

PROYECTO SIMULADOR

Especificación de Caso de Uso

- GENERAR SIMULACIÓN DUMMIES -

Versión 1.0

Marzo de 2017

Historia de Revisiones

Fecha	Versión	Descripción	Autor
07/Mar/2017	1.0	Creación del documento	Diego Toscano

1 RESUMEN EJECUTIVO

1.1 Resumen

000	Característica	Descripción
001	Código	CUS0070
003	Nombre	Generar Simulación de Dummies
	Paquete	
004	Versión	1.0
010	Descripción	El ResponsableDeLaSimulacion sube un archivo de mensajes anexos personalizado y el sistema realiza una simulación automática por cada mensaje le otorga una cuenta aleatoria, con el fin que el usuario visualice los mensajes subidos por él mismo.
007	Autor	Diego Toscano
	Frecuencia	
	Concurrencia	SI
011	Complejidad	ALTA
	Prioridad	1
	Importancia	ALTA
	Duración media:	
	Observaciones	

Temas Pendientes				
Responsable	Descripción	Fecha de inicio	Fecha de fin	Destinatario

Responsable de Revisión Inicial	Fecha	Firma
Diego Toscano	07/Mar/2017	

2 ESPECIFICACIÓN DEL CASO DE USO

2.1 Actores

020	Nombre	Abst.	Cant.	Descripción breve
	ResponsableDeLaSimulacion	No		Usuario logueado con privilegios para realizar la simulación de dummies.

021	Precondición	Comentario
	{SimulacionPrevia} Debe existir una Simulación generada por cada formato (corpo y masivo) para que el sistema pueda tomar como muestra.	

022	Invariantes	Comentario

2.2 Curso normal

025	CUS001		Referencias
	1	El ResponsableDeLaSimulacion accede al módulo de Generación de Simulación de Dummies.	
	2	El sistema presenta 2 secciones independientes.	
	2.1	<p>En la sección 1 el sistema presenta:</p> <ul style="list-style-type: none"> a) una lista desplegable {cboFormato} con los formatos de factura (corpo o masivo). b) una lista desplegable {cboTipo} con el tipo de simulación (grupo). c) una lista desplegable {cboTipoDummy} con el tipo de simulación dummy que serían las secciones: mensajes importantes, mensajes dorso, mensajes de plan y mensajes de pack. d) Campo Nro de Lote {txtLote} e) Campo nro de cuenta {txtCuenta} f) Campo cantidad de líneas {txtLineas} g) Campo cantidad de mensajes por línea {txtMjes} (deshabilitado, en caso de selección de Packs se habilita) h) Un botón Subir archivos {btnUpload} 	{cdd001} {cdd002}

2.2	<p>En la sección 2 el sistema presenta:</p> <ol style="list-style-type: none"> Un campo descripción de la nueva simulación {txtDescripcionSimulacion} un botón Genera Simulación {btnSimulación} una grilla {grillaSimulacion} vacía con las siguientes columnas: <ul style="list-style-type: none"> Lote (nuevo) Descripción Producto (Masivo Grupo) Fecha (de nueva simulación realizada) Estado de la simulación PDF1 (link a archivo pdf) PDF2 <p>Los componentes {grillaSimulacion}, {btnSimulación} se encuentran deshabilitados.</p>	
3	El ResponsableDeLaSimulacion selecciona de la lista desplegable {cboFormato} el valor "Masivo".	
3.1	Si [el ResponsableDeLaSimulacion selecciona de la lista desplegable {cboFormato} el valor "corpo"].	
4	El sistema consulta si en el espacio de trabajo del ResponsableDeLaSimulacion se encuentran datos de edición de domicilio cargados previamente y que aún no han ejecutado la simulación. {grillaEditados}	{cdd003}
5	Si [el sistema determina que existen registros para el ResponsableDeLaSimulacion y para la simulación previa] {Sf004}	{Sf004} {ref004}
6	EL ResponsableDeLaSimulacion presiona el botón Comenzar {btnComenzar} y se habilitan todos los componentes deshabilitados.	
7	El sistema carga el Navegador de Cuentas {nvgCuentas}, con las opciones de Avanzar, Retroceder y se visualiza la cuenta actual. Por defecto aparece seleccionada la primera cuenta de la simulación.	{cdd004}
8	El sistema carga una lista desplegable {cboLinea} con todas las líneas que contiene la cuenta seleccionada. Por defecto aparece una sola línea seleccionada y este campo se encontrará <u>inactivo</u> .	{cdd005} {ref002}
9	SI [el ResponsableDeLaSimulacion presiona el Avanzar o Retroceder del navegador de cuentas nvgCuentas] {Sf002}	{sf002}
10	<p>El sistema carga el formulario {formDomicilio} con los campos de domicilio completos correspondiente a la cuenta seleccionada:</p> <ul style="list-style-type: none"> Calle Número (de calle) Piso Depto Código postal Barrio (también es usado para las entrecalles) Ciudad 	{cdd006} {ref003}
11	El ResponsableDeLaSimulacion cambia el texto en uno o varios campos del formulario {formDomicilio}	
12	El ResponsableDeLaSimulacion completa el texto {txtComentario}	
13	El ResponsableDeLaSimulacion presiona el botón { btnAcumular}	{cdd008}

14	El sistema carga la grilla {grillaEditados} con los datos editados <ul style="list-style-type: none"> Cuenta Línea (la única línea por defecto de la cuenta que no tiene relevancia) Comentario 	
15	SI [el ResponsableDeLaSimulacion presiona el Avanzar o Retroceder del navegador de cuentas nvgCuentas] {Sf002}	{Sf002}
16	SI [el ResponsableDeLaSimulacion presiona el botón borrar {btnBorrar} y Hay un registro seleccionado en la grilla {grillaEditados}] {Sf005}	
17	SI [el ResponsableDeLaSimulacion presiona el botón Salir {btnSalir}] el sistema retorna a la buscador se simulaciones.	
18	SI [el ResponsableDeLaSimulacion ingresa una descripción de la nueva simulación {txtDescripcionSimulacion} y presiona el botón Simular {btnSimulación}] {Sf001}	{Sf001} {ref001} {Ex001}

024	Poscondición	Comentario

Subflujos

025	Sf001	Creación de una nueva simulación	
	N°	Paso	Referencias
		El sistema verifica que se haya ingresado la descripción de la simulación. Caso negativo, emite un mensaje de error descriptivo y no permite continuar con el flujo.	
		El sistema genera los archivos de datos para la simulación	{cdd010}
		El sistema busca de la simulación los archivos de anexos que fueron utilizados en la simulación original	
		El sistema transfiere al directorio de entrada de la simulación los archivos de datos generados para la simulación actual más los archivos de anexos utilizados en simulación previa.	
		El sistema invoca el motor de generación de PDF con los archivos de datos más anexos.	
		El sistema presenta un mensaje indicando que la operación fue exitosa.	
		El sistema retorna a la pantalla de búsqueda de simulaciones	
		{ref001}	{ref001}

025	Sf002	Presionar el Botón Avanzar o Retroceder	
	N°	Paso	Referencias
		El sistema cambia la cuenta seleccionada del navegador de cuentas {nvgCuentas}	
		El sistema vacía lista desplegable {cboLinea}	
		El sistema vacía los campos del formulario de domicilio {formDomicilio}	
		El sistema vacía la caja de ingreso de texto {txtComentario}.	
		{ref002}	{ref002}

025	Sf004	Determinar si que existen registros para el ResponsableDeLaSimulacion	
	N°	Paso	Referencias
		El sistema carga los campos del formulario {formDomicilio} de domicilio con los datos recuperados.	{cdd007}
		{ref004}	{ref004}

025	Sf005	Borrar un registro de la grilla editados	
	N°	Paso	Referencias

	El sistema elimina el registro de la base de datos correspondiente.	{cdd009}
	El sistema elimina la fila seleccionada de la grilla {grillaEditados}	
	El sistema se posiciona en el primer registro de la grilla {grillaEditados}	
	{ref004}	{ref004}

2.3 Cursos excepcionales

Excepciones

026	Ex001	La simulación no se realizó	
	N°	Paso	Referencias
		El sistema presenta un mensaje indicando los errores que se generaron en el proceso de simulación	
		{ref001}	{{ref001}}

3 CONSIDERACIONES DE DISEÑO DATOS

027	Cdd	Componente	Acción
	{cdd001}	{grillaSimulacionOriginal}	<p>Levantar los datos de la simulación previa sobre la cual se realizará la edición de secciones. Utilizar el siguiente query:</p> <pre>SELECT ID_LOTE AS Padre, LF.DESCRIPCION AS FORMATO, LC.DESCRIPCION AS DESCRIPCION, LC.FECHA_HORA_CREACION AS FECHA, LC.CANTIDAD_CUENTAS AS CTAS, USUARIO FROM LOTE_CABECERA LC INNER JOIN TIPO_FORMATO LF ON LC.ID_TIPO_FORMATO = LF.ID_TIPO_FORMATO WHERE LC.ID_LOTE = :ID_LOTE_CABECERA</pre>
	{cdd002}	{cboSecciones}	<p>Levantar los datos de la tabla dbo.SECCION {ID_SECCION, DESCRIPCION} en el combo se debe mostrar la descripción y para esta pantalla filtrar por el ID_SECCION= 3. Utilizar el siguiente query:</p> <pre>select ID_SECCION, DESCRIPCION from dbo.SECCION where ID_SECCION =3</pre>
	{cdd003}	{grillaEditados}	<p>Determinar si existen en el workspace del usuario datos previos para la simulación de origen. Utilizar el siguiente query</p> <pre>select * from dbo.WS_CABECERA where ID_LOTE_CABECERA_FK = :ID_LOTE_CABECERA AND USERNAME = :USERNAME AND ID_SECCION = :ID_SECCION AND ESTADO = 'I'</pre> <p>I - iniciado S - Simulado</p>
	{cdd004}	{nvgCuentas}	<p>Levantar todas las cuentas que participaron en una simulación. Utilizar el siguiente Query:</p> <pre>SELECT LD.CUENTA FROM LOTE_DETALLE LD WHERE EXISTS (SELECT 1 FROM LOTE_REPOSITORIO LR WHERE LD.ID_LOTE_REPOSITORIO = LR.ID_LOTE_REPOSITORIO AND LR.ID_LOTE_CABECERA = :ID_LOTE_CABECERA)</pre>

{cdd005}	{cboLinea}	<p>Levantar todas las líneas que participaron en una cuenta-simulación. Utilizar el siguiente Query:</p> <pre>select LINEA from [dbo].[FN_LINEAS_X_CUENTA_X_LOTE] (:ID_LOTE_CABECERA, :CUENTA)</pre>																																								
{cdd006}	{formDomicilio}	<p>Levantar la sección domicilio de la cuenta seleccionada. Utilizar el siguiente Query:</p> <pre>SELECT FN.REGISTRO , SUBSTRING(FN.TXT, Q.INICIO + 8, Q.LONGITUD) AS TXT , Q.CAMPO, Q.INICIO, Q.LONGITUD FROM [dbo].[FN_REGISTROS_X_CUENTA_X_LOTE] (:ID_LOTE_CABECERA, :CUENTA) FN INNER JOIN (SELECT B.REGISTRO , [CAMPO] , [INICIO] , [LONGITUD] FROM [dbo].[DET_CONFIG_TIPO_ARCHIVO] C INNER JOIN [CONFIG_TIPO_ARCHIVO] B ON B.[ID_CONFIG_TIPO_ARCHIVO] = C.[ID_CONFIG_TIPO_ARCHIVO] INNER JOIN [TIPO_ARCHIVO] A ON A.[ID_TIPO_ARCHIVO] = B.[ID_TIPO_ARCHIVO] WHERE B.ID_TIPO_ARCHIVO = 7 /* 7 masivo u 8 corpo */and B.ID_SECCION = 3 /* domicilio */) Q ON Q.REGISTRO = FN.REGISTRO WHERE FN.ID_SECCION = 3 /* domicilio */ ORDER BY FN.REGISTRO, Q.INICIO</pre> <p>El resultado sería algo así:</p> <table><tr><th>REGISTRO</th><th>TXT</th><th>CAMPO</th><th>INICIO</th><th>LONGITUD</th></tr><tr><td>RES011D</td><td>SARMIENTO</td><td>calle</td><td>1</td><td>61</td></tr><tr><td>RES011D</td><td>2606</td><td>nro</td><td>62</td><td>5</td></tr><tr><td>RES011D</td><td>Piso PB</td><td>piso</td><td>67</td><td>11</td></tr><tr><td>RES011D</td><td></td><td>depto</td><td>78</td><td>11</td></tr><tr><td>RES011D</td><td>./1154317531</td><td>barrio</td><td>89</td><td>43</td></tr><tr><td>RES012D</td><td>B1678HBD</td><td>cod_postal</td><td>1</td><td>9</td></tr><tr><td>RES012D</td><td>CASEROS</td><td>ciudad</td><td>10</td><td>37</td></tr></table> <p>Los campos del formulario deberían llenarse con el nombre de la columna “campo” y el valor de la columna “txt”. Los demás campos deberían no ser visibles al usuario, pero sí serán útiles las columnas “registro” y “longitud” para posteriormente escribir el archivo Dat, ya que la parte de domicilio del dat se compone de la sigte manera:</p> <pre>RES011D SARMIENTO 2606 Piso PB ./1154317531 RES012D B1678HBD CASEROS</pre> <p>1ra línea: Código RES011D (7 caracteres) + espacio (1 carácter) + calle (61 caract. que ya lo informa la columna “longitud”) + nro (5) + piso(11) + depto (11) + barrio (43)</p> <p>2da línea: Código RES012D (7 caracteres) + espacio (1) + cód. postal (9) + ciudad (37)</p>	REGISTRO	TXT	CAMPO	INICIO	LONGITUD	RES011D	SARMIENTO	calle	1	61	RES011D	2606	nro	62	5	RES011D	Piso PB	piso	67	11	RES011D		depto	78	11	RES011D	./1154317531	barrio	89	43	RES012D	B1678HBD	cod_postal	1	9	RES012D	CASEROS	ciudad	10	37
REGISTRO	TXT	CAMPO	INICIO	LONGITUD																																						
RES011D	SARMIENTO	calle	1	61																																						
RES011D	2606	nro	62	5																																						
RES011D	Piso PB	piso	67	11																																						
RES011D		depto	78	11																																						
RES011D	./1154317531	barrio	89	43																																						
RES012D	B1678HBD	cod_postal	1	9																																						
RES012D	CASEROS	ciudad	10	37																																						
{cdd007}	{grillaEditados}	<p>Realizar un join entre WS_CABECERA Y WS_DETALLE y cargar los datos para la visualización de la Grilla. Utilizar el siguiente query:</p> <pre>select * from dbo.WS_CABECERA WSC inner join dbo.WS_DETALLE WSD ON WSD.ID_WS_CABECERA = WSC.ID_WS_CABECERA WHERE ID_LOTE_CABECERA_FK = :ID_LOTE_CABECERA AND USERNAME = :USERNAME AND ID_SECCION = :SECCION</pre>																																								

{cdd008}	{btnAcumular}	<p>Los datos se deberán registrar en las tablas</p> <ul style="list-style-type: none"> • <code>dbo.WS_CABECERA</code> • <code>dbo.WS_DETALLE</code> <p>agregar el registro en la tabla <code>dbo.WS_CABECERA</code> Campos a completar</p> <ul style="list-style-type: none"> • [USERNAME] • [ID_TIPO_FORMATO] • [ID_TIPO_CONF] • [ID_LOTE_CABECERA_FK] • [ID_LOTE_REPOSITORIO_FK] • [ID_SECCION] • [CUENTA] • [LINEA] • [COMENTARIO] • [FECHA_HORA_CREACION]->SE GUARDA SOLO • ESTADO = 'I' -> SE GUARDA SOLO <p>Para la cabecera aplicar un distinct sobre los registros por (ID_TIPO_FORMATO, ID_TIPO_CONF, ID_LOTE_CABECERA_FK, ID_LOTE_REPOSITORIO_FK, ID_SECCION, CUENTA, LINEA)</p> <pre>INSERT INTO DBO.WS_CABECERA (USERNAME, ID_TIPO_FORMATO, ID_TIPO_CONF, ID_LOTE_CABECERA_FK, ID_LOTE_REPOSITORIO_FK, ID_SECCION, CUENTA, LINEA, COMENTARIO) VALUES(:USERNAME, : ID_TIPO_FORMATO, :ID_TIPO_CONF, :ID_LOTE_CABECERA_FK, :ID_LOTE_REPOSITORIO_FK, :ID_SECCION, :CUENTA, :LINEA, :COMENTARIO)</pre> <p>agregar los registros en la tabla <code>dbo.WS_DETALLE</code> Campos a completar:</p> <ul style="list-style-type: none"> • [ID_WS_CABECERA] • [REGLON] • [TXT] <p>Es decir, toma todo el {formDomicilio} e inserta en la tabla WS_DETALLE 2 registros en este caso (RES011D y RES012D).</p> <p>Ejemplo: RES011D SARMIENTO 2606 Piso PB ./1154317531 RES012D B1678HBD CASEROS</p> <p>1ra línea: Código RES011D (7 caracteres) + espacio (1 carácter) + calle (61 caract. que ya lo informa la columna "longitud") + nro (5) + piso(11) + depto (11) + barrio (43)</p> <p>2da línea: Código RES012D (7 caracteres) + espacio (1) + cód. postal (9) + ciudad (37)</p> <p>Insertar por cada línea/renglón: INSERT INTO [dbo].[WS_DETALLE] ([ID_WS_CABECERA],[REGLON],[REGISTRO],[TXT]) VALUES(:ID_WS_CABECERA, :REGLON, :REGISTRO, :TXT)</p> <p><u>Nota:</u> la cantidad de caracteres por cada campo se podría guardar en campos "hidden" en el {formDomicilio}, ya que en query de la sección {cdd006} ya lo trae.</p>
{cdd009}	{grillaEditados}	<p>Los datos se deberán eliminarse de las tablas</p> <ul style="list-style-type: none"> • <code>dbo.WS_CABECERA</code> • <code>dbo.WS_DETALLE</code> <p>Se eliminan los registros de la tabla <code>dbo.WS_DETALLE</code> Se eliminan los registros de la tabla <code>dbo.WS_CABECERA</code></p>

{cdd010}	{btnSimular}	<p>El proceso de generación de archivos DAT deberá ser el siguiente:</p> <p>Obtener los distintos ID_LOTE_REPOSITORIO de la tabla WB_CABECERA para la simulación orinal y usuario.</p> <pre>SELECT DISTINCT ID_LOTE_CABECERA_FK, ID_LOTE_REPOSITORIO_FK FROM WS_CABECERA C WHERE ID_LOTE_CABECERA_FK = :ID_LOTE_CABECERA AND username= :USERMANE</pre> <p><i>Por cada ID_LOTE_CABECERA_FK, ID_LOTE_REPOSITORIO_FK, Recuperado ciclar</i></p> <pre>DO WHILE { Obtener los registros REFERENCIA_ARCHIVO , REG_CABECERA, REG_DETALLE , y para el ID_LOTE_REPOSITORIO_FK, ID_LOTE_CABECERA_FK Del cliclo actual.</pre> <pre>select REFERENCIA_ARCHIVO, CONVERT(VARCHAR(MAX),REG_CABECERA) AS REG_CABECERA, CONVERT(VARCHAR(MAX),REG_DETALLE) AS REG_DETALLE from LOTE_REPOSITORIO WHERE ID_LOTE_CABECERA = :ID_LOTE_CABECERA_FK AND ID_LOTE_REPOSITORIO = :ID_LOTE_REPOSITORIO_FK</pre> <p>GRABAR UN ARCHIVO que se llame como la “REFERENCIA_ARCHIVO” y la primera línea debe ser el REG_CABECERA</p> <p>Recuperar todo el contenido para Grabar en el Archivo</p> <pre>SELECT * FROM [dbo].[FN_ARCHIVO_X_LOTE_REPOSITORIO_USUARIO] (:ID_LOTE_CABECERA_FK, :ID_LOTE_REPOSITORIO_FK, :USERMANE)</pre> <p>DO WHILE {Grabar en el archivo el campo TXT línea a línea.}</p> <p>Grabar en el archivo el campo REG_DETALLE</p> <p>Cerrar el archivo y transferirlo al input</p> <pre>} FIN DEL PRIMER CICLO</pre> <p>Enviar a Simular</p> <p>Si la simulación fue correcta entonces Actualizar el estado = 'S' del campo estado de la tabla WS_CABECERA</p>
----------	--------------	--

4 REGLAS DE NEGOCIO

028	Rn	Nombre	Descripción	Valor actual

5 GLOSARIO

029	Término	Tipo	Descripción

6 OBSERVACIONES

030	Obs	Nombre	Descripción

7 INTERFACES ASOCIADAS

031	Interfaz	Descripción	Referencia
	Int001	Generador de Dummies	

Mejora Grupo de Control

Formato: Tipo: Tipo Dummy:

Claro Masivo Grupo Mensajes Imp. Dorso Plan Pack

Subir archivos

☐ Lote ☐ Cuenta

Cant. Lineas = lineas (masivo) <= lineas (corpo)

Cant. Mjes x Linea en caso de PACK

Descripción

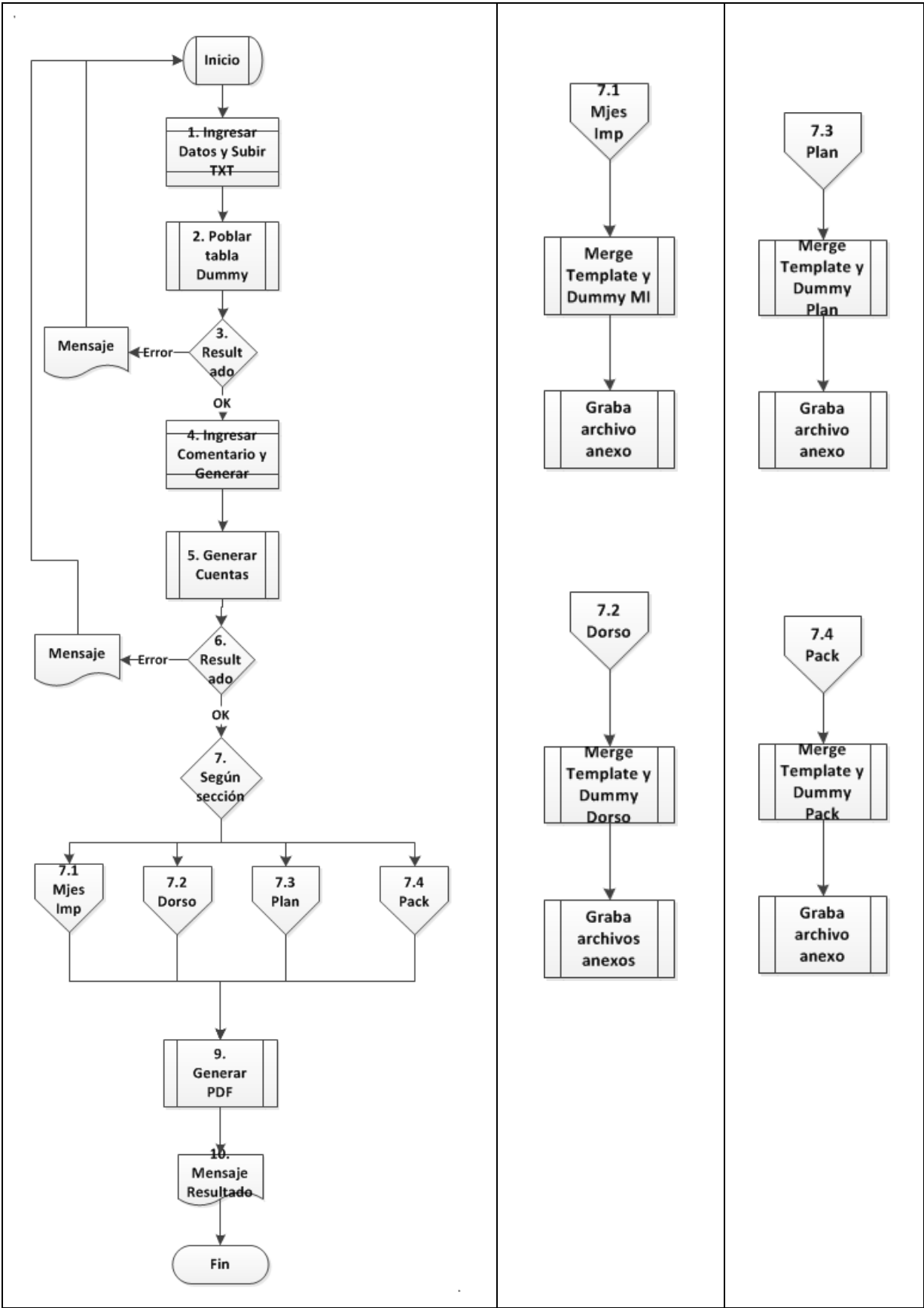
Generar Simulación

Lote	Descripción	Producto	Fecha	Estado	PDF1	PDF2
102030	Dummy Nuevo Dorso	Masivo Grupo	10/02/2017	Simulacion Exitosa	Ver	Ver

8 ARCHIVOS ASOCIADOS

032	Archivo	Descripción	Referencia

9 DIAGRAMAS



0.- Inicio del proceso

1.- Usuario ingresa datos: formato, tipo, tipo dummy (*mensajes imp, dorso, planes, packs*), lote y cuenta ó cantidad líneas (*igual cantidad para masivo, menor-igual para corpo*), cant mensajes x línea (*en caso de mensajes x pack*).

Usuario sube el archivo txt (*mensajes o dorso o infoaumentos (planes) o infopack (packs)*).

2.- Sistema borra tabla dummy correspondiente al tipo de dummy por usuario y agrega registros del archivo subido y por usuario. Luego borra el archivo txt.

*Delete dummies_xx where username = @user ,
y luego Insert into dummies_xx values (@user, ...)*

3.- Sistema verifica que todo se haya subido correctamente, sino emite mensaje de error y vuelve al comienzo.

4.- Usuario ingresa comentario y genera simulación.

5.- Sistema genera cuentas según opciones ingresadas por: lote/nrocuenta ó cantidad de líneas (si es corpo cantLineas <= @lineas), haciendo un random.

```
DECLARE @id_lote INT; DECLARE @cuenta VARCHAR(12)
SELECT @id_lote =id_lote, @cuenta=cuenta
FROM [dbo].[FN_RANDOM_LOTE_CUENTA] (@formato,@cantLineas)
SELECT * FROM [FN_LINEAS_X_CUENTA_X_LOTE](@id_lote,@cuenta)
SELECT @id_lote, @cuenta
```

6.- Si no encuentra resultados, emite error y vuelve al comienzo. Retorna: Id_lote = 0 y Cuenta = vacío.
Ejemplo:

```
select * FROM [dbo].[FN_RANDOM_LOTE_CUENTA] (7,200)
```

7. Según sección seleccionada al principio por el usuario:

7.1.- Mensajes Importantes:

- Sistema hace un merge entre tablas de: template y dummy.

```
exec [dbo].[SP_DUMMY_TEMPLATE_MENSAJESIMPORTANTES] @username
```

- Sistema graba resultado en archivo de anexo → *mensajes.dat*

- Sistema arma el archivo de datos de clientes (dat) a través del sigte query y luego lo graba creando el archivo gp_masivo.dat o gp_corpo.dat, según corresponda:

```
EXEC [dbo].[SP_DUMMY_MENSAJESIMPORTANTES] @ID_LOTE, @CUENTA, @ID_TIPO_FORMATO, @USERNAME
```

7.2.- Mensajes Dorso:

- Sistema hace un merge entre tablas de: templates y dummy de mensajes.

```
exec [dbo].[SP_DUMMY_TEMPLATE_MENSAJESIMPORTANTES] @username
exec [dbo].[SP_DUMMY_TEMPLATE_PLANES] @ID_LOTE, @CUENTA, @ID_TIPO_FORMATO, @cant_lineas, @USERNAME
exec [dbo].[SP_DUMMY_TEMPLATE_DORSO] @ID_LOTE, @CUENTA, @ID_TIPO_FORMATO, @cant_lineas, @USERNAME
exec [dbo].[SP_DUMMY_DORSO] @ID_LOTE, @CUENTA, @ID_TIPO_FORMATO, @cant_lineas, @USERNAME
```

- Sistema graba resultado de todas las consultas en archivos de anexos y el de datos (en total 4 archivos):

- *mensajes.dat* → 1er query anterior
- *gpplanes.dat* → 2do query
- *gpgrilla_dorso.dat* → 3er query
- *gp_masivo1.dat* → 4to query

7.3.- Mensajes Planes:

7.4.- Mensajes Packs:

9.- Sistema utiliza el mismo proceso de generación de archivo PDF con el archivo de datos y el anexo generados anteriormente.

10.- Sistema emite el mensaje de resultado del proceso.

11.- Fin del proceso.