TOTAL = "CAMPO_NUM_ANOTACIONES_TOTAL";
```

Nombre: `web.xml`

Cambio:

Se agregó:

```
<servlet>
    <servlet-name>AnotacionesFolioXMLHTTPServlet</servlet-name>
    <servlet-class>gov.sir.core.web.AnotacionesFolioXMLHTTPServlet</servlet-class>
</servlet>
<servlet-mapping>
    <servlet-name>AnotacionesFolioXMLHTTPServlet</servlet-name>
    <url-pattern>/AnotacionesFolioXMLHTTPServlet</url-pattern>
</servlet-mapping>
```

## 015 Certificado de Pertenencia Simplificación al ingresar a la fase (INFORMATIVO)

### **REQUERIMIENTO**

Certificado de Pertenencia: Simplificación al ingresar a la fase\_015\_141

SIR-248.

Cuando el usuario ingrese con el rol "oficio de pertenencia" el sistema debe mostrarle de una vez la pantalla con el proceso, fase y turnos que están en trámite en ésta fase.

### SEGUIMIENTO ADA:

Se requiere validar la solicitud con el usuario, debido a la modificación de la forma como son desplegadas las listas de turnos, de manera estándar por el aplicativo.

### OBSERVACIONES SNR:

Validar técnicamente con ETB si el cambio es muy impactante en desarrollo, de ser así eliminar incidente y tenerlo muy presente para el SURIC.

### Describir el Impacto en el Desempeño de la Aplicación

Validar técnicamente con ETB si el cambio es muy impactante en desarrollo, de ser así eliminar incidente y tenerlo muy presente para el SURIC. Sería de alto impacto en el aplicativo haciéndolo mucho mas ágil.

### **DESARROLLO**

#### **Análisis de Requerimiento:**

Este caso de simplificar la entrada a los procesos fue resuelto en el caso mantis 0729 y acta de requerimiento No 017 desarrollado por el ingeniero Osbert Linero; Entregado en el despliegue (2.1.0) 08/07/2009 a la SNR.

## 024 Especificaciones de Anotaciones Canceladoras (INFORMATIVO)

**REQUERIMIENTO**

Especificaciones de anotaciones canceladoras.  
SIR-195.

Igual situación se presenta con la corrección de especificación de anotaciones canceladoras, en las aparentemente se realiza bien el proceso, pero al finalizar el turno y consultar el folio, no se ven los cambios efectuados.

**SEGUIMIENTO ADA:**

Favor tener unos folios ejemplo o turnos donde podamos apreciar el problema.. no se ha podido reproducir en desarrollo

**OBSERVACIONES SNR:**

Favor hacer seguimiento si en la mesa de ayuda se han reportado mas casos de esta inconsistencia para determinar si amerita un ajuste a la funcionalidad en el proceso de correcciones al modificar la especificación de anotaciones canceladoras.

**DESARROLLO****Análisis de Requerimiento:**

Se ha revisado en los casos de GLPI y este caso no se ha reportado; aún así se realizaron pruebas realizando cancelaciones de anotaciones en los procesos de calificación y corrección y tampoco se presento este error. Por tanto se cierra este caso.

**036 Se requiere modificación de Query reporte 15 y 18 Bucaramanga  
(INFORMATIVO)****REQUERIMIENTO**

Modificación de Query reporte 15 y 18 - Bucaramanga. ID GLPI 6781

Por favor revisar los reportes 15 y 18 correspondientes a los proceso de devoluciones de dinero, se están generando vacíos, es decir sin datos. Un ejemplo es el turno 2008-300-4-529 pagado hoy y cuando se manda generar el reporte no aparece...por favor revisar.

**DESARROLLO**

Este caso se reporto al operador anterior y no entregaron los archivos adjuntos al caso teniendo en cuenta que no hay datos para las pruebas se da este caso como cerrado.

**038 Se requiere modificación de querys de reportes 25 y 51  
(INFORMATIVO)****REQUERIMIENTO**

Se requiere modificación de querys de reportes 25 y 51. ID GLPI 6538. Bucaramanga.

El día de hoy 05/02/2009 en la división administrativa revisa todo el movimiento de esos días 27/01/2009 y 29/01/2009, y se dan cuenta que los turnos que tenían el problema aparecen con otros valores y además no aparecen en los tiracajas de los respectivos usuario (maximo.fernandez y hugo.mantilla), los turnos se modificaron, estos turno son : 2009-300-6-3906 que esta por valor de \$14.000 hoy aparece con valor de \$ 13.000 y el turno 2009-300-6-3435 que estaba con valor de \$ 50.000 hoy aparece con valor de \$ 9.100.

**DESARROLLO**

Los turnos no se despliegan en los tiracajas por que la información ha sido modificada

<p>verificar casos 6431 y 6383:</p> <p>1. 2009-300-6-3435  Solicitud -&gt;2009-300-6-931  Fecha Solicitud -&gt; 27/01/2009  Fecha Pago -&gt; 09/01/2009  Fecha Anulación -&gt; 28/01/2009  NOTA: Este turno se modifco por el operador ADA con el ID GLPI 6383</p> <p>2. 2009-300-6-3906  Solicitud -&gt;2009-300-6-945  Fecha Solicitud -&gt; 27/01/2009  Fecha Pago -&gt; 09/01/2009  Fecha Anulación -&gt; 28/01/2009  NOTA: Este turno se modifco por el operador ADA con el ID GLPI 6431.</p>	
<b>039 Se requiere modificación de querys de reporte 25 (INFORMATIVO)</b>	
<p><b>REQUERIMIENTO</b></p> <p>Se requiere modificación de querys de reportes 51 (tiracaja). ID GLPI 6601. Bucaramanga.</p> <p>El usuario José Luis nos informa que se saco el tiracaja del día 19 de enero de 2009 del usuario maximo.fernandez, pero en el no aparece el turno 2009-300-6-1918 el cual se anulo por duplicidad con el turno 1920.</p>	
<b>DESARROLLO</b>	
<p>El turno 2009-300-6-1918 no se despliega en el reporte 25 (Tiracaja) por que no se registro ningún pago en la fecha 19/01/2009 en esta fecha se anulo el turno.</p>	
<b>040 Se requiere modificación de querys de reporte G-03 (INFORMATIVO)</b>	
<p><b>REQUERIMIENTO</b></p> <p>Se requiere modificación de querys de reportes g-03. ID GLPI 6613. Manizales.</p> <p>Al intentar generar el reporte la pantalla se queda en blanco....Se realizan pruebas por todos los nodos. Se revisa con detalle el caso, y se encuentra que el sistema si esta generando el reporte, pero no corresponde al circulo que se solicita, para este caso puntual, se genera el reporte de Manizales, y muestra la información de Bucaramanga.</p>	
<b>DESARROLLO</b>	
<p>El código fuente del reporte G-03 que entrego el operador ADA no tiene concordancia se solicito por correo a los usuarios de la SNR la descripción de lo que debe mostrar este reporte - Informe de Turnos en Registro por Fases; esta descripción la deben realizar a través de un formato de requerimientos. Este correo fue enviado 14/07/2009 a la usuaria Imelda Muñoz al correo <a href="mailto:imelda.munoz@supernotariado.gov.co">imelda.munoz@supernotariado.gov.co</a></p>	
<b>041 Se requiere modificación de querys de reportes 11 y G-03 (INFORMATIVO)</b>	
<p><b>REQUERIMIENTO</b></p> <p>Se requiere modificación de querys de reporte 11. ID GLPI 6668. Pasto. Al tratar de generar el reporte g-03 se generan solo títulos nada mas; el reporte de estadísticas con el rango mencionado tampoco lo genera es el reporte 11. Miguel Galvis.</p>	

## **DESARROLLO**

El reporte 11- Estadísticas por actos de registro se genero satisfactoriamente

El código fuente del reporte G-03 que entrego el operador ADA no tiene concordancia; se solicito por correo a los usuarios de la SNR la descripción de lo que debe mostrar este reporte - Informe de Turnos en Registro por Fases; esta descripción la deben realizar a través de un formato de requerimientos.

## **042 Se requiere modificación de querys de reporte 57 (INFORMATIVO)**

### **REQUERIMIENTO**

Se requiere modificación de querys de reporte 57. ID GLPI 6770. La Mesa.

El usuario Gladys Varón nos envía por fax el reporte 57 del día 10-02-2009 el cual se imprimió el día 10 de febrero de 2009, donde salen 6 matriculas, pero al imprimir el mismo reporte el día 16 de febrero de 2009 el reporte esta sacando matriculas adicionales sin turno, por favor verificar.

## **DESARROLLO**

Cuando se realizo el proceso de radicación de las matriculas 166-82930, 166-82931, 166-82932, 166-82933 y 166-82934 no se registro en la tabla SIR\_REL\_REG\_FOLIO por eso se desplegaba el campo de turno en blanco. Se actualizaron los datos los cuales se pueden ver reflejados en la generación del reporte.

Con el siguiente script se actualizaron los registros.

Ing. Guillermo Cabrera  
INSERT INTO SIR.SIR\_REL\_REG\_FOLIO VALUES ('525', 6, '166', '2009',  
'166-82930', '10-02-2009', '10-02-2009', 525, 2009);  
INSERT INTO SIR.SIR\_REL\_REG\_FOLIO VALUES ('525', 6, '166', '2009',  
'166-82931', '10-02-2009', '10-02-2009', 525, 2009);  
INSERT INTO SIR.SIR\_REL\_REG\_FOLIO VALUES ('525', 6, '166', '2009',  
'166-82932', '10-02-2009', '10-02-2009', 525, 2009);  
INSERT INTO SIR.SIR\_REL\_REG\_FOLIO VALUES ('525', 6, '166', '2009',  
'166-82933', '10-02-2009', '10-02-2009', 525, 2009);  
INSERT INTO SIR.SIR\_REL\_REG\_FOLIO VALUES ('525', 6, '166', '2009',  
'166-82934', '10-02-2009', '10-02-2009', 525, 2009);

## **046 Creación Rol CREACION\_MASIVA\_AS**

### **REQUERIMIENTO**

Crear un rol en el aplicativo SIR que permita: 1- crear folios (pantalla administrativa (PA) *creación directa de folio*. 2 *consulta índices de propietarios*. 3. *ver detalles de folio*. Este requerimiento se ha hecho con los oficios: 2009 – EE2408 y 2009 –EE 5988 del 05 de febrero/09 y marzo 11 de 2009 respectivamente. Así mismo crear un reporte que permite visualizar el listado de folios creados por los usuarios que tengan el nuevo rol.

Para dar continuidad al desarrollo del contrato por medio del cual se pretende ingresar al nuevo sistema (SIR o FOLIO MAGNETICO), la información de predios que se encuentran en libros físicos denominados de “Antiguo Sistema”, es necesario asignar en el SIR a los contratistas que realizarán esta labor, un menú que contenga únicamente las pantallas administrativas mencionadas en el objeto del presente formato.

Este ROL debe ser asignado a los usuarios que fueron contratados para tal fin, de tal manera que sea completamente restringido el acceso a otras funcionalidades, el Rol se

denominará: CREACION\_MASIVA\_AS.

Con el reporte solicitado, se pretender brindar una herramienta para que la ORIP pueda realizar seguimiento al trabajo realizado por los contratistas, básicamente debe reportar la misma información del reporte “*diario radicador de matriculas*”, con la diferencia que debe pedir como parámetros rango de fechas a consultar.

## **DESARROLLO**

### **Análisis de Requerimiento:**

1. Lo que se solicita ya existe en SIR y está en dos pantallas administrativas llamadas CREACION DIRECTA DE FOLIO y CONSULTA DE INDICES DE PROPIETARIOS.

En estas pantallas solo sería necesario modificar dos cosas:

- El acceso solo lo debe Tener como lo dice el ROL CREACION\_MASIVA\_AS
- En la Pantalla de CREACION DIRECTA DE FOLIO es necesario que se coloque como campo requerido DOCUMENTO ORIGEN tal como se ve en la siguiente imagen:

The screenshot shows the 'ADMINISTRACIÓN - CREAR FOLIO' form. At the top, there's a note: 'Tenga en cuenta las siguientes recomendaciones:' with two bullet points. The first point is about using the 'Cargar' button to upload a previously saved file. The second point is about the system generating the matrícula number after entering the folio details. The 'Carga de Folio' section has a red circle around the 'Cargar' button. The 'Apertura' section includes fields for 'Fecha Apertura' (set to 01/06/2009) and 'Número de Radicación'. The 'Estado folio' field is a dropdown menu with a red circle around it. Below these are sections for 'Ubicación' (Círculo: LA PLATA) and 'Datos Documento' (with a red circle around the 'Tipo' dropdown). The status bar at the bottom shows the date as 'Lunes 01/06/2009' and the time as '11:31 a.m.'

La pantalla se debe ver similar a esta

The screenshot shows a web page titled "SNR :: Superintendencia de Notariado y Registro :: Ministerio del Interior y de Justicia :: Rep - Windows Internet Explorer". The URL is <http://192.168.202.2:7778/SNR/seguiradado>. The page features the Colombian flag and the date "01/6/2008". On the left, there's a sidebar with a "ROLES" section containing "Selección de un Rol" buttons for "Aceptar" and "Administración". The main content area has a banner for "Condiciones Generales de Uso | Política de Privacidad" and a "Roles" section. A note at the bottom says: "Espere 5 segundos a que se muestre el diálogo de advertencia del Aplicativo de impresión. Haga click en Siempre, o haga click en el link de iniciar Aplicativo de impresión." A button labeled "Iniciar Aplicativo de Impresión" is present. The taskbar at the bottom shows various open applications.

Y las pantallas administrativas deben estar por la opción de "Administración" mostrando únicamente los grupos así.

The screenshot shows a web page titled "SNR :: Superintendencia de Notariado y Registro :: Ministerio del Interior y de Justicia :: Rep - Windows Internet Explorer". The URL is <http://192.168.202.2:7778/SNR/seguiradado>. The page displays a navigation menu on the left with sections like "MANEJO DE TURNO", "IMPRESION", "USUARIOS", "UBIC GEOGRAFICAS Y DIRECCIONES", "TARIFAS PAGOS Y RECIBOS", "FOLIOS ANOTACIONES Y ACTOS", "REPARTO NOTARIAL", "CIRCULOS Y OFICINAS", "PROCESO REGISTRO DE DOCUMENTOS", "PROCESO DE FOTOCOPIAS", "VALIDACIONES Y RESTRICCIONES", "OTROS", "REPORTES", "PROCESO DE CONSULTAS", "RECURSOS Y REVOCATORIAS DIRECTAS", and "MIGRACION INCREMENTAL". The right side of the screen is mostly blank, labeled "cuerpo". The taskbar at the bottom shows various open applications.

De la imagen de ejemplo NO DEBE IR Reporte de Gestión SQL y en reportes debe ir únicamente el Reporte solicitado que es igual al Reporte 57 pero que en su estructura no trae el campo Turno, debido a que estos folios no se originan por turnos.

### Consideraciones Adicionales

Es necesario que en BD se tengan enlazadas las tablas de Auditoría respectivas para

tener trazabilidad de la información generada por este ROL y su respectivo usuario.

Análisis: Se verificó la viabilidad de diseñar este rol y es posible cumpliendo con todas las especificaciones propuestas por la SNR.

Solución: creo el rol por medio de los siguientes scripts:

```
INSERT INTO sir_op_rol(id_rol, roll_nombre, roll_descripcion) VALUES ('SIR_ROL_CREACION_MASIVA_AS','ROL CREACION MASIVA') INTO SISTEMA,'SIR_ROL_CREACION_MASIVA_AS');

INSERT INTO SIR_OP_ROL_PANTALLA(id_rol,id_pantalla_administrativa)VALUES('SIR_ROL_CREACION_MASIVA_AS',61) INTO
INSERT INTO SIR_OP_ROL_PANTALLA(id_rol,id_pantalla_administrativa)VALUES('SIR_ROL_CREACION_MASIVA_AS',1077)
```

y se creo en desarrollo el usuario DESARROLLO.PRUEBAS para realizar las pruebas pertinentes.

## REPORTE

Se creo el Reporte Folios Masivos el cual consulta por un rango de fechas y circulo los folios que se han creado. Para esto se modificaron los siguientes archivos:

1. En la ruta SIR-APP\web\WEB-INF\classes se modificaron los archivos:
  - a. org.auriga.smart.vistas.xml (Se agregaron las líneas)

```
<!-- AHERRENO 19/08/2009
report-Folios-Masivos -->

<vista>
    <autenticacion>true</autenticacion>
    <excepcion>gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector</excepcion>
    <excepcion>gov.sir.core.web.acciones.excepciones.ValidacionParametrosException</excepcion>
    <nOMBRE>admin.reportes.report-FOLIOS-MAS.param-form.view</nOMBRE>
    <parametro><llave>error</llave><valor>/WEB-INF/jspSeguro/comun/ERROR_MULTIPLE.jsp</valor></parametro>
    <parametro><llave>head</llave><valor>/WEB-INF/jspSeguro/comun/HEADER_LOGOUT.jsp</valor></parametro>
    <parametro><llave>body</llave><valor>/WEB-INF/jspSeguro/administracion/ADMINREPORTES_REPORT_FOLIOS_MAS_PARAMFORM.jsp</valor></parametro>
    <parametro><llave>bottom</llave><valor>/WEB-INF/jspSeguro/comun/FOOTER_DEFAULT.jsp</valor></parametro>
</vista>
```

- b. org.auriga.smart.web.xml (Se agregaron las líneas)

```
<!-- AHERRENO 19/08/2009
report-Folios-Masivos -->
<navegacion>
    <resultado>REPORTE_FOLIOS_MAS_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK</resultado>
    <url-redirect>admin.reportes.report-FOLIOS-MAS.param-form.view</url-redirect>
</navegacion>
```

2. En la ruta SIR-APP\web\WEB-INF\jspseguro\administracion se agrego el jsp:
  - a. ADMINREPORTES\_REPORT\_FOLIOS\_MAS\_PARAMFORM.jsp

- b. Old.REPORT\_LIST (Se agregaron las líneas)

```
<%-- + + + + + + + + --%>
<%-- AHERRENO 19/08/2009 --%>
<%-- report_FOLIOS_MAS --%>
<tr>
    <td> </td>
    <td> onMouseOver="this.style.backgroundColor="#F7F9FD"
onMouseOut="this.style.backgroundColor="">admin.reportes.report-FOLIOS-MAS.param-form.view </td>
```

```

<td>
<%
legendHelper.setMessage( "Despliega el conjunto de matriculas que se han creado masivamente en un determinado
rango de fechas" );
legendHelper.setRegionId( "report_FOLIOS_MAS" );
legendHelper.setRegionName( ": Folios Masivos" );

try {
    legendHelper.render( out );
}
catch( java.io.IOException e ){
    out.write( e.toString() );
}

%>
</td>

<td> </td>
<td onMouseOver="this.style.backgroundColor="#F7F9FD"" onMouseOut="this.style.backgroundColor=""> </td>
</tr>

```

3. En la ruta SIR-APP\src\gov\sir\core\web\acciones\administracion  
a. AwReportes.java (Se agregaron las líneas)

```

// AHERRENO 19/08/2009
// Reporte Folios Masivos
public static final String REPORTE_FOLIOS_MAS_SEND_ACTION = "REPORTE_FOLIOS_MAS_SEND_ACTION";
public static final String REPORTE_FOLIOS_MAS_BACK_ACTION = "REPORTE_FOLIOS_MAS_BACK_ACTION";

// AHERRENO 19/08/2009
// Reporte Folios Masivos
public static final String REPORTE_FOLIOS_MAS_LOADCIRCULOSBYUSUARIO = =
"REPORTE_FOLIOS_MAS_LOADCIRCULOSBYUSUARIO";

// -----
// AHERRENO 19/08/2009
// Declaración de las Variables para el reporte Folios Masivos
public static final String REPORTE_FOLIOS_MAS_PARAM_CMDKEY = "reporte-FOLIOS_MAS.report";
public static final String REPORTE_FOLIOS_MAS_NOMBRE = "";
public static final String REPORTE_FOLIOS_MAS_PARAM_PFECHAINI = "P_APERTURAFO利O_FECHA_INI";
public static final String REPORTE_FOLIOS_MAS_PARAM_PFECHAFIN = "P_APERTURAFO利O_FECHA_FIN";
public static final String REPORTE_FOLIOS_MAS_PARAM_PPAGINA = "P_NUM_PAGINA";
public static final String REPORTE_FOLIOS_MAS_PARAM_PCIRCULONOMBRE = "P_CIRCULO_NOMBRE";

//AHERRENO 19/08/2009
//Reporte Folios Masivos
if (REPORTE_FOLIOS_MAS_LOADCIRCULOSBYUSUARIO.equals(accion)) {
    return do_ReporteXX_LoadCirculosByUsuario_Action(request);
}

// AHERRENO 19/08/2009
// Reporte Folios Masivos
if (REPORTE_FOLIOS_MAS_SEND_ACTION.equals(accion)) {
    do_Reporte_FOLIOS_MAS_Send_Action(request);
}
if (REPORTE_FOLIOS_MAS_BACK_ACTION.equals(accion)) {
    String[] itemIds = new String[]{ REPORTE_FOLIOS_MAS_PARAM_PFECHAINI, REPORTE_FOLIOS_MAS_PARAM_PFECHAFIN,
    REPORTE_FOLIOS_MAS_PARAM_PPAGINA, REPORTE_FOLIOS_MAS_PARAM_PCIRCULONOMBRE };
}

// -----
// AHERRENO 19/08/2009
// Método encargado de validar las Variables del reporte Folios Masivos
private void do_Reporte_FOLIOS_MAS_Send_Action(HttpServletRequest request)
throws ValidacionParametrosException {

    ValidacionParametrosException exception =
        new AdminReportes_InvalidParametersExceptionCollector();

    String param_01 = request.getParameter(REPORTE_FOLIOS_MAS_PARAM_PFECHAINI);
}

```

```

String param_02 = request.getParameter(REPORTE_FOLIOS_MAS_PARAM_PFECHAFIN);
String param_03 = request.getParameter(REPORTE_FOLIOS_MAS_PARAM_PPAGINA);
String param_04 = request.getParameter(REPORTE_FOLIOS_MAS_PARAM_PCIRCULONOMBRE);

if ((null == param_01) || ("".equalsIgnoreCase(param_01))) {
    exception.addError("parametro: Fecha Inicio: Debe digitar un valor");
} else if ((null == param_02) || ("".equalsIgnoreCase(param_02))) {
    exception.addError("parametro: Fecha Final: Debe digitar un valor");
} else {
    String send1 = null;
    String send2 = null;
    Date fecha1 = null;
    Date fecha2 = null;
    try {
        fecha1 = DateFormatUtil.parse(param_01);
        send1 = DateFormatUtil.format("yyyy-MM-dd", fecha1);
        param_01 = send1;
    } catch (Exception e) {
        exception.addError("parametro: Fecha Inicio : La fecha es inválida" + e.getMessage());
    }
    try {
        fecha2 = DateFormatUtil.parse(param_02);
        send2 = DateFormatUtil.format("yyyy-MM-dd", fecha2);
        param_02 = send2;
    } catch (Exception e) {
        exception.addError("parametro: Fecha Final : La fecha es inválida" + e.getMessage());
    }
}

if ((null == param_03) || ("".equalsIgnoreCase(param_03))) {
    exception.addError("parametro: No. de Página: Debe digitar un valor");
} else {
    for (int i = 0; i < param_03.length(); i++) {
        if (!Character.isDigit(param_03.charAt(i))) {
            exception.addError("parametro: No. de Página: Debe ser un valor numerico");
            break;
        }
    }
}

if ((null == param_04) || ("".equalsIgnoreCase(param_04))) {
    exception.addError("parametro: Circulo: Debe digitar un valor");
} else {
    if (WebKeys.SIN_SELECCIONAR.equals(param_04)) {
        exception.addError("parametro: Circulo: Debe digitar un valor");
    }
}

String[] itemIds = new String[]{
    REPORTE_FOLIOS_MAS_PARAM_PFECHAINI,           REPORTE_FOLIOS_MAS_PARAM_PFECHAFIN,
    REPORTE_FOLIOS_MAS_PARAM_PPAGINA, REPORTE_FOLIOS_MAS_PARAM_PCIRCULONOMBRE
};

do_ReporteXX_SaveState(itemIds, request);

// raise -----
if (exception.getErrores().size() > 0) {
    throw exception;
}
//-----

String[] params = new String[]{
    REPORTE_FOLIOS_MAS_PARAM_CMDKEY,
    REPORTE_FOLIOS_MAS_PARAM_PFECHAFIN,           REPORTE_FOLIOS_MAS_PARAM_PFECHAINI,
    REPORTE_FOLIOS_MAS_PARAM_PCIRCULONOMBRE,       REPORTE_FOLIOS_MAS_PARAM_PPAGINA,
};

String[] values = new String[]{
    REPORTE_FOLIOS_MAS_PARAM_CMDKEY, param_01, param_02, param_03, param_04
};

String url = makeUrl(params, values);
reportsServices_ConfigureParameterUri(request, url);

do_ReporteXX_RemoveState(itemIds, request);

```

<pre>         do_ReporteXX_RemoveState_LoadCirculosByUsuario_Action(request);     }  4. En la ruta SIR-APP\src\gov\sir\core\web\navegacion\administracion a. controlNavegacionReportes.java (Se agregaron líneas)  //AHERRENO 19/08/2009 //Reporte Folios Masivos public static final String REPORTE_FOLIOS_MAS_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK =     "REPORTE_FOLIOS_MAS_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK";  //AHERRENO 19/08/2009 //Reporte Folios Masivos listaReportesJasper.add(AWReportes.REPORTE_FOLIOS_MAS_SEND_ACTION);  //AHERRENO 19/08/2009 // navegacion sobre pagina if( AWReportes.REPORTE_FOLIOS_MAS_LOADCIRCULOSBYUSUARIO.equals( accion ) ){     return REPORTE_FOLIOS_MAS_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK; }  //AHERRENO 19/08/2009 //Reporte Folios Masivos if( AWReportes.REPORTE_FOLIOS_MAS_BACK_ACTION.equals( accion ) ) {     return REPORTE_XX_BACK_EVENTRESP_OK; }  5. En la ruta SIR-APP\src\gov\sir\hermod\dao\impl\jdogenie a. JDORreportesJasperDAO (Se agregaron líneas)  //AHERRENO 19/08/2009 //Reporte FOLIOS MASIVOS //Se obtienen los parámetros para enviar al procedimiento listaParametros = new ArrayList(); listaParametros.add(AWReportes.REPORTE_FOLIOS_MAS_PARAM_PFECHAINI); listaParametros.add(AWReportes.REPORTE_FOLIOS_MAS_PARAM_PFECHAFIN); listaParametros.add(AWReportes.REPORTE_FOLIOS_MAS_PARAM_PCIRCULONOMBRE); mapaReportes.put(AWReportes.REPORTE_FOLIOS_MAS_PARAM_CMDKEY, new Object[] {     "SELECT seq_reporte_57.NEXTVAL FROM dual",     "sp_reporte_FOLIOS_MAS(?, ?, ?, ?)", listaParametros,     "DELETE FROM sir_op_reporte_57 WHERE sec_proceso = " });  6. En la ruta SIR-APP\src a. org.auriga.smart.vistas.xml  &lt;!-- AHERRENO 19/08/2009 report-FOLIOS MASIVOS --&gt;  &lt;vista&gt;     &lt;autenticacion&gt;true&lt;/autenticacion&gt;     &lt;excepcion&gt;gov.sir.core.web.acciones.excepciones.AdminReportes_InvalidParametersExceptionCollector&lt;/excepcion&gt;     &lt;excepcion&gt;gov.sir.core.web.acciones.excepciones.ValidacionParametrosException&lt;/excepcion&gt;     &lt;nombre&gt;admin.reportes.report-FOLIOS-MAS.param-form.view&lt;/nombre&gt;     &lt;parametro&gt;&lt;llave&gt;error&lt;/llave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/ERROR_MULTIPLE.jsp&lt;/valor&gt;&lt;/parametro&gt;     &lt;parametro&gt;&lt;llave&gt;head&lt;/llave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/HEADER_LOGOUT.jsp&lt;/valor&gt;&lt;/parametro&gt;     &lt;parametro&gt;&lt;llave&gt;body&lt;/llave&gt;&lt;valor&gt;/WEB- INF/jspSeguro/administracion/ADMINREPORTES_REPORT_FOLIOS_MAS_PARAMFORM.jsp&lt;/valor&gt;&lt;/parametro&gt;     &lt;parametro&gt;&lt;llave&gt;bottom&lt;/llave&gt;&lt;valor&gt;/WEB-INF/jspSeguro/comun/FOOTER_DEFAULT.jsp&lt;/valor&gt;&lt;/parametro&gt; &lt;/vista&gt;  b. org.auriga.smart.web.xml  &lt;!-- AHERRENO 19/08/2009 report-FOLIOS MASIVOS --&gt; &lt;navegacion&gt;     &lt;resultado&gt;REPORTE_FOLIOS_MAS_LOADCIRCULOSBYUSUARIO_EVENTRESP_OK&lt;/resultado&gt;     &lt;url-redireccion&gt;admin.reportes.report-FOLIOS-MAS.param-form.view&lt;/url-redireccion&gt; </pre>	
---	--

</navegacion>

## **060 Inconvenientes al Calificar Turnos (INFORMATIVO)**

### **REQUERIMIENTO**

Inconvenientes al calificar turnos. Cuando se calificaron los turnos 2009-070-6-7146 y 2009-070-6-7147, el calificador envío el turno 2009-070-6-7147, a mayor valor y luego le dio deshacer cambios para eliminar dicho proceso, pero el sistema no elimina los cambios que se hicieron con el turno, sino que elimina todos los temporales que tiene el folio. Por lo tanto se perdieron datos que se habían hecho en el turno 2009-070-6-7146 al deshacer cambios en el turno 2009-070-6-7147.

Se verifica el inconveniente reportado, al revisar el folio 070-55388 en SIR se encuentran 3 anotaciones, se imprime el formulario de calificación del turno 2009-070-6-7146 y efectivamente aparece la anotación 4, por lo tanto si se eliminó el dato, cuando el usuario dio deshacer cambios en el turno 2009-070-6-7147.

El ingeniero Edgar Soto soluciona el caso puntual, pero se debe revisar el inconveniente por Desarrollo pues el botón deshacer cambios debe eliminar los datos realizados por el turno únicamente, no todo lo que tenga el folio. Caso de Referencia Id Glpi 10040.

### **DESARROLLO**

#### **Análisis de Requerimiento:**

Este problema radica en el borrado de datos temporales al deshacer cambios de un folio, no tiene en cuenta el turno que envió la solicitud de borrado y por esto borra todos los cambios sobre el folio.

#### **Observaciones:**

Para resolver esto se creo un nuevo método de borrado de estos temporales que tiene en cuenta solo los cambios generado por el turno que envió la solicitud el deshacer cambios.

#### **1.) /SIR-APP/src/gov/sir/forseti/ForsetiService.java**

```
/*
 * Deshace los cambios que están siendo aplicados a los folios del turno, borra toda la información temporal
 * @param turnoID
 * @param usuario
 * @return
 * @throws ForsetiException
 */
public boolean deshacerCambiosFolioByTurno(TurnoPk turnoID, Usuario usuario, boolean desbloquear) throws
ForsetiException{
try {
    forseti.getFolioDAO().deshacerCambiosFolioByTurno(turnoID, usuario);

    if (desbloquear) {
        forseti.getFolioDAO().desbloquearFolios(turnoID, usuario);
    }
    return true;
}
catch (DAOException e) {
    ForsetiException fe = new ForsetiException("Fallo en el servicio de deshacer los cambios temporales de un turno", e);
    fe.setHashErrores(e.getHashErrores());
    fe.setErrores(e.getErrores());
    Log.getInstance().error(ForsetiService.class, fe.getMessage(), e);
    throw fe;
}
}
```

## 2.) /SIR-APP/src/gov/sir/forseti/interfaz/ForsetiServiceInterface.java

```
/**  
 * Deshace los cambios que están siendo aplicados a los folios del turno, borra toda la información temporal  
 * @param turnoID  
 * @param usuario  
 * @return  
 * @throws DAOException  
 */  
public boolean deshacerCambiosFolioByTurno(TurnoPk turnoID, Usuario usuario, boolean desbloquear) throws Throwable;
```

## 3.) /SIR-APP/src/gov/sir/forseti/ejb/ForsetiServiceEJBBean.java

```
/**  
 *  
 * @ejb:interface-method view-type="both"  
 */  
public boolean deshacerCambiosFolioByTurno(TurnoPk turnoID, Usuario usuario, boolean desbloquear) throws  
ForsetiException {  
    return forseti.deshacerCambiosFolioByTurno(turnoID, usuario, desbloquear);  
}
```

## 4.) /SIR-APP/src/gov/sir/forseti/dao/FolioDAO.java

```
/**  
 * Deshace los cambios que están siendo aplicados a los folios del turno, borra toda la información temporal  
 * @param turnoID  
 * @param usuario  
 * @return  
 * @throws DAOException  
 */  
public boolean deshacerCambiosFolioByTurno(TurnoPk turnoID, Usuario usuario)  
throws DAOException;
```

## 5.) /SIR-APP/src/gov/sir/forseti/dao/impl/jdogenie/JDOGenieFolioDAO.java

linea 14970:

```
/**  
 * Deshace los cambios que están siendo aplicados a los folios del turno, borra toda la información temporal  
 * @param turnoID  
 * @param usuario  
 * @return  
 * @throws DAOException  
 */  
public boolean deshacerCambiosFolioByTurno(TurnoPk turnoID, Usuario usuario)  
throws DAOException {  
  
    PersistenceManager pm = AdministradorPM.getPM();  
    boolean rta = false;  
  
    FolioEnhanced folio;  
    UsuarioEnhanced us;  
  
    try {  
        pm.currentTransaction().begin();  
  
        //Validación de turno  
        TurnoEnhanced turno = this.getTurnoByID(new TurnoEnhancedPk(turnoID), pm);  
  
        if(turno==null){  
            throw new DAOException("El turno no existe");  
        }  
  
        //Validación de usuario  
        UsuarioEnhancedPk uid = new UsuarioEnhancedPk();  
        uid.idUsuario = usuario.getIdUsuario();  
  
        us = this.getUsuarioByID(uid, pm);  
  
        if (us == null) {  
            throw new DAOException("El usuario especificado no existe: " +  
                uid.idUsuario);  
        }  
    } catch (Exception e) {  
        e.printStackTrace();  
        rta = false;  
    } finally {  
        pm.currentTransaction().commit();  
    }  
}
```

```

    }

SolicitudEnhanced sol = turno.getSolicitud();

SolicitudFolioEnhanced solFol;

//Se filtran las solicitudes folio, solo se itera sobre las definitivas
List solicitudesFolio = this.getSolicitudesFolioDefinitivas(sol, pm);

for(Iterator it = solicitudesFolio.iterator(); it.hasNext();){
    solFol = (SolicitudFolioEnhanced)it.next();

    if(solFol.getIdSolicitud().equals(turno.getSolicitud().getIdSolicitud())){
        //Se valida el bloqueo del folio:
        this.validarBloqueoFolio(solFol.getFolio(), usuario, pm);

        //Se llama al servicio que deshace los cambios:
        this.deshacerCambiosFolioByTurno(solFol.getFolio(), us, turno, pm);
    }
}

pm.currentTransaction().commit();
rta = true;
} catch (JDOException e) {
    if (pm.currentTransaction().isActive()) {
        pm.currentTransaction().rollback();
    }
    Log.getInstance().error(JDOGenieFolioDAO.class, e.getMessage());
    throw new DAOException(e.getMessage(), e);
} catch (DAOException e) {
    if (pm.currentTransaction().isActive()) {
        pm.currentTransaction().rollback();
    }
    throw e;
} catch (Exception e) {
    if (pm.currentTransaction().isActive()) {
        pm.currentTransaction().rollback();
    }
    Log.getInstance().error(JDOGenieFolioDAO.class, e.getMessage());
    throw new DAOException(e.getMessage(), e);
} finally{
    pm.close();
}

return rta;
}

```

#### linea 16362:

```

if(turno.getIdWorkflow().equals(anotaTmp.getNumRadicacion())||
    turno.getIdWorkflow().equals(anotaTmp.getIdWorkflow())){
    pm.deletePersistent(anotaTmp);
}

```

#### linea 16374:

```

{
    if(turno.getIdWorkflow().equals(anotaTmp.getNumRadicacion())||
        turno.getIdWorkflow().equals(anotaTmp.getIdWorkflow())){
        pm.deletePersistent(anotaTmp);
    }
}

```

#### Clase:

```

/SIR-APP/src/gov/sir/core/negocio/acciones/registro/ANCalificacion.java
//forseti.deshacerCambiosFolio( local_ElementFolio, local_UsuarioSir);
forseti.deshacerCambiosFolioByTurno(turnoPk, local_UsuarioSir, true);

```

**OBSERVACION ETB:**

Este requerimiento se soluciono con el despliegue (2.1.3) 15/07/2009

**065 Certificados Masivos****REQUERIMIENTO**

Se comunica con la Mesa de Ayuda Jose Arboleda por que desde el día martes 7 de julio a los cajeros cuando esta realizando certificados masivos salen todos los turnos no entregados y el aplicativo se vuelve lento cuando se abre la pantalla como cuando toman un turno para trabajarlos. Se realiza un remoto con la oficina para saber en realidad cual es el inconveniente que se presenta con esta oficina, este mismo inconveniente lo presenta también la oficina de Cartagena. Se anexan los pantallazos del remoto. Se detecta que al entrar a los roles que crean turnos de certificados masivos (cajero certificados masivos, usuario operativo, superdoc) muestra en el listado de turnos, turnos anteriores que ya han sido finalizados, como ejemplo se pueden visualizar los turnos 2007-420-20-496 y 2007-420-20-1526. Se requiere hacer una revisión de estos turnos ya finalizados para que no aparezcan en el listado ya que esta puede ser la causa de la lentitud que provoca en el sistema el ingresar a los turnos de estos roles. Florencia.

**DESARROLLO**

Análisis: El problema radica en que los turnos derivados de turnos de certificados masivos se están entregando en la fase entrega certificados, por esto los certificados masivos nunca son entregados y quedan en el listado de entregables.

Solución: Para esto ya no saldrán los turnos derivados de turnos masivos en los entregables individuales, para que no sean entregados por esta pantalla y solo sean entregados por las entrega de certificados masivos. A demás de esto se debe correr un script para actualizar los turnos masivos ya finalizados y que aun aparecen en pantalla.

Clase:

gov.sir.hermod.dao.impl.jdogenie.JDOTurnosDAO

Método:

```
public List getTurnosEjecucion( Estacion estacion, Rol rol, Usuario usuario, Proceso proceso, Fase fase , Circulo circulo) throws DAOException
```

Líneas agregadas:

```
Iterator iter = col.iterator();
TurnoEnhanced turno=null;
while (iter.hasNext()) {
    turno = (TurnoEnhanced) iter.next();
    pm.makeTransient(turno);
    rta.add(turno);
}
```

el script es:

```
begin
    sp_finalizar_turnos_masivos;
end;
```

**073 Error en la Creación Directa de Folio****REQUERIMIENTO**

En el aplicativo SIR, se presenta un error en la creación directa de folio, ya que, en el momento de realizar el proceso de creación de un folio no se pueden hacer definitivos los datos temporales, además no permite crear un folio por pantallas administrativas

pues el sistema presenta error. Id referencia glpi 10618, 10628.

## **DESARROLLO**

### **Análisis de Requerimiento:**

Al momento de convertir definitivo el folio se validaba que las anotaciones temporales correspondieran al turno, pero en este caso no existe turno, ya que, se realiza desde una pagina administrativa, causando un nullPointerException en esta validación.

### **Observaciones:**

Se valida antes que el turno sea null, esto quiere decir que viene de la pagina administrativa.

codigo afectado:gov\sir\forseti\dao\impl\jdogenie\JDOGenieFolioDAO.java,  
linea: 16362  
if(turno == null ||  
turno.getIdWorkflow().equals(anotaTmp.getNumRadicacion())||  
turno.getIdWorkflow().equals(anotaTmp.getIdWorkflow())){  
pm.deletePersistent(anotaTmp);  
}

## **078 Anotación Temporal Corrección Ciudadanos**

### **REQUERIMIENTO**

Anotación temporal corrección ciudadanos. Se detecta problema en SIR cuando se realiza el proceso de corrección de ciudadano en un turno de correcciones, cuando se procede a tomar la misma matrícula para realizar un turno de registro en la fase CALIFICACION la anotación a la cual se le modificó el usuario aparece como si fuera temporal, debido a la modificación en el turno de correcciones, mostrando tanto la información definitiva como la información que se creó temporal. se adjuntan imágenes en las cuales se documenta y reproduce el error. Id Glpi 10745.

Al corregir anotaciones por turnos de correcciones, estos turnos avanzan de fase correctamente e imprimen los formularios de corrección correctamente y finalizan sin inconvenientes, pero al asociar los folios corregidos a otro turno de correcciones o registro, las anotaciones corregidas anteriormente quedan con la información definitiva y con información temporal que hace conflicto con el trámite que se realiza en ejecutar corrección simple y calificación. Mesa de Ayuda. Id Glpi 11346.

## **DESARROLLO**

### **Análisis de Requerimiento:**

El problema radica cuando se modifican anotaciones de un folio, el campo no idWorkFlow no era guardado en la anotación temporal de la base de datos, cuando se intentaba borrar este temporal una vez ya se había hecho definitivo, se valida contra este campo(idWorkFlow), al no existir no lo borraba y por esto salía en cualquier pantalla la anotación aun en temporal.

### **Observaciones:**

Para solucionar este caso solo fue necesario guardar el idWorkFlow en el temporal de la anotación.

## **082 Resolución Turnos de Registro Manuales**

### **REQUERIMIENTO**

Resolución turnos de registro manuales. Al crear turnos por la pantalla administrativa manual turno registro, si el usuario no selecciona resolución (inscrito - devuelto), el sistema continúa con el trámite del turno y deja el turno sin resolución y no hay forma

por el sistema de colocar dicha resolución al turno ya creado. Se requiere escalar este caso a desarrollo para crear un acta de requerimiento que solucione este problema, haciendo que el sistema exija seleccionar un tipo de resolución para el turno si no se ha seleccionado al avanzar de paso en la pantalla manual turno registro. Mesa de Ayuda. Id Glpi 11440.

## DESARROLLO

### Análisis de Requerimiento:

Al momento de validar los campos requeridos de este formulario, no hace referencia al campo de la respuesta de calificación del resultado de la calificación. Nunca es validado.

### Observaciones:

Al momento de realizar las validaciones se agrego la validación de campo requerido para el campo de la respuesta de calificación del resultado de la calificación.

### Clase:

gov.sir.core.web.acciones.administracion.AWTurnoManualRegistro

### Método:

```
private EvnTurnoManualRegistro liquidarRegistro(HttpServletRequest request)
    throws AccionWebException
```

Código linea 1321:

```
/*
 * @author: Fernando Padilla
 * @change: Se valida que la respuesta de la resolución del resultado
 * de calificación haya sido seleccionada.
 */
//Se campura el historial del turno
String respuestaCalificacion
request.getParameter(CRespuestaCalificacion.RESPUESTA_CALIFICACION_TURNO_REGISTRO_ADMINISTRACION);
if(respuestaCalificacion == null) {
    exception.addError("Debe seleccionar la respuesta de la resolución del resultado de la calificación");
}
```

## 083 Casilla Valor y Saldo consignación

### REQUERIMIENTO

Casilla Valor y saldo consignación. Nos informa la funcionaria Mónica Robledo e informa que al ingresar una consignación ya registrada al momento de verificar las casillas de valor de la consignación y el saldo los valores no salen completos. Se toma por remoto el proceso de ingreso de consignación ya registrada se puede apreciar que los valores de las casillas Valor consignación y saldo consignación no salen completos, los valores son muy grandes y al parecer por esto no son visibles. Se adjunta pantalla para verificar información. Pereira. Id Glpi 11414.

## DESARROLLO

### Análisis de Requerimiento:

Al momento de realizar los pagos y elegir consignaciones ya registradas en la base de datos, el valor de la consignación y el valor del saldo no es formateado antes de mostrarlo en pantalla, cuando los valores son muy grandes se presentan de forma exponencial (ejemplo, 22000000 lo muestra como 2.2e7) no entendiendo el usuario cual es el verdadero valor de la consignación ni del saldo.

### Observaciones:

Se formatean estos valores antes de presentárselos al usuario. el formato es ##0.0  
(ejemplo 22000000 lo muestra como 2000000.0).

Clase:

gov.sir.core.web.acciones.comun.AWPago

Método:

```
public void doEnd(HttpServletRequest request, EventoRespuesta eventoRespuesta)
```

Código linea 921:

```
    /**
     * @author Fernando Padilla
     * @change Se agrega para formatear los valores de la consignación
     * cuando esta ya está registrada en la base de datos.
     * Caso mantis 0001929: Acta - Requerimiento No 083.
     */
    DecimalFormatSymbols simbolos = new DecimalFormatSymbols();
    simbolos.setDecimalSeparator('.');
    NumberFormat numberFormat = new DecimalFormat("##0.0", simbolos);
```

Código linea 939:

```
    /**
     * @author Fernando Padilla
     * @change Se agrega para formatear los valores de la consignación
     * cuando esta ya está registrada en la base de datos.
     * Caso mantis 0001929: Acta - Requerimiento No 083.
     */
    session.setAttribute(AWPago.SALDO_CONSIGNACION_YA_REGISTRADA,
    numberFormat.format(consignacion.getSaldoDocumento()));
    session.setAttribute(AWPago.VALOR_CONSIGNACION_YA_REGISTRADA,
    numberFormat.format(consignacion.getValorDocumento()));
```

#### DESPLIEGUE 16/07/2009 – 2.1.4

Actualización proceso creación directa de folio.

#### DESPLIEGUE 15/07/2009 – 2.1.3

Actualización procesos:

- Hacer definitivas las anotaciones.
- Deshacer cambios de folios.

#### DESPLIEGUE 14/07/2009 – 2.1.2

Actualización librería mapas.jar; en la cual se elimina la funcionalidad de prefabricados.

#### DESPLIEGUE 10/07/2009 – 2.1.1

Revisión y actualización del reporte 25 – Informe de recaudo del día (Tiracaja) en la versión 2.1.1 de mantenimiento del sistema SIR.

#### DESPLIEGUE 08/07/2009 – 2.1.0

#### INCIDENCIAS RESUELTA

ID GLPI	OBSERVACIONES
Pantalla administración – Reportes – Listado de Reportes	
9139	<p><b>REQUERIMIENTO</b></p> <p>Yerly se comunica y menciona que se tiene inconsistencia en el reporte 51, debido a que no descontó un turno anulado 2009-200-6-7311 del usuario diana.piragua como se referencia en el tiracaja de este usuario.</p> <p>Al revisar el tiracaja se observa que el valor total aplicado del usuario es \$8220730, pero en el Reporte 51 el valor es \$8267880, es decir el Reporte 51 no descontó el turno anulado 2009-200-6-7311. Por favor verificar.</p> <p><b>DESARROLLO</b></p> <p>Se modificó el script para que no sumara los turnos anulados.</p>

9211	<p><b>REQUERIMIENTO</b></p> <p>Se comunica Ángela Vergara e informa que al generar reportes los valores no coinciden. El valor correcto arrojado por los recibos físicos que se encuentran a la oficina es \$999.120 los reportes arrojan otros valores diferentes. Favor realizar el ajuste debido a que no coinciden los valores de los reportes. Los reportes inconsistentes son: reporte 25 (tiracaja) reporte 51</p> <p><b>DESARROLLO</b></p> <p>Se modifico el script del reporte 25 y procedimiento del reporte 51</p>
<b>REPORTES 25 Y 15</b>	
9076	<p><b>REQUERIMIENTO</b></p> <p>Informa Sixta que el turno 2007-148-4-5 no aparece en el reporte 15 ni en el reporte 25 del día 22/04/2009. Favor verificar.</p> <p><b>DESARROLLO</b></p> <p>El turno 2007-148-4-5 no aparece en el reporte 25(Informe de recaudos del día por usuario) del día 22/04/2009 por que el pago de la devolución se realizo el día 30/04/2009; y no se despliega en el reporte 15(Informe cuentas por pagar devoluciones) por que la devolución ya fue pagada.</p>
<b>REPORTE 90</b>	
9339	<p><b>REQUERIMIENTO</b></p> <p>Josefina se comunica e informa que intenta generar reportes entre enero y abril, pero no se generan sale en blanco. Se generan reportes de prueba de varias fechas entre enero y abril, y todos los reportes salen en blanco. Se hacen pruebas con varios usuarios, y por diferentes nodos. Se detectas que al verificar reportes cercanos a la fecha actual estos si salen de la forma correcta.</p> <p><b>DESARROLLO</b></p> <p>No se estaban desplegando datos en fechas anteriores ya que el reporte estaba validando que desplegara los turnos que se encontraran en la fase de entrega siempre y cuando no hayan pasado por la fase de entregado (Finalizado). Teniendo en cuenta las observaciones de Javier Rocha el reporte desplegara los turnos que tengan la fase de entrega sin validar que el turno se halla finalizado.</p>
<b>REPORTE REL 01</b>	
9756	<p><b>REQUERIMIENTO</b></p> <p>Informa Sandra Ramírez que al generar el Reporte Rel 01: Reporte genérico de relaciones con las relaciones 2009-070-895, 2009-070-896 y 2009-070-897 no se visualizan completos los turnos, favor verificar.</p> <p><b>DESARROLLO</b></p> <p>Se modifico el tamaño del campo ID RELACION.</p>
<b>APLICACIÓN</b>	
9595	<p><b>REQUERIMIENTO</b></p> <p>Nos informan de la oficina de registro que no están saliendo los certificados inmediatos impresos. Los turnos a los que no salieron los certificados son: 2009-230-1-49655 2009-230-1-49582 2009-230-1-49752 2009-230-1-49379 Por favor enviar PDF de los certificados para enviárselos a la oficina de</p>

	<p>registro.</p> <p>Nos informan de la oficina que los folios 230-123165, 230-147565, 230-127570 no están saliendo con todas las anotaciones.</p>
	<p><b>DESARROLLO</b></p> <p>Se generaron y enviaron los certificados.</p>
7078	<p><b>REQUERIMIENTO</b></p> <p>Lady Clavijo reporta que el sistema presenta error cuando el cajero permanece dentro del rol, es decir empieza a radicar turnos seguidos, y sistema no permite el proceso.</p> <p>Se comunica en el día de hoy la oficina de Honda la usuaria Luz Mary reportando el siguiente problema: El valor a pagar no coincide con el valor Liquidado, este error se presente en el rol liquidación registro el cajero hace la liquidación y cuando va en el turno 5 o 6 genera el error. Este problema es heredado y viene desde el último despliegue que corrió SIR.</p> <p><b>DESARROLLO</b></p> <p>Análisis: Este problema no pudo ser replicado en desarrollo, pero se encontró que no se estaba quitando de la sesión el objeto pago que contiene los elementos para realizar la liquidación.</p> <p>Solución: Se removió el atributo PAGO de la sesión.</p>
8870	<p><b>REQUERIMIENTO</b></p> <p>Nos informan de la oficina de registro que algunos certificados están saliendo con la información incompleta. Nos informan de la oficina de registro que algunos certificados están saliendo con la información incompleta, por lo tanto se indica a la funcionaria Mery Tamayo indicarnos los turnos y folios correspondientes a los que no les está saliendo la información completa y enviar los certificados que salieron incompletos, se verifica y algunas anotaciones fueron calificadas en el mes de abril y son las que no están saliendo reflejadas en los certificados.</p> <p>Los turnos son los siguientes:</p> <p>2009-230-1-40674 con folio 230-126405 tiene 8 anotaciones y solo salieron impresas en el certificado 5.</p> <p>2009-230-1-41042 con folio 230-35523 tiene 28 anotaciones y solo salieron impresas en el certificado 27.</p> <p>Por favor tener en cuenta que en la oficina al momento de verificar que no sale completa la información proceden a realizar la reimpresión de los mismos y si sale la información completa. Villavicencio.</p> <p>Nos informan de la oficina de la ceja que está pasando lo mismo, tenemos como ejemplo el folio 017-8171 el cual tiene 9 anotaciones pero al momento de imprimir el certificado este sale con 8 anotaciones.</p> <p>Se anexa el certificado. Por favor tener en cuenta que en la oficina al momento de verificar que no sale completa la información proceden a realizar la reimpresión de los mismos y si sale la información completa.</p> <p><b>DESARROLLO</b></p> <p>Análisis: el problema radica en el proceso de generar las impresiones, es muy costoso en memoria del cliente y al no tener memoria disponible simplemente no sigue realizando el proceso.</p> <p>Solución: Se modifico el print_client.jnlp aumentando el costo de memoria máxima a 512 MB.</p>
8934	<b>REQUERIMIENTO</b>

	<p>Necesito que por favor me regalen el id de impresión de el turno 2009-300-6-17702 es un turno con mas de 1000 matriculas hemos intentado varias veces enviar el temporal pero no nos sale. Por favor indicar el número del ID del formulario para generar la impresión. Bucaramanga. Bucaramanga.</p> <p><b>DESARROLLO</b></p> <p>Análisis: el problema radica en el proceso de generar las impresiones, es muy costoso en memoria del cliente y al no tener memoria disponible simplemente no sigue realizando el proceso.</p> <p>Solución: Se modifco el print_client.jnlp aumentando el costo de memoria máxima a 512 MB.</p>
9305	<p><b>REQUERIMIENTO</b></p> <p>Nos informan de la oficina de registro que al tratar de crear una anotación con departamento Caqueta-Florencia en el sistema genera error. Nos comunicamos con la oficina para realizar el proceso por remoto, toman el turno 2009-420-3-198 y realizan el proceso para crear una anotación, pero al momento de seleccionar en departamento Caquetá, ciudad Florencia y notaria segunda el sistema notifica el error que se adjunta. Florencia.</p> <p><b>DESARROLLO</b></p> <p>Análisis: Se encontró inconsistencia en los datos para esta consulta en la tabla SIR_NE_VEREDA.</p> <p>Solución: Se actualizó la tabla SIR_NE_VEREDA en el campo V.VRDA_CABECERA = '0' cuando id_vereda es 390 y id_municipio es 001 y id_departamento es '18'.</p>
8729	<p><b>REQUERIMIENTO</b></p> <p>Impresión Liquidación Fotocopias. En el recibo generado al liquidar fotocopias, en el texto donde se especifica las tarifas de cobro, esta información se encuentra desactualizada. Por favor modificar con las nuevas tarifas, según Resolución 0035 del 9 de enero de 2009, que corresponde al incremento de tarifas de acuerdo al IPC.      Medio Óptico o microfilmado \$860      Archivo físico \$540.      Por favor escalar este caso a desarrollo para su solución general. Grupo SAMANIEGO.</p> <p><b>DESARROLLO</b></p> <p>Análisis: Este texto esta en la base de datos, en la tabla SIR_OP_TEXTO_IMPRIMIBLE.</p> <p>Solución: Se realizó un UPADTE sobre la tabla SIR_OP_TEXTO_IMPRIMIBLE para el cambio del texto de la impresión de fotocopias</p>
8727	<p><b>REQUERIMIENTO</b></p> <p>Orden de impresión de recibos de turnos certificados masivos.</p> <p>Cuando se expiden certificados masivos, al imprimir los recibos correspondientes por cada turno de certificados individuales, invierte el orden de impresión de los recibos y no existe coincidencia del nro. del recibo preimpreso con el consecutivo asignado por el sistema. Como ejemplo, cito:      Turno 2009-250-1-1792 Recibo físico 35111190 Recibo Consecutivo Sistema 35111191      Turno 2009-250-1-1791 Recibo físico 35111191 Recibo Consecutivo Sistema 35111190      Por favor escalar este caso a desarrollo para la solución general de este</p>

	<p>proceso. Grupo SAMANIEGO</p> <p><b>DESARROLLO</b></p> <p>Este error no se replicó en desarrollo (usuarios usuarios.pruebas y HUGO.MANTILLA), ni en producción (usuario usuarios.pruebas); aun así se revisó el flujo de la aplicación y no se encontró en el código fuente algo que cause este problema.</p>
9369	<p><b>REQUERIMIENTO</b></p> <p>Javier se comunica y menciona que al intentar avanzar los turnos de corrección, de la fase de corrección simple, el sistema genera un error. Se intenta modificar el orden de las anotaciones y editar un ciudadano. Por favor revisar el error, se adjunta imagen. Montería.</p> <p><b>DESARROLLO</b></p> <p>Sobre el error reportado y escalado al grupo DBA con el tiquete IM100460, se obtuvo el siguiente diagnostico y solución:</p> <pre>/"01/06/09 12:07:49 (martdurr): SE RECOMPILO TODO EL AMBIENTE QUEDANDO 317 OBJETOS INVALIDOS. SE DESHABILITO EL TRIGGER QUE EVITABA COMPILEACIONES EN AMBIENTE PRODUCTIVO."</pre>
9366	<p><b>REQUERIMIENTO</b></p> <p>Román se comunica y menciona que al intentar avanzar los turnos de corrección, de la fase de corrección simple, el sistema genera un error. Se intenta modificar el orden de las anotaciones y editar un ciudadano. Purificación.</p> <p><b>DESARROLLO</b></p> <p>Sobre el error reportado y escalado al grupo DBA con el tiquete IM100460, se obtuvo el siguiente diagnostico y solución:</p> <pre>/"01/06/09 12:07:49 (martdurr): SE RECOMPILO TODO EL AMBIENTE QUEDANDO 317 OBJETOS INVALIDOS. SE DESHABILITO EL TRIGGER QUE EVITABA COMPILEACIONES EN AMBIENTE PRODUCTIVO."</pre>
9386	<p><b>REQUERIMIENTO</b></p> <p>Es el mismo error de correcciones SQL:ora-00604. Gracias. Guamo.</p> <p><b>DESARROLLO</b></p> <p>Sobre el error reportado y escalado al grupo DBA con el tiquete IM100460, se obtuvo el siguiente diagnostico y solución:</p> <pre>/"01/06/09 12:07:49 (martdurr): SE RECOMPILO TODO EL AMBIENTE QUEDANDO 317 OBJETOS INVALIDOS. SE DESHABILITO EL TRIGGER QUE EVITABA COMPILEACIONES EN AMBIENTE PRODUCTIVO."</pre>
9365	<p><b>REQUERIMIENTO</b></p> <p>Informa Yamina que al avanzar el turno 2009-146-3-67 EJECUTAR CORRECCION SIMPLE X-EJEC.CORR.146 aparece error, se le pidió enviar error por fax. Relacionado con el caso 9366. Lorica.</p> <p><b>DESARROLLO</b></p> <p>Sobre el error reportado y escalado al grupo DBA con el tiquete IM100460, se obtuvo el siguiente diagnostico y solución:</p> <pre>/"01/06/09 12:07:49 (martdurr): SE RECOMPILO TODO EL AMBIENTE QUEDANDO 317 OBJETOS INVALIDOS.</pre>

	SE DESHABILITO EL TRIGGER QUE EVITABA COMPILACIONES EN AMBIENTE PRODUCTIVO."
9363	<p><b>REQUERIMIENTO</b></p> <p>Al turno 2009-290-3-398 se le incluyó la anotación 12, pero esta la dejaron como 5 y al momento de avanzar el turno el sistema notifica error.</p> <p>El turno 2009-290-3-398 se encuentra en la fase ejecutar corrección simple, y al folio 290-48695 se le incluyó la anotación 12, pero la usuario cambió el orden cronológico de las anotaciones y dejó la anotación 12 que creó como anotación 5, pero al momento de avanzarlo de la fase ejecutar corrección simple el sistema está notificando el error que se adjunta. Pereira.</p> <p><b>DESARROLLO</b></p> <p>Sobre el error reportado y escalado al grupo DBA con el tiquete IM100460, se obtuvo el siguiente diagnostico y solución:</p> <p>/"01/06/09 12:07:49 (martdurr): SE RECOMPILO TODO EL AMBIENTE QUEDANDO 317 OBJETOS INVALIDOS. SE DESHABILITO EL TRIGGER QUE EVITABA COMPILACIONES EN AMBIENTE PRODUCTIVO."</p>
9376	<p><b>REQUERIMIENTO</b></p> <p>Informa Juan Carlos que al avanzar el turno 2009-120-3-278 EJECUTAR CORRECCION SIMPLE X-EJEC.CORR.120 aparece error:"validación servicio negocio orden de anotaciones error SQL:ORA_00604.". Popayán.</p> <p><b>DESARROLLO</b></p> <p>Sobre el error reportado y escalado al grupo DBA con el tiquete IM100460, se obtuvo el siguiente diagnostico y solución:</p> <p>/"01/06/09 12:07:49 (martdurr): SE RECOMPILO TODO EL AMBIENTE QUEDANDO 317 OBJETOS INVALIDOS. SE DESHABILITO EL TRIGGER QUE EVITABA COMPILACIONES EN AMBIENTE PRODUCTIVO."</p>
9374	<p><b>REQUERIMIENTO</b></p> <p>Validación (ServicioNegocio) Orden de anotación SQL:ORA-00604 El usuario SIR no puede crear ni modificar objetivos. Cartagena.</p> <p><b>DESARROLLO</b></p> <p>Sobre el error reportado y escalado al grupo DBA con el tiquete IM100460, se obtuvo el siguiente diagnostico y solución:</p> <p>/"01/06/09 12:07:49 (martdurr): SE RECOMPILO TODO EL AMBIENTE QUEDANDO 317 OBJETOS INVALIDOS. SE DESHABILITO EL TRIGGER QUE EVITABA COMPILACIONES EN AMBIENTE PRODUCTIVO."</p>
9373	<p><b>REQUERIMIENTO</b></p> <p>Al tratar de avanzar los turnos de la fase ejecutar corrección simple el sistema notifica error. Nos informan de la oficina de registro que al tratar de avanzar los turnos 2009-420-3-220 y 2009-420-3-223 de la fase ejecutar corrección simple el sistema notifica los errores que se adjuntan. Florencia.</p> <p><b>DESARROLLO</b></p> <p>Sobre el error reportado y escalado al grupo DBA con el tiquete IM100460, se obtuvo el siguiente diagnostico y solución:</p> <p>/"01/06/09 12:07:49 (martdurr): SE RECOMPILO TODO EL AMBIENTE QUEDANDO 317 OBJETOS INVALIDOS.</p>

	<p>SE DESHABILITO EL TRIGGER QUE EVITABA COMPILACIONES EN AMBIENTE PRODUCTIVO."</p>
9372	<p><b>REQUERIMIENTO</b>  Validación Servicio-NEgocios Erro SQL-ORA-00604. El usuario Sir no puede crear ni modificar objetos. Pereira.</p> <p><b>DESARROLLO</b>  Sobre el error reportado y escalado al grupo DBA con el tiquete IM100460, se obtuvo el siguiente diagnostico y solución:</p> <p>/"01/06/09 12:07:49 (martdurr):  SE RECOMPILO TODO EL AMBIENTE QUEDANDO 317 OBJETOS INVALIDOS.  SE DESHABILITO EL TRIGGER QUE EVITABA COMPILACIONES EN AMBIENTE PRODUCTIVO."</p>
9136	<p><b>REQUERIMIENTO</b>  El incidente detectado en el sistema, SNR hace requerimiento mediante correo electrónico para escalarlo a desarrollo y poder dar una solución general al problema reportado:</p> <p>ingenieros</p> <p>Reciban un cordial saludo,</p> <p>El registrador Principal de la Orip de Santa Marta, mediante oficio ORIPSM-123DA informa que la reimpresión de certificados el Aplicativo SIR, permite reimprimir sin limites un certificado de tradición, además permite reimprimir de cualquier vigencia, como se puede comprobar, los cambios de matricula y certificados generados con las radicaciones 2008-080-1-34560.  Lo establecido por la SNR es que el sistema permita al Usuario de Entrega de Certificados, la reimpresión de certificados una sola vez, si en los casos que el sistema no le permita la reimpresión del certificado a este usuario, el jefe operativo debe tener la reimpresión de certificados solamente por una vez.</p> <p>Por lo tanto se solicita implementar los controles para la reimpresión de certificados para reducir el riesgo de reimpresión indefinidamente.</p> <p>Este es una falla heredada del operadora anterior. Snrinformatica.</p> <p><b>DESARROLLO</b>  Análisis: Luego de verificar el flujo de trabajo de este caso, se concluyo que el sistema permite restringir el número de reimpresiones de certificados por medio de la actualización de un campo de base de datos. se debe advertir que según caso TFS 4588(Siglo 21), el sistema permite que los usuarios con rol de registrador puedan reimprimir estos certificados numero de veces que deseen.</p> <p>Solución:  Se modifico la tabla SIR_OP_LOOKUP_CODES en el campo lkcd_valor con el valor de 2 cuando el id_tipo es igual a CERTIFICADOS y id_codigo es igual a MAX_IMPR_SUP.</p> <p>Desde el esquema SIR, ejecutar:</p> <pre>UPDATE   SIR_OP_LOOKUP_CODES I SET   I.lkcd_valor='2'</pre>

	<p>WHERE</p> <p>L.id_tipo = 'CERTIFICADOS' and L.id_codigo = 'MAX_IMPR_SUP';</p>
9805	<p><b>REQUERIMIENTO</b></p> <p>Esmeralda se comunica y menciona que el turno 2009-420-3-256 no avanza, genera el error: Fallo en el servicio para hacer definitivos los datos temporales. Por favor verificar para que el turno avance a mesa de control correcciones.</p> <p><b>DESARROLLO</b></p> <p>Análisis: En el proceso de hacer definitivas las anotaciones se presenta un error al momento de borrar los datos temporales en la BD, trata de borrar datos inexistente y al no encontrarlos devuelve la excepción: "The Instance is transient".</p> <p>Solución: Se valida la existencia dentro del objeto AnotacionTMP la nulidad del id la anotación temporal para poder borrar este registro. esta solución se encuentra solo en ambiente de desarrollo y solo hasta la tercera despliegue se verá en el ambiente productivo.</p>

REQUERIMIENTOS RESUELTOS
006_141_Número de teléfono recibo
<p><b>Descripción del Requerimiento:</b>            Número de teléfono recibo.            SIR-294            Según oficio 5284, es necesario que en los campos de radicación donde exista la opción para digitar número de teléfono, si se digitán valores en este campo estos valores se impriman en el recibo de radicación.</p> <p><b>SEGUIMIENTO ADA:</b>            Se requiere mayor claridad en el requerimiento.            No se especifica que tipo de radicación Se requiere mayor claridad en el requerimiento.            No se especifica que tipo de radicación.</p> <p><b>OBSERVACIONES SNR:</b>            En las pantallas de radicación de los procesos de registro, correcciones Devoluciones de Dinero y fotocopias en los datos del solicitantes en el campo de Teléfono si se ingresa información del número telefónico, al imprimir el recibo correspondiente se debe visualizar esa información.</p> <p><b>Describir el Impacto en el Desempeño de la Aplicación</b>            No es de alto impacto, para el desempeño y operativamente es necesario.</p> <p><b>Ánalysis de Requerimiento:</b>            Se identificaron los archivos a modificar para este requerimiento, son:            sir/core/negocio/modelo/imprimibles/ImprimibleFotocopiaCrearSolicitud.java,            sir/core/negocio/modelo/imprimibles/ImprimibleRecibo.java.</p> <p><b>Observaciones:</b>            Se agrego en los métodos respectivos los números telefónicos del solicitante del servicio.</p>
<b>007_141_Mayor Valor simplificación proceso</b>

**Descripción del Requerimiento:**

Mayor Valor: simplificación proceso.

SIR-289.

Se digita la nota informativa que corresponde al mayor valor y en otro paso adicional se crea el mayor valor. Esta situación genera perdida de tiempo.

**SEGUIMIENTO ADA:**

Especificar el requerimiento.

**OBSERVACIONES SNR:**

Se aclara este requerimiento, la inconformidad proviene de las ORIPS manuales que migraron al SIR, toda vez que estas manuales no incluían a ningún sistema un mayor valor solo generaban la nota informativa, las ORIP de folio magnético, debían generar la nota para el ciudadano (nota informativa o devolutiva según la denominara la orip) y luego cargaban en el sistema el mayor valor, funcionalidad que se desarrolló igualmente en SIR,

Al cargar al SIR o crear el mayor valor en SIR se encuentra un campo que debe digitar la razón del mayor valor esta caja de texto es muy grande, la mayoría de las oficinas digitaban toda la información para el ciudadano en esta caja de texto , teniendo en cuenta que lo que se digita en esta caja se refleja en una columna en el reporte de mayores valores ingresados del día, se debe restringir y/o minimizar la caja de texto, se les hace dispendioso volver a ingresar nuevamente en la nota informativa toda la información digitada en la caja anterior de ahí nace este requerimiento. Se soluciona achicando la caja de texto de la razón del mayor valor a una línea.

**Describir el Impacto en el Desempeño de la Aplicación**

Funcionalmente es recomendable realizar el ajuste, no es de alto impacto.

**Análisis de Requerimiento:**

Se solicita por parte del equipo del SNR cambiar el formato al campo en el cual se ingresa la razón de pago en el proceso de registrar el Mayor valor, se solicita disminuir el tamaño del campo y limitarlo a una línea.

**Observaciones:**

Se modificó la Instrucción anterior que construía el campo por:

```
<td>
<input type="text" id="RAZON_MAYOR_VALOR"
MaxCharsRAZON_MAYOR_VALOR="null" name="RAZON_MAYOR_VALOR"
class="camposformtext" maxlength="130" style="width:822px; height:20px" />
<td width="20%" align="left"> Máximo 130 Caracteres</td>
```

La cual crea un campo de texto de máximo 130 caracteres y de las siguientes dimensiones style="width:822px; height:20px".

**008\_141\_Correcciones Eliminar Folio SegregadoMatriz****Descripción del Requerimiento:**

Correcciones: Eliminar Folio Segregado/Matriz

SIR-281

Cuando se necesita eliminar un folio segregado y/o un folio matriz de otro folio, el

sistema exige que se asocien las matriculas inmobiliarias que se van a eliminar. Partiendo del presupuesto que sobre el número de matricula que se va a eliminar del campo matrices y segregados, no se va a efectuar ninguna corrección, consideramos que no es necesario su bloqueo.

**SEGUIMIENTO ADA:**  
Especificar el Requerimiento.

**OBSERVACIONES SNR;**

Este caso debe cerrase toda vez que la funcionalidad es correcta ya que el folio que se esta desvinculando ya se ha publicitado con ese vinculo, por lo tanto el folio eliminado debe asociarse al turno de corrección para efectuar la corrección de eliminar el numero de folio matriz y efectuar salvedad.

**Describir el Impacto en el Desempeño de la Aplicación**

No es de alto impacto a nivel funcional.

**Análisis de Requerimiento:**

La incidencia está resuelta por observaciones de SNR

**Observaciones:**

Según las observaciones de SNR:

"Este caso debe cerrase toda vez que la funcionalidad es correcta ya que el folio que se esta desvinculando ya se ha publicitado con ese vinculo, por lo tanto el folio eliminado debe asociarse al turno de corrección para efectuar la corrección de eliminar el numero de folio matriz y efectuar salvedad."

**016\_141 Certificados Antiguo Sistema Seguridad en creación de hoja de ruta**

**Descripción del Requerimiento:**

Certificados Antiguo Sistema: Seguridad en creación de hoja de ruta.

SIR-245.

Actualmente el aplicativo no permite la creación de hoja de ruta y de folios para aquellos casos en la oficina requiera pasar la tradición de un inmueble de los libros de antiguo sistema a nuevo sistema en forma oficiosa, es decir sin que medie solicitud del usuario. Se sugiere que esta funcionalidad se implemente con las mayores seguridades y restricciones del caso para evitar la creación de folios de matricula inmobiliaria sin el respectivo soporte jurídico y operativo.

**SEGUIMIENTO ADA:**  
Se requiere mayor especificación del requerimiento.

**OBSERVACIONES SNR:**

Los folios se pueden crear por la pantalla administrativa “Creación directa de folios”, cerrar caso, se solicitó creación de ROL para ingreso de folios oficiosamente.

**Describir el Impacto en el Desempeño de la Aplicación**  
No se considera de impacto.

**Análisis de Requerimiento:**  
Según SNR, los folios se pueden crear por la pantalla administrativa “Creación directa de

folios", cerrar caso, se solicitó creación de ROL para ingreso de folios oficiosamente."

## 017\_141\_Fase Calificación Ingreso inmediato a la lista de turnos

### Descripción del Requerimiento:

Fase Calificación: Ingreso inmediato a la lista de turnos.

SIR- 239

Se sugiere que una vez el funcionario ingrese con el rol de calificador, el sistema lo lleve directamente a los turnos que tiene asignados en esta fase, sin necesidad de pasar por las 3 pantallas anteriores (estaciones, calificaciones, calificación).

### SEGUIMIENTO ADA:

Validar la necesidad de este requerimiento, debido a que esto afecta la lógica de despliegue del sistema.

### OBSERVACIONES SNR:

La solicitud se hizo también con la funcionalidad de certificados de pertenencia:

"*Simplificación al ingresar a la fase\_015\_141*" formato externo 014, SIR-248.

Debe desarrollarse este requerimiento ,teniendo en cuenta que fue posible realizarlo en el ROL DE CONFORNTACION y se nota la agilidad para llegar al listado de turnos.

### Describir el Impacto en el Desempeño de la Aplicación

El cambio es muy impactante en desarrollo Sería de alto impacto en el aplicativo haciéndolo mucho mas ágil.

### Análisis de Requerimiento:

No se estaba validando si el rol sólo tiene asociado una estación, un solo proceso y una sola fase. En estos casos no es necesario mostrarle las opciones al usuario si no ir directamente a la página donde se muestran los turnos obviando las dos pantallas que precede a esta.

### Observaciones:

Para solucionar esta incidencia se realizaron los siguientes cambios en la aplicación:

ARCHIVO	MODIFICACIÓN
gov.sir.core.web. WebKeys.java	<b>Se crearon las constantes NUM_ESTACIONES,</b> <b>NUM_PROCESOS y NUM_FASES</b>
gov.sir.core.web.helpers .comun. MenuHelper.java	<b>Se modificó el método</b> “drawGUI(HttpServletRequest request, JspWriter out)” <b>cambiando la línea 243 “out.write("&lt;form name=</b> “ <i>Estaciones</i> <b>” method=</b> “post <b>\”</b> <b>action=</b> “estaciones.view\”>”),” <b>por</b> “out.write(“<form name=“ <i>EstacionesV</i> <b>”</b> <b>method=</b> “post <b>\”</b> action=“estaciones.view\”>”);”  <b>Debido a que luego en el código se continua</b> <b>renderizando formularios con el mismo nombre y</b> <b>necesitamos enviar el formulario “Estaciones” si existe</b>

	<b>un solo proceso en la lista.</b>
<b>gov.sir.core.web.navegacion.comun. ControlNavegacionSeguridad.java</b>	<p><b>Se modificó el método</b>  <b>“procesarNavegacion(HttpServletRequest request)”</b>  <b>agregándole las líneas</b>  <b>“request.setAttribute(WebKeys.NUM_ESTACIONES, new Integer(((List)  session.getAttribute(WebKeys.LISTA_ESTACIONES)).size()  ()));”, “request.setAttribute(WebKeys.NUM_PROCESOS,  new Integer(((List)  session.getAttribute(WebKeys.LISTA_PROCESOS)).size()  ()));” y “request.setAttribute(WebKeys.NUM_FASES, new  Integer(fases.size()));”</b></p> <p><b>para guardar en el request los atributos que nos dirán el número de estaciones, procesos y fases, respectivamente, que tiene asociado el rol para redirigir la petición en los casos en que sólo sea uno el valor de la lista.</b></p>
<b>WEB-INF. jspSeguro.comun. LISTAR_PROCESOS.jsp</b>	<p><b>Se modificó la página agregándole las líneas:</b>  <b>“&lt;%@page import="gov.sir.core.web.WebKeys" %&gt;</b>  <b>&lt;%</b>  <b>    Integer numEstaciones =</b>  <b>(Integer)request.getAttribute(WebKeys.NUM_ESTACIONES);</b>    <b>    request.removeAttribute(WebKeys.NUM_ESTACIONES);</b>  <b>    Integer numProcesos =</b>  <b>(Integer)request.getAttribute(WebKeys.NUM_PROCESOS);</b>    <b>    request.removeAttribute(WebKeys.NUM_PROCESOS);</b>  <b>    %&gt;</b>  <b>    &lt;script&gt;</b>  <b>    &lt;%</b>  <b>        if(numEstaciones!=null &amp;&amp;</b>  <b>numEstaciones.intValue() == 1 &amp;&amp; numProcesos!=null</b>  <b>&amp;&amp; numProcesos.intValue() == 1){</b>  <b>        %&gt;</b>  <b>            document.Estaciones.submit();</b>  <b>        %&gt;</b>  <b>        }</b>  <b>    %&gt;</b>  <b>    &lt;/script&gt;”</b></p> <p><b>para reenviar la petición al siguiente pantalla solo en los casos en que las listas sólo tengan una sola estación y un solo proceso. También se elimina del request los atributos “NUM_ESTACIONES” y “NUM_PROCESOS”</b></p>
<b>WEB-INF. jspSeguro.comun.</b>	<b>Se modificó la página agregándole las líneas:</b> <b>“&lt;%@page import="gov.sir.core.web.WebKeys" %&gt;</b>

<b>LISTAR_FASES.jsp</b>	<pre> &lt;%     Integer numFases = (Integer)request.getAttribute(WebKeys.NUM_FASES);     request.removeAttribute(WebKeys.NUM_FASES); %&gt; &lt;script&gt; &lt;%     if(numFases!=null &amp;&amp; numFases.intValue() == 1){ %&gt;         document.FASE.submit(); &lt;%     } %&gt; &lt;/script&gt;” </pre> <p><b>para reenviar la petición al siguiente pantalla en los casos en que las listas sólo tengan una sola fase.</b></p> <p>También se elimina del request el atributo “NUM_FASES”</p>
-------------------------	---

### 019\_141\_Asociar liquidación de solicitud vinculada liquidador registro

#### **Descripción del Requerimiento:**

Asociar liquidación de solicitud vinculada liquidador registro.

SIR-228

Existe un botón denominado "asociar liquidación de solicitud vinculada", el cual no funciona y tampoco cumple la finalidad para la que se pretendió crear.

#### SEGUIMIENTO ADA:

Especificar la funcionalidad para la cual fue creada inicialmente el botón.

**OBSERVACION SNR:** La funcionalidad se creó para el rol liquidador registro, en la pantalla de inicio de la liquidación esta el botón arriba mencionado, pero no está funcionando actualmente, debe permitir incluir en el campo de texto el número de turno asignado para la solicitud y una vez agregado este validar la liquidación y llevarlo inmediatamente a la pantalla de registrar el pago, toda vez que si el ciudadano que se acerca a la ventanilla del liquidador trae la liquidación antes generada solo faltaría validarla y pagar, el liquidador no tendría que salir al menú a cambiar de rol.

#### **Análisis de Requerimiento:**

Se solicita que al botón "asociar liquidación de solicitud vinculada" del proceso que pertenece al Rol Liquidador Registro se le agregue la funcionalidad de incluir el numero del turno generado y proceder a realizar el pago, sin la necesidad de cambiar de Rol.

#### **Observaciones:**

Se opto por clonar la funcionalidad utilizada para el proceso de Registro de Pago del Rol Cajero.

La cual antes de realizar el proceso tiene la salvedad de verificar si el usuario tiene el Rol de Cajero, el cual se hace necesario para verificar que el usuario que realice el proceso desde el Rol Liquidador Registro también posea este Rol para realizar el proceso.

A continuación mostramos los cambios realizado en el archivo REGISTRO\_DOCUMENTOS.jsp

```

<!-- JAlcazar 29/05/2009
 *Se modifco el TextHelper tornandose similar al definido en el proceso Cajero-
>Registro
 *Se Utiliza una nueva funcion JavaScript en el boton llamada cambiarAccion2 la cual
tiene el mismo
    comportamiento del proceso Cajero->Registro
-->
<td>
L-<%=sYear%>-<%=sCirculo%>-<%=sProceso%>-
<%
// :turnovinculado: render the helper

try {

    TextHelper textHelper = new TextHelper();
    textHelper.setNombre("AUXA");
    textHelper.setCssClase("camposformtext");
    textHelper.setId("AUXA");
    textHelper.setSize("15");
    textHelper.setTipo("text");
    textHelper.render(request, out);

}

catch( HelperException re ){
    out.println( "ERROR " + re.getMessage() );
}

%>
<!-- JAlcazar 29/05/2009
 *Script para limpiar el input AUXA
-->
<script>
    if(document.all("AUXA")){
        document.all("AUXA").value = "";
    }

</script>
</td>

<td align="right" rowspan="2">
<table border="0" cellpadding="0" cellspacing="2" class="camposform">
<tr>
    <td style="padding-left:20px;width:160px;">Asociar Número de Liquidación
Para Registrar Pago</td>
    <td><a
href="javascript:cambiarAccion2('<%=AWLiquidacionRegistro.BUSCAR_SOLICITUD
_PAPO%>','<%=sYear%>','<%=sCirculo%>','<%=sProceso%>')"></a></td>
</tr>
</table>
</td>

/* JAlcazar 29/05/2009
 *Se Agrego una nueva funcion JavaScript llamada cambiarAccion2 la cual tiene el
mismo
comportamiento del proceso Cajero->Registro
*/
function cambiarAccion2(text,anio,circulo,proceso,sec) {
document.REGISTRO.<%=AWLiquidacionRegistro.ID_SOLICITUD%>.value=ani
o+"-"+circulo+"-"+proceso+"-"+
document.REGISTRO.AUXA.value;
document.REGISTRO.ACCTION.value = text;
document.REGISTRO.submit();
}

```

### 020\_141\_Campos Antiguo Sistema Liquidador Registro

**Descripción del Requerimiento:**

Campos Antiguo Sistema Liquidador Registro.

SIR-225.

En cuanto a los campos a diligenciar en el formulario de radicación, el referente a datos de antiguo sistema, debería ser uno solo con suficientes caracteres que le permitan al funcionario incluir la información necesaria.

SEGUIMIENTO ADA:

Validar el requerimiento con los funcionarios de la SNR, lo anterior de acuerdo a que la información perdería su consistencia a nivel de datos, puesto que un usuario X, podría colocar simplemente números asumiendo que esto se refiere al tomo y otro colocar la palabra tomo: , Libro: , Pagina: , etc.

OBSERVACIONES SNR:

Verificar pantalla de radicación a ver que campos se pueden eliminar, verificar si a nivel de estructura de datos está almacenado en campos diferentes una vez efectuada la verificación y si esta información no es relevante para alguna validación en otra fase y proceso se sugiere eliminar el campo documento, fecha y numero, municipio y oficina de origen, solo dejar los campos libro y tomo y el campo comentario siendo opcional el diligenciar el comentario o los campos libro y tomo.

**Describir el Impacto en el Desempeño de la Aplicación**

No es de alto impacto.

**Análisis de Requerimiento:**

En los formularios para ingresar y mostrar la información de Antiguo Sistema se debe quitar los campos de "Datos del documento" y "Oficina de origen (Documento)".

**Observaciones:**

Se realizaron los siguientes cambios:

- \* En la clase "gov.sir.core.web.helpers.comu.MostrarAntiguoSistemaHelper.java", en el método "drawGUI(HttpServletRequest request, JspWriter out)", se colocó display none en los estilos de las tablas de los campos que se deseaban quitar (style="display:none") para no mostrar estos campos del formulario, además se colocaron TODO's informando donde se realizaron los cambios para eliminar el código donde están los campos en vez de ocultarlos, si la SNR está de acuerdo con el cambio.
- \* En la clase "gov.sir.core.web.helpers.comun.AntiguoSistemaHelper.java", en el método "drawGUI(HttpServletRequest request, JspWriter out)", se colocó display none en los estilos de las tablas de los campos que se deseaban quitar (style="display:none") para no mostrar estos campos del formulario, además se colocaron TODO's informando donde se realizaron los cambios para eliminar el código donde están los campos en vez de ocultarlos, si la SNR está de acuerdo con el cambio.

Hay que aclarar a la SNR que los folios que fueron creados con la información del Antiguo Sistema "Datos del documento" y "Oficina de origen" serán desplegados en el sistema sin dicha información.

**022\_141\_Simplificación de la operación Modulo Correcciones****Descripción del Requerimiento:**

Simplificación de la operación Modulo Correcciones.

**SIR-202**

Los usuarios del sistema sugieren que se simplifiquen las pantallas en el proceso de correcciones para agilizar más su operatividad. (DR E - 1004).

**SEGUIMIENTO ADA:**

Se requiere documentar este requerimiento de igual forma como fue documentado el proceso de devoluciones.

**OBSERVACIONES SNR:**

En el proceso de Correcciones al ingresar al Rol Responsable de Correcciones se debe eliminar el paso y pantalla de los botones “CORRECCIONES y CORRECCION” debe llevar al usuario directamente al pantallazo de las fases “REVISION Y ANALISIS – Y REVISAR Y APROBAR” de esta pantalla se debe eliminar el botón CORRECCIONES, toda vez que la funcionalidad que tenía este botón se fusionó con el rol de mesa de control correcciones..En los ROLES AUXILIAR DE CORRECCIONES Y MESA DE CONTROL CORRECCIONES una vez se ingrese al rol debe llevar al Usuario al Pantallazo del listado de turnos.

**Describir el Impacto en el Desempeño de la Aplicación**

El modulo de correcciones en general es de alto impacto para las ORIPS.

**Análisis de Requerimiento:**

Se solicita por parte del personal de SNR eliminar el paso y pantalla de los botones “CORRECCIONES y CORRECCION” debe llevar al usuario directamente a la pantalla de las fases “REVISION Y ANALISIS – Y REVISAR Y APROBAR”, de esto se debe eliminar el botón CORRECCIONES además se solicita que en los roles ROLES

AUXILIAR DE CORRECCIONES Y MESA DE CONTROL CORRECCIONES la pantalla envíe al usuario directamente al listado de turnos.

**Observaciones:**

Después de realizar el seguimiento al Flujo del Proceso se identifico lo siguiente:

1) Eliminar Botón CORRECIONES

```
/* JAlcazar 26/05/2009
 * Elimina de la Lista Procesos el Proceso CORRECCIONES
 * que corresponde al Rol SIR_ROL_RESPONSABLE_CORRECCIONES
 */
if(CRol.SIR_ROL_RESPONSABLE_CORRECCIONES.equals(((Rol)
session.getAttribute(WebKeys.ROL)).getRoleId())){
int tamProcesos = procesos.size();
for (int iP = 0; iP < tamProcesos; iP++){
if ("CORRECCION".equals(((Proceso)procesos.get(iP)).getDescripcion())){
procesos.remove(iP);
}
}
}
```

2) Modificar Proceso Botón CORRECION

- \* Para los casos en que se involucra el botón CORRECCION y los ROLES AUXILIAR DE CORRECCIONES Y MESA DE CONTROL CORRECCIONES la funcionalidad fue corregida en caso Mantis 729, Requerimiento 017, Desarrollador Ing. Osbert Linero
- \* Para Realizar efectiva la modificación realizada, se agrego a la linea 143 del método procesarNavegacion de la clase ControlNavegacionSeguridad lo siguiente:  
/\* JAlcazar 20/05/2009

Agrega un atributo al request con el tamaño de la lista de procesos  
request.setAttribute(WebKeys.NUM\_PROCESOS, new Integer(((List) procesos).size()));

**047\_141\_Eliminar-botón-de-pago-impuesto-DR-I-301**

**Descripción del Requerimiento:**

Eliminar botón para cobro de impuesto

Toda vez que el aplicativo es parametrizado, en cuanto a si recaudan o no impuesto de registro, se requiere que para las ORIPS que no cobran impuesto, el formulario o pantalla de radicación no pida información al respecto.

**Describir el Impacto en el Desempeño de la Aplicación**

No es de alto impacto, por estandarización y parametrización se realiza el requerimiento.

**Análisis de Requerimiento:**

El sistema ya realiza lo que se pide en el requerimiento. Se adjunta impresión de la pantalla.

<p>The screenshot shows a Windows application window titled "SIR :: Superintendencia de Recursos y Registro :: Almacén del Información de Justicia :: SIR - Internet Internet Explorer". The window has a header with the logo of the Superintendencia de Recursos y Registro, the date "Martes 20 de Enero de 2009", and the location "República de Colombia". Below the header, there's a status bar with "Última actualización: 13/05/2009" and "Versión: 1.4.0". The main area contains several input fields and dropdown menus for creating a new appointment ("Nuevo turno"). Fields include "Número de Turno" (number 001), "Descripción" (description "GLPI 8211. CIRCULO 300-100"), "Número de Telefono" (phone number 3111111111), and "Nombre" (name "HUGO.MANTILLA"). There are also sections for "Otra Técnica" and "Datos Voluntario". At the bottom, there's a footer with "Última actualización: 13/05/2009" and "Versión: 1.4.0".</p>	
<p><b>Observaciones:</b> La prueba se realizó con el usuario HUGO.MANTILLA del circulo de Bucaramanga-300.</p>	
<p style="background-color: #ADD8E6;"><b>052_141_Turnos manuales solo se puede hacer el pago en efectivo</b></p>	
<p><b>Descripción del Requerimiento:</b> El aplicativo SIR tiene el inconveniente de que para radicar turnos manuales solo se puede hacer el pago, en efectivo porque el sistema no permite otra forma de pago, entonces se tiene conflicto con las oficinas donde solo se reciben consignaciones como en el caso de Honda. Por favor verificar en el aplicativo cual sería la opción, mejora o modificación para poder dar solución al inconveniente. ID GLPI 8211. Honda.</p>	
<p><b>Describir el Impacto en el Desempeño de la Aplicación</b> El sistema en algunos casos toma por defecto el pago en efectivo. El caso se podría hacer teniendo en cuenta cuando se registra un turno inmediato, se anexa imagen de registrar pago en el turno inmediato.</p>	
<p><b>Análisis de Requerimiento:</b> Para este caso se identifico que luego de pulsar en el botón de eliminar o agregar se redireccionaba a una pagina que no debía enviarse.</p>	
<p><b>Observaciones:</b> Se me modifico el archivo org.auriga.smart.web.xml redireccionando a la pagina que pertinente, este archivo se encuentra en el mapas.jar...</p>	
<p style="background-color: #ADD8E6;"><b>053_141_El Reporte 90 solo se genera de fechas recientes</b></p>	
<p><b>Descripción del Requerimiento:</b> El Reporte 90 solo se genera de fechas recientes.</p>	
<p>Josefina se comunica e informa que intenta generar reportes entre enero y abril, pero no se generan sale en blanco.</p>	
<p>Se generan reportes de prueba de varias fechas entre enero y abril, y todos los reportes</p>	

salen en blanco. Se hacen pruebas con varios usuarios, y por diferentes nodos. Se detecta que al verificar reportes cercanos a la fecha actual estos si salen de la forma correcta. Por favor verificar. Id GLpi 9339. Neiva.

#### **Describir el Impacto en el Desempeño de la Aplicación**

No se considera de impacto alto.

#### **Análisis de Requerimiento:**

No se estaban desplegando datos en fechas anteriores ya que el reporte estaba validando que desplegará los turnos que se encontraran en la fase de entrega siempre y cuando no hayan pasado por la fase de entregado (Finalizado).

#### **Observaciones:**

Teniendo en cuenta las observaciones de Javier Rocha el reporte desplegará los turnos que tengan la fase de entrega sin validar que el turno se halla finalizado. Se adjuntan reportes; verificar si el usuario los necesita para enviarlos; y se les aclara que el cambio será reflejado NO en el despliegue que se realizara en este mes de Junio si no en el próximo.

Los archivos adjuntos son;

- Reporte\_90\_Cucuta\_15\_01\_2009.pdf.
- Reporte\_90\_Neiva\_10\_03\_2009.pdf.
- Reporte\_90\_Neiva\_15\_01\_2009.pdf.
- Reporte\_90\_Neiva\_17\_04\_2009.pdf.
- Reporte\_90\_Bucaramanga\_15\_01\_2009.pdf.
- Reporte\_90\_Neiva\_18\_02\_2009.pdf.
- Reporte\_90\_Neiva\_26\_05\_2009.pdf.
- Reporte\_90\_Neiva\_27\_05\_2009.pdf.

#### **DESPLIEGUE 09/06/2009 – 2.0.0**

#### **INCIDENCIAS RESUELTA**

ID GLPI	OBSERVACIONES
Pantalla administración – Reportes – Listado de Reportes	
<b>Reporte 6</b>	
8303	<p><b>REQUERIMIENTO</b>  Mery indica que tiene inconveniente con el repote 6, pues la página salió cortada después de la página 999 salio la 100. Por favor verificar el reporte 6 el campo de la página no está lo suficientemente amplio para que los números salgan completos, se puede ver en el reporte adjunto que después del número 999 el sistema no está mostrando la página correctamente ya que aparece 100 y no 1000.</p> <p><b>DESARROLLO</b>  Se modifco la configuración del campo de la página en el rep. 6.</p>
8257	<p><b>REQUERIMIENTO</b>  No sale completo el numero de radicación en el en el reporte 6B.</p> <p><b>DESARROLLO</b>  Se modifco la configuración del campo de radicación en el reporte 6B.</p>
<b>Reporte 11</b>	

8337	<p><b>REQUERIMIENTO</b> Nos informa Angela que esta intentando generar el reporte 11 del día 08/04/2009 y del día 22/04/2009 y no es posible generarlo, se queda en blanco.</p> <p><b>DESARROLLO</b> El error se presentaba por validación al momento de generar el porcentaje se adjuntan los reportes del día 08/04/2009 y 22/04/2009; verificar si el usuario los necesita y se les envía y se les aclara que el cambio será reflejado el mes entrante en producción.</p>
<b>Reporte 24</b>	
6780	<p><b>REQUERIMIENTO</b> El reporte 24 mayores valores ingresados en el mes de enero/2009 repitió el mayor del turno 2009-230-6-813 varias veces, fue detectado al hacer la contabilización. Mery Tamayo.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 24.</p>
<b>Reporte 25</b>	
6870	<p><b>REQUERIMIENTO</b> El reporte 25 menciona el turno 2009-314-6-1085, pero este fue anulado el día 18/02/2009. Tiene descuadre de 14.000, solicita se revise el reporte con detalle.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 25.</p>
8809	<p><b>REQUERIMIENTO</b> El reporte 25 no coincide con lo generado hoy en la oficina.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 25 y procedimiento del reporte 91.</p>
8856	<p><b>REQUERIMIENTO</b> Informa Rodrigo que no coinciden los valores generados en los Reportes 51 y Reporte 25 del día 11/05/09 del usuario Jhon.Rojas. Favor verificar los reportes 51 y 25 usuario Jhon.Rojas.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 51 y script del reporte 25.</p>
8773	<p><b>REQUERIMIENTO</b> Se tiene inconsistencia debido a que el Reporte 51 no descontó los turnos anulados 2009-300-6-19382 y 2009-300-6-19384; referenciados en el Reporte 25.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 51 y script del reporte 25.</p>
<b>Reporte 51</b>	
7337	<p><b>REQUERIMIENTO</b> En el Reporte 51, usuario María Pedraza, en devoluciones de dinero se</p>

	<p>puede observar que refleja un valor de 434.000 y en el Reporte 25 \$441.000, esto da de la sumatoria de las devoluciones. Hay una diferencia entre el Rep 51 y el Rep 25 de \$7.000, la usuaria consulta donde están los \$7000.</p> <p>Revisando el Reporte 51 (Resumen general por conceptos) aparecen 3 turnos en devoluciones, pero si vemos el Reporte 25 encontramos 6 turnos de devoluciones.</p> <p><b>DESARROLLO</b> Se modifico el procedimiento del reporte 51 y script del reporte 51.</p>
6791	<p><b>REQUERIMIENTO</b> El Reporte 51 del día 17/02/09 es inconsistente con respecto al usuario Natalia Aristizabal, el valor total aplicado de este reporte no coincide con el valor total aplicado del tiracaja del usuario en cuestión.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 51.</p>
6794	<p><b>REQUERIMIENTO</b> El reporte 51 del día 4 del mes de febrero no ha sido posible generarse. Al ingresar los datos la pantalla queda en blanco.</p> <p><b>DESARROLLO</b> Se valido en el reporte 51 el tamaño de la solicitud para verificar si es de tipo certificado masivo – 20 ya que cuando es una solicitud de menos de 9 caracteres genera una excepción que no permite generar el reporte.</p>
6845	<p><b>REQUERIMIENTO</b> El reporte 51 del día 20 de febrero se visualiza el turno 2009-078-6-300 y este se encuentra anulado, en el tiracaja de nancy.bernal si se observa el turno anulado.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 51.</p>
6925	<p><b>REQUERIMIENTO</b> Inconsistencia con el Reporte 51. Deivi se comunica e informa que el Reporte 51, es inconsistente pues no coincide con el tiracaja del usuario Ayda Ortega Ortega pues el Valor total en este reporte es de \$721,680, mientras que en el Reporte 51 el valor es \$787,830 \$, No se esta descontando el turno anulado 2009-342-6-551 por valor de \$66,150.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 51.</p>
7435	<p><b>REQUERIMIENTO</b> Se comunica la oficina de registro para informar que el reporte 51 no está saliendo con los datos correctos, ya que al compararlo con el 25 está sumando \$28000 de más.</p> <p><b>DESARROLLO</b></p>

	<p>Se reviso y no existe inconsistencia en los reportes 25 y 51 de la fecha 15/04/2009 para la funcionaria Violeta Rojas.</p>
8365	<p><b>REQUERIMIENTO</b> El reporte 51 no elimino dos turnos los cuales fueron anulados. Se reporte la incidencia de los turnos son: 2009-342-6-1231 y 2009-342-6-1232. Se le indica al usuario Deivi que en el reporte no aparecen los turnos anulados y por tal razón el reporte tiene todos los valores de los turnos creados en el SIR actualmente. Se requiere que en los reportes no se vea reflejados los valores de los de los turnos anulados. El usuario que estaba haciendo los turnos que fueron anulados es AIDA.ORTEGA.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 51.</p>
7267	<p><b>REQUERIMIENTO</b> Verificar Datos Reporte 51 9/3/2009 ORIP BUCARAMANGA. Este reporte para el usuario HUGO.PRADA en el concepto de Cert. Radicados muestra en el valor aplicado por cheque, un valor de 7000, al revisar el reporte 25 para el mismo usuarios HUGO.PRADA no se refleja ningún valor de 7000 ni tampoco pagado en cheque. Se requiere investigar de donde se genera este valor y si es un error quitarlo del reporte para que los valores no sean errados.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 51.</p>
7555	<p><b>REQUERIMIENTO</b> Incoherencias reporte 51 y 25 ORIP Cúcuta días 5 y 6 de marzo. Según oficio de la Oficina de registro de Cúcuta, el cual manifiesta que de manera urgente se verifique y se corrija los inconvenientes presentados los días 5 y 6 de marzo del presente año, el los reportes 25 (tiracaja y 51 informe de ingresos e ingresos), ya que no se ha podido cuadrar con los cajero del banco los ingresos de los respectivos días ya que hay una diferencia entre los dos reportes.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 51 y 25.</p>
7383	<p><b>REQUERIMIENTO</b> Se anula el turno: 2009-017-6-694 pero aparece en el reporte 51 los valores cobrados en dicho turno, se pide que se elimine el dinero que aparece en este turno anulado para que no aparezca en el reporte 51.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 51.</p>
7273	<p><b>REQUERIMIENTO</b> Nos informa Yaneth Carvajal que en el Reporte 51 - Ingresos y Egresos del día 27/02/09 el sistema no resta el valor \$ 32.430 del turno anulado 2009-007-6-494.</p> <p><b>DESARROLLO</b> Se modifico el script del reporte 51.</p>

8047	<b>REQUERIMIENTO</b> Se duplica el turno 2009-372-1-3314 en le reporte 74.
	<b>DESARROLLO</b> Se modifco el script del reporte 74.
<b>Reporte 91</b>	
9029	<b>REQUERIMIENTO</b> El reporte 91 de los días 21/04/09, 28/04/09 y 06/05/09 no se visualiza el total.
	<b>DESARROLLO</b> Se modifco la configuración de los campos en el reporte 91 agregándoles la opción de Estirar con desbordamiento.
8053	<b>REQUERIMIENTO</b> El reporte 91 no muestra la sumatoria en los Actos con Cuantía y en los Totales.
	<b>DESARROLLO</b> Se modifco la configuración del campo de sumatorias en el reporte 91.
<b>Reporte 156</b>	
8943	<b>REQUERIMIENTO</b> En el reporte 156 existe un problema general con los turnos anulados ya que cuando se hace una entrega masiva de dichos turnos, en el reporte diario de anulados aparecen estos turnos teniendo en cuenta que la fecha de anulación es otra, lo que quiere decir que el sistema está tomando la fecha de entrega como la fecha de anulación.
	<b>DESARROLLO</b> Se modifco el script del reporte 156.
8459	<b>REQUERIMIENTO</b> Nos informa Yoli que en el reporte 156 - anulados del día generado el día 23/04/2009 y 27/04/2009 aparecen turnos anulados de fechas diferentes.
	<b>DESARROLLO</b> Se modifco el script del reporte 156.
8267	<b>REQUERIMIENTO</b> Reporte 156 - Diario Radicador de Anulados. Nos informa Yoli que en el reporte 156: Diario radicador de anulados generado el día de ayer aparece 4 turnos del 2008 que fueron anulados. Los siguientes turnos fueron anulados en el 2008, el día de ayer se hizo una entrega de masivos y los turnos aparecen en el diario radicador de anulados generado el día de ayer.
	<b>DESARROLLO</b> Se modifco el script del reporte 156.
<b>LIBRERIAS</b>	
LIBRERIA	OBSERVACIONES
jasperreports-3.0.0.jar	La versión de reportes jasperreports-2.0.2.jar se actualizo a la versión jasperreports-3.0.0.jar
jfreechart-1.0.3.jar	Permite graficar todo tipo de datos en diferentes tipos de vistas (en modo de barra, de pastel, lineal, etc). Esta librería

	se implemento para el reporte reporte-G06.Grafica de operación de turnos por círculo y proceso el cual no se desplegaba por falta de las librerías jfreechart y jcommon.
jcommon-1.0.0.jar	Jcommon, es una colección de clase de utilidad, usadas por JFreeChart.
org.portal.modelo.jar	Librería encargada de la integración de la aplicación VUR.

#### PROCEDIMIENTOS

SP_REPORTE_91	Actualización del procedimiento encargado del reporte 91 (Informe de ingresos del impuesto de registro) ya que estaba teniendo en cuenta valores de impuesto de turnos anteriores.
SP_REPORTE_51	Actualización del procedimiento encargado del reporte 51 (Resumen diario de ingresos y egresos) ya que no estaba desplegando la cantidad real de turnos de devolución.

#### INDICES

Se agregaron los siguientes índices que permiten un mejor tiempo de respuesta y no generen full en las tablas de que se llaman en las consultas de los reportes:

REPORTE	INDICE
Reporte 6B – Diario Radicador de Certificados del día.	IDX_SIR_OP_REPORTE_06B
Reporte 5 – Diario mayores valores y devoluciones del día.	IDX_SIR_OP_REPORTE_05

#### CAMBIOS SCRIP REPORTES

REPORTE	OBSERVACION
Reporte 6B – Diario Radicador de Certificados del día.	Costo Anterior: 2124 Costo Actual: 7
Reporte-37. Informe de documentos de antiguo sistema	Costo Anterior: 2236 Costo Actual: 1438

#### CIERRE CURSORES

CURSOR	PROCESO
[JDOGenieZonaRegistralDAO.java at line 3538, column 20] PreparedStatement pst = con.prepareStatement(sql);	Nuevo registro (Fase Liquidador Registro).
[JDOGenieZonaRegistralDAO.java at line 3347, column 13] CallableStatement cs = null;	Trasladar Folio. (Pantalla Administrativa – Folios Anotaciones y Actos - Administración de Traslados de Folios)
[JDOGenieCiudadanoDAO.java at line 1051, column 20] PreparedStatement pst = con.prepareStatement(sql);	Crear ciudadano. (Ciudadano que se agrega a una anotación dentro de un folio)
[JDOGenieFolioDAO.java at line 15524, column 16] PreparedStatement pst = connection.prepareStatement(sql);	Hacer definitivo folio. (Teniendo en cuenta que los datos temporales pasan hacer definitivos después de la fase de firma en un turno de documento o de la fase de revisar y aprobar en un turno

	de corrección.)
[JDOGenieFolioDAO.java at line 15575, column 18] PreparedStatement pst1 = connection.prepareStatement(sb);	Hacer definitivo folio. (Teniendo en cuenta que los datos temporales pasan hacer definitivos después de la fase de firma en un turno de documento o de la fase de revisar y aprobar en un turno de corrección.)
[JDOGenieFolioDAO.java at line 17339, column 20] OracleCallableStatement cst = (OracleCallableStatement) con.prepareCall("{call PCK_SIR_VALIDACIONES.SP _DELEGAR_BLOQUEO_FOLIOS (?,?,?,?,?)}");	Delegar Bloqueo Folio. (Cuando el folio se en cuenta asociado a un turno en trámite este no podrá ser trabajando por otro usuario.)
[JDOGenieFolioDAO.java at line 17448, column 23] OracleCallableStatement cst = (OracleCallableStatement) con.prepareCall("{call SP_SG(?)}");	Segregar Folio. (Se puede realizar con un turno de Documento en la Fase de calificación o por un turno de corrección en la fase de ejecutar corrección simple.)
[JDOGenieFolioDAO.java at line 18761, column 20] PreparedStatement pst = con.prepareStatement(sql);	Búsqueda de Folio Padre. ( <ul style="list-style-type: none"><li>• Pantalla Administrativa – Manejo de turnos – Ver detalles de turno</li><li>• Pantalla Administrativa – Folios anotaciones y actos – Ver detalles de Folio)</li></ul>
[JDOGenieFolioDAO.java at line 25818, column 17] PreparedStatement ps = null;	Última dirección de Folio. (Información que se registra en la apertura de un folio y que puede ser modificada por medio de un turno de corrección.)
[JDVariablesOperativasDAO.java at line 1664, column 11] PreparedStatement ps=null;	Causal de restitución. (Pantalla Administrativa – Recursos y Revocatorias Directas – Restitución de Turnos.)
[JDVariablesOperativasDAO.java at line 3824, column 20] PreparedStatement pst = con.prepareStatement(sql);	Nuevo proceso reparto. (Fase Reparto Registro)
[JDORecibosDAO.java at line 1419, column 20] PreparedStatement pst = con.prepareStatement(sql);	Almacenar motivo incremento.
[JDReportesJasperDAO.java at line 337, column 11] PreparedStatement pst;	Generar reporte (cualquiera).
[JDReportesJasperDAO.java at line 338, column 11] CallableStatement cst;	Generar reporte (cualquiera).
[JDOTrasladoTurnosDAO.java at line 14, column 25] import java.sql.PreparedStatement;	Traslado de Turnos. (Pantalla Administrativa – Manejo de turnos – Traslados de Turnos.)
[JDOSolicitudesDAO.java at line 5018, column 20] PreparedStatement pst = con.prepareStatement(sql);	Solicitud. (Fase de Liquidador Registro.)

[OracleProcessor.java at line 630, column 17] CallableStatement call = null;	Crear, avanzar, anular o habilitar turno.
[OracleWorkflow.java at line 340, column 17] CallableStatement call = null;	Crear, avanzar turno.
[OracleFenrirJDO.java at line 983, column 20] PreparedStatement pst = con.prepareStatement(sql);	Login (Pantalla logeo de los usuarios).
[OracleFenrirJDO.java at line 1132, column 11] PreparedStatement ps = null;	Menú (Menú que se despliega al logearse un usuario)
[OracleFenrirJDO.java at line 1422, column 11] PreparedStatement ps = null;	Consulta de usuarios por círculo y rol. (Pantalla Administrativa – Usuarios – Administración de Perfiles.)
[OracleFenrirJDO.java at line 1688, column 12] PreparedStatement ps = conn.prepareStatement (consulta);	Crear nuevo usuario. (Pantalla Administrativa – Usuarios – Administración de Usuarios.)
[OracleFenrirDAO.java at line 126, column 12] PreparedStatement prepared = conn.prepareStatement (estaLogeado);	Proceso auditoria.
[OracleFenrirDAO.java at line 127, column 12] PreparedStatement prepared2 = conn.prepareStatement (insertarAuditoria);	Proceso auditoria.
[OracleFenrirDAO.java at line 128, column 12] PreparedStatement prepared3 = conn.prepareStatement (getID);	Proceso auditoria.
[OracleFenrirDAO.java at line 130, column 12] PreparedStatement prepared4 = conn.prepareStatement (avanzarSecuencia);	Proceso auditoria.
[OracleFenrirDAO.java at line 131, column 12] PreparedStatement prepared5 = conn.prepareStatement (conseguirSecuencia);	Proceso auditoria.
[OracleFenrirDAO.java at line 218, column 12] PreparedStatement prepared2 = conn.prepareStatement (consultaUsuario);	Crear nuevo usuario, asignación de rol. (Pantalla Administrativa – Usuarios – Administración de Usuarios.)
[JDOGenieImpresionDAO.java at line 980, column 12] PreparedStatement pstSel = con.prepareStatement(sqlSel);	Impresiones fallidas.
[JDOGenieImpresionDAO.java at line 985, column 13] PreparedStatement pstUpd = con.prepareStatement(sqlUpd);	Impresiones fallidas.
[JDOGenieImpresionDAO.java at line 991, column 13] PreparedStatement pstIns = con.prepareStatement(sqlIns);	Impresiones fallidas.
[JDOGenieImpresionDAO.java at line 1293, column	Proceso de impresión.

20]                   PreparedStatement pst = con.prepareStatement(sql);	
[JDOGenieImpresionDAO.java at line 1423, column 11]       PreparedStatement pst;	Consulta de reimpresiones. (Pantalla Administrativa - Impresión)