

Información técnica del proyecto

Nombre del proyecto: Dictámenes

Tecnología utilizada: .NET Framework 4.5

Arquitectura utilizada: MVC

Motor de base de datos: SQL Server con Entity Framework y Linq

Relevamiento

Captura de relevamiento inicial de la información a ingresar en Excel:

STyC
RELEVAMIENTO DE INFORMACIÓN

Gerencia: JURÍDICA
Fecha: jul-21
Proyecto: REGISTRO DE DICTÁMENES

NRO CAMPO	NOMBRE DE CAMPO	CONCEPTO (BREVE EXPLICACIÓN DE QUÉ ES)	REQUERIDO U OPCIONAL	TIPO	LONGITUD	FORMATO	VALIDACIÓN DEL CAMPO (SI VALIDA CONTRA TABLA INDICAR LA TABLA)	VALIDACIÓN CON OTRO CAMPO	VALIDACIÓN CON OTRO SISTEMA/BASE/FORMULARIO	COMENTARIO / OBSERVACIONES
1	nroGDE o nroDictamen		Requerido	carácter	30					
2	fechaCreacion		Requerido	Fecha	8	AAAA/MM/DD				
3	usuarioGenerador		Requerido				tabla de referencias			
4	idReferencia o idAsunto									
5	referenciaDetalle o asunto									
6	tipoDictamen						DISCU o FOLIO			
7	tipoSujetoObligado						tabla aseguradora, reas, productor, denunciante, etc			
8	numeroObligadoNombre									
9	idSujetoObligado	cuit cui matricula dependiendo del tipoSujetoObligado								
10	nroExpediente	es parte del nroGDE								
11	asunto 1									
	usuario	usuario que ingresa la info								
	fechaCarga	fecha en que ingresa la info								

el registro de Dictámenes debería contener como criterios de búsqueda:

1_ por VOZ (sería bueno que la búsqueda se hiciera extensiva al cuerpo del dictámen, no sólo a la referencia)
2_ por SUJETO OBLIGADO a través del Nº de CUIT/CUIL (aseguradoras/ reaseguradoras/ productor/ denunciante etc) .
3_ por ASUNTO 1 en este caso podemos armar una especie de nomenclador con los " posibles asuntos" ej: aumento de capital; fusiones, multa, etc..., como criterio adicional podíamos agregar si corresponde a una Resolución Particular o Resolución General)
4_ Por NUMERO DE DICTAMEN
5_ Por NUMERO DE EXPEDIENTE

Sería importante que los criterios además puedan acumularse, es decir, si conozco numero de dictamen y numero de expediente, poder utilizar ambos conceptos y así acotar el resultado en función a las necesidades.

Fecha creación	Usuario gen	Referencia	REFERENCIA
04/01/2021 15:39	RNAYAR	Ex-2020-35453931 GERENCIAMIENTO MENSUAL MAYO 2020 por juicios	GERENCIAMIENTO
04/01/2021 15:39	RNAYAR	Ex-2020-58100485 GERENCIAMIENTO MENSUAL AGOSTO 2020 por pleitos	GERENCIAMIENTO
04/01/2021 15:40	RNAYAR	EX-2020-81493306 Multiples juicios individuales	JUICIOS INDIVIDUALES
04/01/2021 15:40	RNAYAR	EX-2019-06709111 GERENCIAMIENTO MENSUAL ENERO 2019	GERENCIAMIENTO
04/01/2021 15:40	RNAYAR	EX-2019-91134258 - APN-GA#SSN Registro de Liquidadores de Siniestros y Averías – Alta Anexo Diciembre 2020	ALTA/INSCRIPCION REGISTRO LIQUIDADORES DE SINIESTROS Y AVERIAS

Copia de intercambio de emails en el relevamiento de la funcionalidad de la aplicación:

Cliente:

el registro de Dictámenes debería contener como criterios de búsqueda:

1_ por VOZ (sería bueno que la búsqueda se hiciera extensiva al cuerpo del dictámen, no sólo a la referencia)

2- por SUJETO OBLIGADO a través del Nº de CUIT/CUIL (aseguradoras/ reaseguradoras/ productor/ denunciante etc) .

3- por ASUNTO (en este caso podemos armar una especie de nomenclador con los " posibles asuntos" ej; aumento de capital; fusiones, multa, etc..., como criterio adicional podíamos agregar si corresponde a una Resolución Particular o Resolución General)

4 - Por NUMERO DE DICTAMEN

5_ Por NUMERO DE EXPEDIENTE

Sería importante que los criterios además puedan acumularse, es decir, si conozco numero de dictamen y numero de expediente, poder utilizar ambos conceptos y así acotar el resultado en función a las necesidades.


Ciente:

Ale

Al ingresar en la planilla, y hacer click en el hipervínculo de la celda que contiene el nro IF, se abre el Dictamen tal como adjunto el la captura de pantalla.

Fijate que el nro IF es el primero de la planilla que te adjunté antes, y yo tomo el ese nro IF, como el número de cada dictamen para que no haya doble numeración (o sea que no haya una numeración gde y una interna de Jurídicos que sólo serviría para saber qué cantidad de dictámenes se firman, dado que el número que identifica el dictamen es el IF que otorga el GDE)

mayor número de firma que presenta problemas.



República Argentina - Poder Ejecutivo Nacional
2020 - Año del General Manuel Belgrano

Dictamen Jurídico

Número: IF-2021-00462538-APN-GAJ#SSN

Referencia: Ex-2020-35453931 GERENCIAMIENTO MENSUAL MAYO

A UNIDAD SUPERINTENDENTE

I.- Objeto del Dictamen

Vienen las presentes actuaciones a consideración de esta Gerencia de respecto de un proyecto de resolución elaborado por la Gerencia Administrativa para aprobar la transferencia de fondos a PREVENCIÓN ASEGURADORA en su calidad de Gerenciadora del Fondo de Reserva (Art. 34 Ley 24.557) por Cuatrocientos Ochenta y Tres Mil Cuatrocientos (\$2.483.400), en concepto de pago mensual del período Mayo de 2020, con cargo al Fondo de Reserva.

Ciente:

Alejandra

Cumplo con lo indicado por Marina y te envío mi modesto Registro de Dictámenes que es un Excel ubicado en

\\svr_archivos\juridica\Registro de Dictámenes Gerencia de Asuntos Jurídicos

Lo que hago es

ingresar al GDE,

filtro por fecha todo lo firmado en la Gerencia,

le agrego que filtre los DICJU y FOJDT (que son las dos formas de crear Dictamen Jurídico).

Luego elijo la opción del GDE para que pase todo lo encontrado a Excel.

De esa planilla selecciono las columnas de nro de if, fecha, firmante, referencia y pego todo en mi planilla Excel (hay una planilla de dictámenes por cada año).

Configuro previamente que al descargar cada dictamen, lo guarde en la carpeta pdf creada.

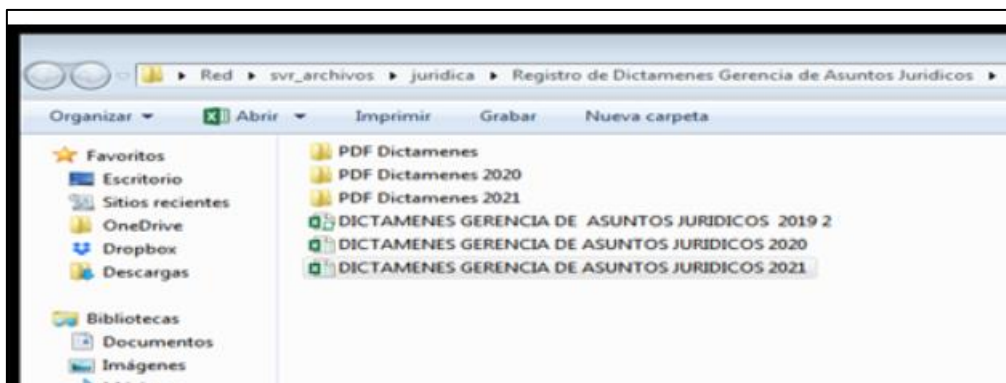
Luego, descargo cada dictamen en la carpeta de PDF seleccionada (una carpeta por año) y le hago un hipervínculo a cada dictamen pdf en la celda de Excel correspondiente.

También le agrego una referencia corta del tema del dictamen en la última columna que elijo de una lista desplegable que creé en una segunda hoja de la planilla.

El nro de IF que se genera por cada dictamen, yo lo tomo como NRO DE DICTAMEN para no generar una nueva numeración interna de cada dictamen.

Te adjunto una captura de pantalla de cómo se ve guardado en el servidor y cómo se ve una planilla.

Espero te sirva de referencia mi modesto y básico registro.



Nro. GDE	Fecha creación	Usuario generador	Referencia	REFERENCIA
IF-2021-00462538-APN-GAJ#SSN	04-01-2021 15:39	RNAYAR	Ex-2020-35453931 GERENCIAMIENTO MENSUAL MAYO 2020 por juicios	GERENCIAMIENTO
IF-2021-00462677-APN-GAJ#SSN	04-01-2021 15:39	RNAYAR	Ex-2020-58100485 GERENCIAMIENTO MENSUAL AGOSTO 2020 por pleitos	GERENCIAMIENTO
IF-2021-00462904-APN-GAJ#SSN	04-01-2021 15:40	RNAYAR	Ex-2020-81493306 Múltiples juicios individuales	JUICIOS INDIVIDUALES
IF-2021-00463011-APN-GAJ#SSN	04-01-2021 15:40	RNAYAR	Ex-2019-06709111 GERENCIAMIENTO MENSUAL ENERO 2019	GERENCIAMIENTO
IF-2021-00463105-APN-GAJ#SSN	04-01-2021 15:40	RNAYAR	EX-2019-91134258 - APN-GA#SSN Registro de Liquidadores de Siniestros y Averías – Alta Anexo Diciembre 2020	ALTA/INSCRIPCION REGISTRO LIQUIDADORES DE SINIESTROS Y AVERIAS
IF-2021-00542816-APN-GAJ#SSN	04-01-2021 18:49	RNAYAR	EX-2019-75027190-APN-GA#SSN - FONDO DE RESERVA ART. 34 DE LA LEY 24.557 - PRESTACIONES DINERARIAS.-	FONDO DE RESERVA
IF-2021-00543609-APN-GAJ#SSN	04-01-2021 18:51	RNAYAR	EX-2019-75850202-APN-GA#SSN Assurant Argentina Compañía de Seguros Sociedad Anónima s/Autorización de Plan	AUTORIZACION DE PLAN/ RAMO
IF-2021-00543653-APN-GAJ#SSN	04-01-2021 18:52	RNAYAR	EX-2020-11481993- -APN-GA#SSN - PRUDENTIAL SEGUROS S.A. - Modificaciones Seguro de Salud Colectivo	MODIFICACION CLAUSULA PLAN/RAMO

Consultor:

CONSULTA

cuando mencionan la búsqueda por

4 - Por NUMERO DE DICTAMEN

5_ Por NUMERO DE EXPEDIENTE

Esto es el IF que carga Sandra en la primer columna?

o es un dato que está dentro del IF

o...es algo que hay que agregar?

Respuesta cliente:

En la primera columna de mi registro yo coloco el NUMERO DE IF que da el GDE cuando se firma un DICJU o un FOJDT, a ese nro if yo lo uso como número de Dictamen GAJ (porque es único), porque si a ese IF yo le doy otro número interno sólo sería una numeración cuantitativa únicamente para identificar cuantos dictámenes se firmaron, y me parece que sólo aportaría a confusión.

El número de Expediente debería estar identificado SIEMPRE en la referencia del IF para facilitar la búsqueda, salvo que en el nuevo registro exista un campo para colocar e identificar el nro de EX en donde se dictó el dictamen.

Espero haber sido clara...

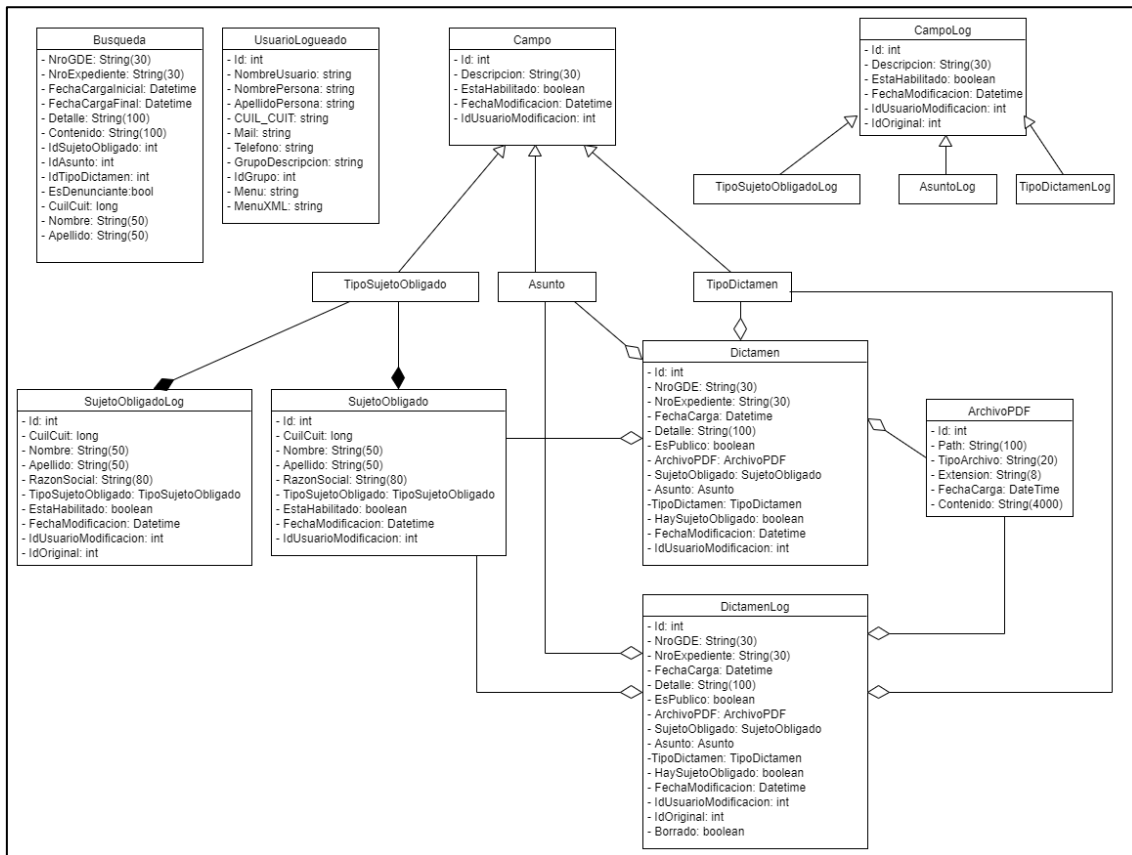
Cliente:

La sugerencia que le hice a Marina, es que en la REFERENCIA de cada Dictamen (DICJU o FOJDT) siempre se consigne el nro de expediente y tema (además del nombre de la cia, productor o lo que sea) porque eso facilitaría la búsqueda.

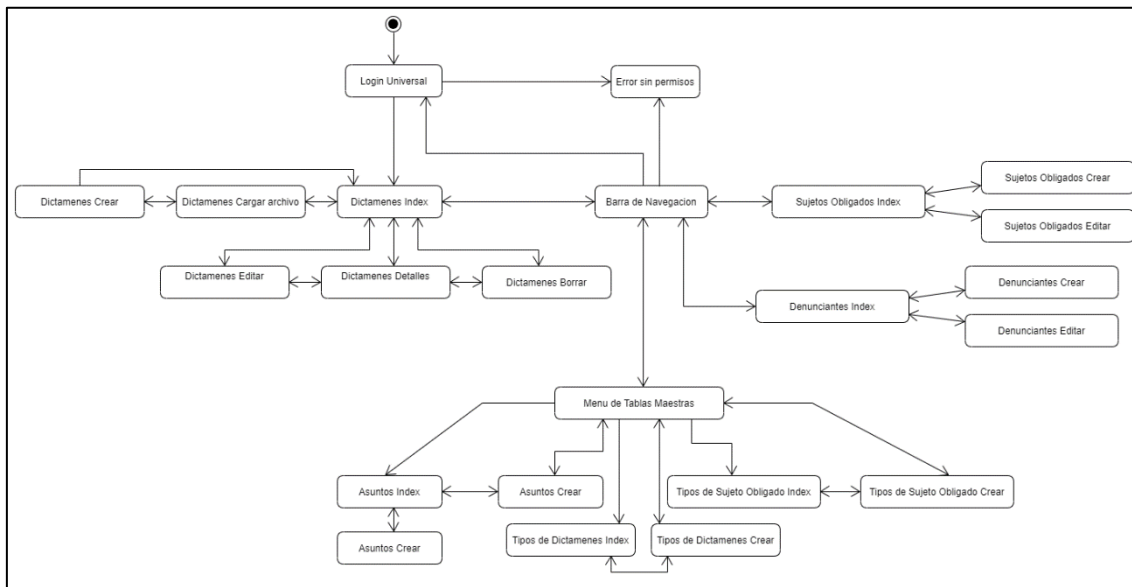
Para agregar el tema, habría que crear una lista con los temas para consignar, así se limitan a especificar correctamente los temas, se unificaría cada tema.

Por ejemplo si se consigna CAMBIO DE DENOMINACION y en otro dictamen se coloca CAMBIO DE NOMBRE, a pesar de que es lo mismo a la hora de buscar se dificultaría.

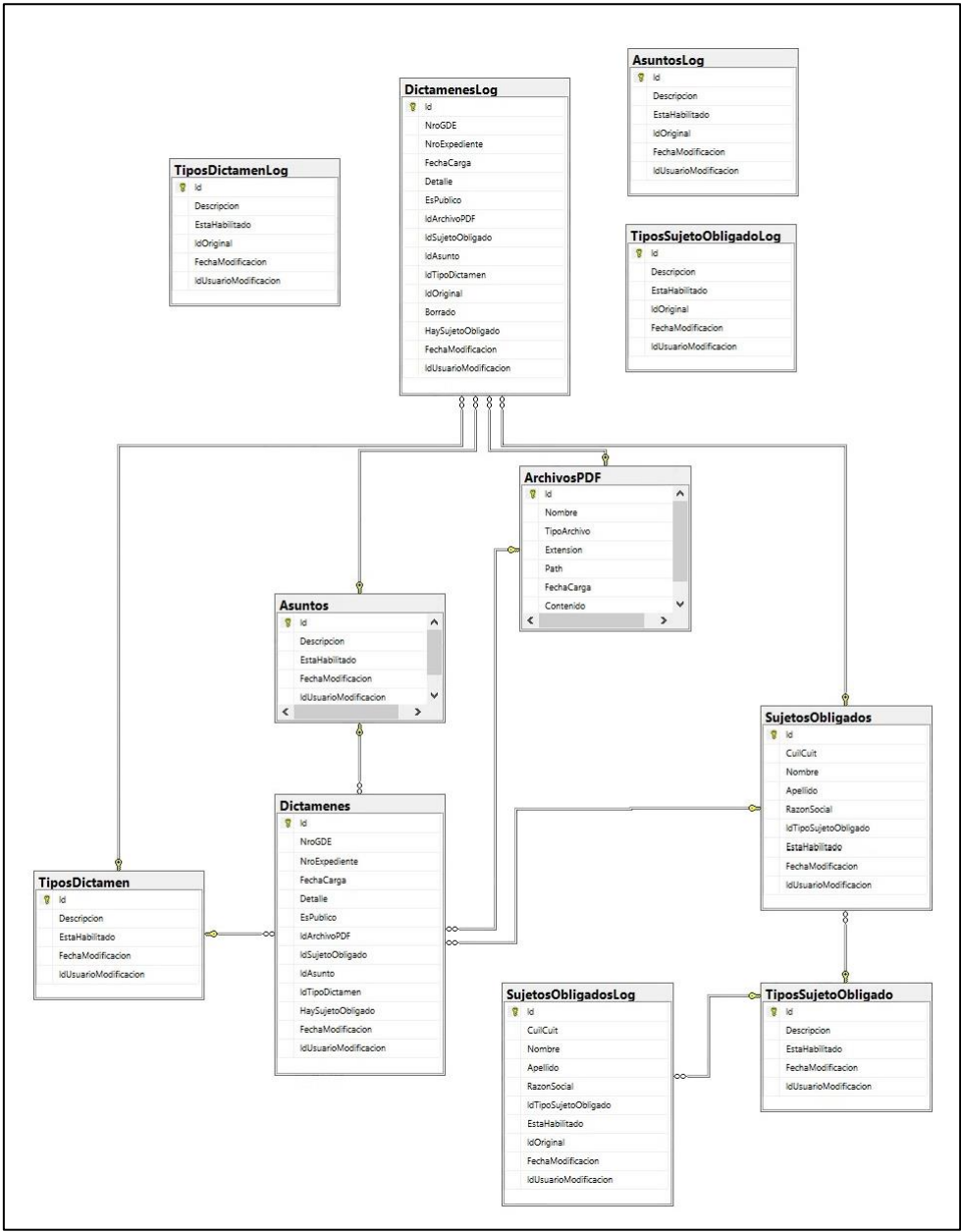
UML Diagrama de Clases



UML Diagrama de navegabilidad



Tablas de Base de Datos



Modelos de Negocio

Dictámenes

Atributos

Numero de GDE: Este campo es obligatorio y debe ser único (No puede haber dos dictámenes con el mismo Número de GDE). Este campo verifica el valor ingresado con este formato regex: `"[iI][fF]-[0-9]{4}-[0-9]+-[aA][pP][nN]-[A-Za-z]+#[A-Za-z]+"`.

Número de Expediente: Este campo es obligatorio y no es único (Puede haber más de un dictamen con el mismo Número de Expediente). Este campo verifica el valor ingresado con este formato regex: `"[eE][xX]-[0-9]{4}-[0-9]+-[aA][pP][nN]-[A-Za-z]+#[A-Za-z]+"`.

Fecha de carga: Este campo corresponde a la fecha de carga en el GDE y es obligatorio. El formato de la fecha es: "26/10/2021 15:40:00".

Archivo: En este campo se almacena el objeto del archivo del Dictamen. Puede ser null en el caso de que se haya cargado sin archivo o se le haya borrado después.

Detalle: Este campo es obligatorio y sería un campo de texto para escribir un detalle del dictamen.

Asunto: Este campo es obligatorio y el valor viene de la tabla maestra de Asuntos.

Tipo de Dictamen: Este campo no es obligatorio y el valor vendría de la tabla maestra de Tipos de Dictamen.

Sujeto Obligado: Contiene el objeto Sujeto Obligado, puede ser "null".

Hay Sujeto Obligado: Valor booleano que indica si el Dictamen tiene o no tiene un Sujeto Obligado.

Id Usuario Modificación: Contiene el ID del usuario que realizó la última modificación de este mismo objeto. Dicho ID viene del login universal.

Fecha Modificación: Indica el último momento cuando se realizó la modificación de este mismo objeto.

Sujetos Obligados:

Atributos:

CUIL/CUIT:

Este campo recibe el CUIL/CUIT del Sujeto Obligado, ser válido (se realiza comprobación matemática del mismo) y también único (no puede haber dos Sujetos Obligados con el mismo CUIL/CUIT). En caso de modificar este valor, se modificará en todos los lugares donde haya sido utilizado.

Comprobación del CUIL/CUIT:

```

46 1 referencia
47 public class CuilCuitValidator : ValidationAttribute
48 {
49 0 referencias
50     protected override ValidationResult IsValid(object value, ValidationContext validationContext)
51     {
52         if (value == null)
53         {
54             var errorMessage = "El Cuit/Cuil no debe der nulo";
55             return new ValidationResult(errorMessage);
56         }
57         string cuilCuit = value.ToString();
58
59         if (cuilCuit.Length != 11)
60         {
61             var errorMessage = "El Cuit/Cuil debe tener 11 carateres";
62             return new ValidationResult(errorMessage);
63         }
64         Regex rg = new Regex("[A-Z_a-z]");
65         cuilCuit = cuilCuit.Replace("-", "");
66         if (rg.IsMatch(cuilCuit))
67         {
68             var errorMessage = "El Cuit/Cuil no es valido";
69             return new ValidationResult(errorMessage);
70         }
71
72         char[] cuitArray = cuilCuit.ToCharArray();
73         double sum = 0;
74         int bint = 0;
75         int j = 7;
76         for (int i = 5, c = 0; c != 10; i--, c++)
77         {
78             if (i >= 2)
79                 sum += (Char.GetNumericValue(cuitArray[c]) * i);
80             else
81                 bint = 1;
82             if (bint == 1 && j >= 2)
83             {
84                 sum += (Char.GetNumericValue(cuitArray[c]) * j);
85                 j--;
86             }
87         }
88         if ((cuitArray.Length - (sum % 11)) != Char.GetNumericValue(cuitArray[cuitArray.Length - 1]))
89         {
90             var errorMessage = "El Cuit/Cuil no es valido";
91             return new ValidationResult(errorMessage);
92         }
93
94         return ValidationResult.Success;
95     }

```

Razón Social:

Este campo recibe la Razón Social de la empresa o compañía. El valor de este campo es el que se va a tomar en la lista desplegable. En caso de modificar este valor, se modificará en todos los lugares donde haya sido utilizado.

Nombre:

Este campo recibe el Nombre del Sujeto Obligado.

Apellido:

Este campo recibe el Apellido del Sujeto Obligado.

Está habilitado:

Tildar esta opción hace que el valor esté disponible para ser seleccionado en las listas desplegables correspondientes. Destildar esta opción hace que no esté disponible para ser seleccionado en las listas desplegables correspondientes, este no elimina el valor de la tabla por lo que en un futuro podría ser reactivado.

Tipo de Sujeto Obligado:

El valor de este campo viene de la tabla maestra de Tipo de Sujeto Obligado y es obligatorio.

Id Usuario Modificación: Contiene el ID del usuario que realizó la última modificación de este mismo objeto. Dicho ID viene del login universal.

Fecha Modificación: Indica el último momento cuando se realizó la modificación de este mismo objeto.

Tipo de sujetos obligados:

Particulares o denunciantes:

CUIL/CUIT: Igual.

Razón Social: Valor "null".

Nombre: Este campo recibe el Nombre del Sujeto Obligado.

Apellido: Este campo recibe el Apellido del Sujeto Obligado.

Está habilitado: Siempre en true.

Tipo de Sujeto Obligado: El valor del Tipo de Sujeto Obligado es cualquiera de la tabla maestra de Tipo de Sujeto Obligado

Compañía:

CUIL/CUIT: Igual.

Razón Social: Este campo recibe la Razón Social de la empresa o compañía.

Nombre: Valor "null".

Apellido: Valor "null".

Está habilitado: Siempre en true.

Tipo de Sujeto Obligado: El valor del Tipo de Sujeto Obligado es siempre "Denunciante".

Campos:

Descripción:

Valor que va a tomar el campo de la tabla maestra. En caso de modificar este valor, se modificará en todos los lugares donde haya sido utilizado.

Está habilitado:

Tildar esta opción hace que el valor esté disponible para ser seleccionado en los desplegables correspondientes. Destildar esta opción hace que no esté disponible para ser seleccionado en los desplegable correspondientes, este no elimina el valor de la tabla por lo que en un futuro podría ser reactivado.

Id Usuario Modificación: Contiene el ID del usuario que realizo la última modificación de este mismo objeto. Dicho ID viene del login universal.

Fecha Modificación: Indica el último momento cuando se realizó la modificación de este mismo objeto.

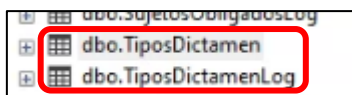
Los objetos Asunto, TipoDictamen y TipoSujetoObligado heredan del objeto Campo.

Objetos con terminación “Log”:

Estos objetos son los utilizados en las tablas Log para hacer el registro de todos los cambios realizados. Para ellos cuentan con un campo extra llamado “IdOriginal” que hace referencia al objeto que era en la tabla principal. Cada archivo Log cuenta con la del momento en que fue modificado (fechaModificacion) y el id del usuario (idUserUsuarioModificacion) que realizo los cambios que se encuentran en el mismo registro. Este comportamiento se repite también en el objeto en la tabla principal que guarda esos dos datos junto a la información actual del objeto.

Ejemplo de funcionamiento:

Para el ejemplo utilizaremos la tabla TiposDictamen y TiposDictamenLog:



Estado inicial de la tabla TiposDictamen (tabla principal):

	Id	Descripcion	EstaHabilitado	FechaModificacion	IdUsuarioModificacion
1	1	DJFC	1	2021-11-03 12:15:28.380	7777
2	2	DICJU	1	2021-11-03 12:15:35.187	7777
3	3	FOJDT	1	2021-11-03 12:15:44.447	7777

Se puede observar que hay objetos cargados por el usuario con ID “7777” el día 3-11-2021 al mediodía.

Estado inicial de la tabla TiposDictamenLog (tabla log):

Id	Descripcion	EstaHabilitado	IdOriginal	FechaModificacion	IdUsuarioModificacion
----	-------------	----------------	------------	-------------------	-----------------------

Se puede observar que no se registró ningún cambio de los objetos de la tabla principal.

Ahora vamos a realizar un cambio sobre el objeto con ID 2, que sería el “DICJU”:

Editar Tipo de Dictamen

Descripción:

☒ Esta habilitado

[← Volver atrás](#) [Guardar](#)

Vamos a modificar el valor de “Esta habilitado”:

Editar Tipo de Dictamen

Descripción:

☐ Esta habilitado

[← Volver atrás](#) [Guardar](#)

El cambio se efectuó:

Listado de todos los Tipos de Dictamen

[+ Cargar nuevo Tipo de Dictamen](#)

Mostrar registros Buscar:

Descripción	1: Esta habilitado	
DICJU	✖	✎
DJFC	✔	✎
FOJDT	✔	✎

Mostrando registros del 1 al 3 de un total de 3 registros

[Anterior](#) [1](#) [Siguiente](#)

Y las bases de datos quedaron de esta manera:

Estado final de la tabla TiposDictamen:

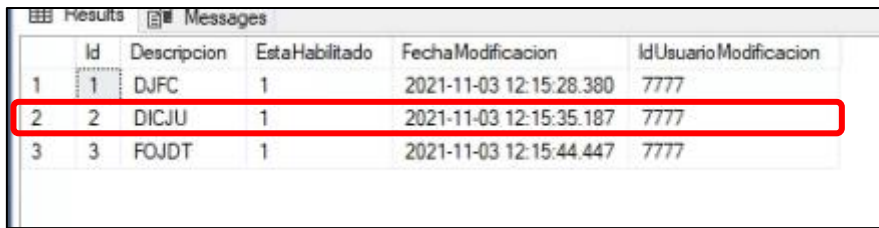
	Id	Descripcion	EstaHabilitado	FechaModificacion	IdUsuarioModificacion
1	1	DJFC	1	2021-11-03 12:15:28.380	7777
2	2	DICJU	0	2021-11-17 11:36:30.237	2310
3	3	FOJDT	1	2021-11-03 12:15:44.447	7777

Podemos observar que el valor se actualizo, “EstaHabilitado” paso de 1 a 0, pero también cambiaron los valores de “FechaModificacion” y “IdUsuarioModificacion” a la fecha de hoy (17-11-2021 al casi mediodía) y al usuario logeado en este momento.

Estado **final** de la tabla TiposDictamenLog:

	Id	Descripcion	EstaHabilitado	IdOriginal	FechaModificacion	IdUsuarioModificacion
1	1	DICJU	1	2	2021-11-03 12:15:35.187	7777

Estado **inicial** de la tabla principal para comparar:



	Id	Descripcion	EstaHabilitado	FechaModificacion	IdUsuarioModificacion
1	1	DJFC	1	2021-11-03 12:15:28.380	7777
2	2	DICJU	1	2021-11-03 12:15:35.187	7777
3	3	FOJDT	1	2021-11-03 12:15:44.447	7777

Acá podemos observar que se agregó un registro con toda la información antigua del objeto que modificamos, pero con el agregado del “IdOriginal” que corresponde al id que tiene en la tabla principal.

Roles:

Consultar:

Este rol tiene como objetivo poder consultar los dictámenes cargados en el sistema.

Tiene acceso a:

- /dictámenes/index
- /dictámenes/details

Cargar:

Este rol tiene como objetivo poder cargar los dictámenes cargados en el sistema.

Tiene acceso a:

- /dictámenes/index
- /dictámenes/details
- /dictámenes/cargarfile
- /dictámenes/create
- /dictámenes/edit
- /dictámenes/delete
- /campos/menu
- /asuntos/index
- /asuntos/create
- /asuntos/edit
- /TiposSujetoObligado/index
- /TiposSujetoObligado/create
- /TiposSujetoObligado/edit
- /TiposDictamen/index
- /TiposDictamen/create
- /TiposDictamen/edit
- /sujetosobligados/index
- /sujetosobligados/create
- /sujetosobligados/edit
- /denunciantes/index
- /denunciantes/edit

Comportamientos específicos por ruta:

/*:

En todas las rutas se verifican el rol para permitir el acceso o no. El rol se extrae de la cookie del usuario.

```
//El usuario que no sea tenga rol de Cargar o Consulta no podra acceder a esta opcion
if (!FrameworkMVC.Security.LoginService.IsAllowed(new[] { Models.Rol.CARGAR.ToString(), Models.Rol.CONSULTAR.ToString() }))
{
    return RedirectToAction("ErrorNoPermisos", "Login");
}
```

En caso de que no se cuente con el permiso necesario para acceder a alguna vista se redirecciona a esta vista:

No tienes permiso

Tu usuario no pertenece al grupo necesario para realizar esta acción. Puedes volver atrás o iniciar sesión con otro usuario que cuente con los permisos requeridos.

[← Volver atrás](#)[Cerrar sesión](#)

/dictámenes/index

Esta vista hace una consulta a través de Linq que devuelve todos los dictámenes que hayan sido cargados **el año actual**. Este filtro utiliza la fecha de carga del dictamen.

```
32 //Trae todos los dictámenes que sean del año actual
33 var dictámenes = db.Dictámenes.Where(d => d.FechaCarga.Year == DateTime.Today.Year).Include(d => d.Asunto).Include(d => d.TipoDictamen);
34 ViewData["IdAconte"] = new SelectList(db.Accontas.Where(a => a.FechaHabilitado <= DateTime.Today), "Id", "Descripcion");
```

/dictámenes/cargarfile - POST

Este endpoint recibe el archivo del Dictamen, lo guarda, lo procesa para sacarle la información y te redirecciona a la carga de los demás datos del Dictamen.

El procesamiento lo hace en dos partes:

Primero extrae todo el texto con la función `ExtractTextFromPdf` dentro del `FileController` que recibe la dirección relativa de donde se encuentra el archivo y devuelve un string con todo el texto del PDF:

```
43
44 3 referencias
45 public static string ExtractTextFromPdf(string path)
46 {
47     string pathAbsolute = HostingEnvironment.MapPath(path);
48     PdfReader reader2 = new PdfReader(pathAbsolute);
49     string strText = string.Empty;
50
51     for (int page = 1; page <= reader2.NumberOfPages; page++)
52     {
53         ITextExtractionStrategy its = new SimpleTextExtractionStrategy();
54         PdfReader reader = new PdfReader(pathAbsolute);
55         String s = PdfTextExtractor.GetTextFromPage(reader, page, its);
56
57         s = Encoding.UTF8.GetString(ASCIIEncoding.Convert(Encoding.Default, Encoding.UTF8, Encoding.Default.GetBytes(s)));
58         strText += s;
59         reader.Close();
60     }
61     return strText;
62 }
```

En la segunda parte del procesamiento se intenta extraer el Número de GDE, el Número de Expediente y la Fecha de Carga del contenido del archivo previamente cargado. Para eso se usa la función `ExtratDictamenFromString` dentro del `DictámenesController` que recibe el contenido del Archivo en forma de string y devuelve un objeto `Dictamen` con la información que encontró rellena en los campos correspondientes:

```

745 //referencia
746 private static Dictamen ExtratDictamenFromString(string contenido)
747 {
748     Dictamen dict = new Dictamen();
749
750     Regex numeroGDE = new Regex("If-[0-9]{4}-[0-9]+-APN-[A-Z]+#[A-Z]+", RegexOptions.IgnoreCase);
751     MatchCollection matches = numeroGDE.Matches(contenido);
752     try
753     {
754         //obtiene el primer resultado
755         dict.NroGDE = matches[0].Value;
756     }
757     //en caso de que no encuentre ninguna, no hace nada y deja el valor por defecto
758     catch { }
759
760     Regex numeroExpediente = new Regex("[E][X] *- * [0-9]{4} *- * [0-9]+? ?- ?-? ?APN *- * [A-Z]+ *# * [A-Z]+|[E][X] *- * [0-9]{4} *- * [0-9]+", RegexOptions.IgnoreCase);
761     matches = numeroExpediente.Matches(contenido);
762     try
763     {
764         //obtiene el primer resultado y normaliza el campo
765         //ya que el numero de expediente es extraido directamente de la descripción del dictamen
766         dict.NroExpediente = matches[0].Value.Replace(" ", "").Replace("-", "");
767     }
768     catch { }
769
770     Regex date = new Regex("[0-9]{4}.[0-9]{2}.[0-9]{2} [0-9]{2}:[0-9]{2}:[0-9]{2}", RegexOptions.IgnoreCase);
771     matches = date.Matches(contenido);
772     try
773     {
774         DateTime fecha;
775         //trata de parsear la fecha con ese formato
776         DateTime.TryParseExact(matches[0].Value, "yyyy.MM.dd HH:mm:ss", System.Globalization.CultureInfo.InvariantCulture, System.Globalization.DateTimeStyles.None, out fecha);
777         //si la instrucción anterior no eleva una excepción, guarda la fecha en el dictamen
778         dict.FechaCarga = fecha;
779     }
780     catch
781     {
782         //en caso de no lograrlo o no haber resultado de la búsqueda, devuelve la fecha actual
783         dict.FechaCarga = DateTime.Now;
784     }
785     return dict;
786 }

```

Debido a que los PDF generados por el GDE ya no incorporaban el Número de GDE como texto seleccionable se tuvo que extraer el mismo del nombre del archivo:

```

142 dictamen.ArchivoPDF = archivo;
143 dictamen.NroGDE = Path.GetFileNameWithoutExtension(file.FileName).Replace("%", "#");
144

```

Luego el controlador hace un return de la vista de Create con el objeto dictamen resultante del procesamiento.

```

129 // cargo la información para el formulario Create y devuelvo la VIEW del create con la información extraída del Archivo
130 //que está dentro de la variable dictamen
131 return View("Create", dictamen);
132 }

```

/dictamenes/create - POST

Para crear el dictamen primero se comprueba que no exista un dictamen con el mismo Número de GDE:

```

164
165 //en caso de que exista un dictamen con el mismo numero de GDE
166 if (db.Dictamenes.FirstOrDefault(d => d.NroGDE == dictamen.NroGDE) != null)
167 {
168     //se agrega un error en ese campo y se vuelve a solicitar la carga
169     ModelState.AddModelError("NroGDE", "Ya existe un Dictamen con ese Número de GDE");
170 }
171

```

Luego limpio y estandarizo la información de algunos campos para mayor homogeneidad en toda la aplicación:

```

172 //limpio y estandarizo la información
173 dictamen.FechaModificacion = DateTime.Now;
174 dictamen.IdUsuarioModificacion = LoginController.GetUserDataIdentity(User.Identity).Id;
175 dictamen.NroGDE = dictamen.NroGDE.ToUpper();
176 dictamen.NroExpediente = dictamen.NroExpediente.ToUpper();
177 dictamen.Detalle = dictamen.Detalle.ToUpper();
178 dictamen.EsPublico = false;
179

```

Después ejecuto el bloque de código para agregar el Sujeto Obligado:

```

181 //compruebo si la data del formulario del denunciante es distinto al valor por defecto
182 if (dictamen.SujetoObligado.CuilCuit > 0)
183 {
184     //busco si existe otro sujeto obligado con el mismo cuilCuit
185     SujetoObligado sujetoObligadoExistente = db.SujetosObligados.FirstOrDefault(s => s.CuilCuit == dictamen.SujetoObligado.CuilCuit);
186     if (sujetoObligadoExistente != null)
187     {
188         //chequeo si el sujetoObligado existente no es denunciante
189         if (sujetoObligadoExistente.IdTipoSujetoObligado != db.TiposSujetoObligado.First(m => m.Descripcion == "Denunciante").Id)
190         {
191             //devuelvo un error ya que ese cuilCuit corresponde a una empresa
192             ModelState.AddModelError("SujetosObligados.CuilCuit", "Ya existe un Sujeto Obligado con ese Número de Cuil");
193         }
194         else
195         {
196             //en caso de ser un denunciante, simplemente uso el sujetoObligado denunciante existente
197             dictamen.SujetoObligado = sujetoObligadoExistente;
198         }
199     }
200     //en caso de no existir un sujetoObligado con el mismo cuilCuit, lo creo
201     else
202     {
203         dictamen.SujetoObligado.IdTipoSujetoObligado = db.TiposSujetoObligado.First(m => m.Descripcion == "Denunciante").Id;
204         dictamen.SujetoObligado.IdUsuarioModificacion = LoginController.GetUserDataIdentity(User.Identity).Id;
205         dictamen.SujetoObligado.FechaModificacion = DateTime.Now;
206         db.SujetosObligados.Add(dictamen.SujetoObligado);
207     }
208 }
209 //en caso de no haber un valor en el formulario de Denunciante
210 else
211 {
212     //remuevo la comprobacion del campo cuilCuit
213     ModelState.Remove("SujetoObligado.CuilCuit");
214     //seteo el SujetoObligado para que corresponda con el IdSujetoObligado
215     //en caso que el IdSujetoObligado sea null Find() devuelve null tambien
216     dictamen.SujetoObligado = db.SujetosObligados.Find(dictamen.IdSujetoObligado);
217 }
218

```

/dictamenes/buscar – POST

Este endpoint recibe los parámetros de búsqueda desde el buscador de “Búsqueda avanzada”:

+ Cargar nuevo Dictamen

Búsqueda Avanzada

Número de GDE	Tipo de Dictamen
<input type="text" value="Ingrese Número de GDE"/>	<input type="text" value="Elija uno"/>
Número de Expediente	Asunto
<input type="text" value="Ingrese Número de Expediente"/>	<input type="text" value="Elija uno"/>
Fecha de Carga	CUIL/CUIT
Desde: <input type="text" value="01/01/2021 00:00"/>	<input type="text" value="Ingrese el CUIL/CUIT completo del Sujeto Obligado"/>
Hasta: <input type="text" value="dd/mm/aaaa --:--"/>	<input type="checkbox"/> Sujeto Obligado es Denunciante
Contenido	Empresas
<input type="text" value="Ingrese Texto a buscar en el cuerpo de los dictamenes"/>	<input type="text" value="Elija uno"/>
Detalle	Tipo de Sujeto Obligado
<input type="text" value="Ingrese el detalle del Dictamen"/>	<input type="text" value="Elija uno"/>

Borrar filtros

Buscar

Se ejecuta un Store Procedure con los parámetros recibidos:

```

247 //ejecuto el StoreProcedure "sp_FiltrarDictamenes" de la base de datos con los valores pasado por parametro
248 var dictamenes = db.Sp_FiltrarDictamenes(búsqueda.NroGDE, búsqueda.NroExp, búsqueda.FechaCargaInicio, búsqueda.FechaCargaFinal,
249 búsqueda.Detalle, búsqueda.Contenido, búsqueda.IdAsunto, búsqueda.IdTipoDictamen, búsqueda.IdSujetoObligado,
250 búsqueda.IdTipoSujetoObligado, búsqueda.CuilCuit, búsqueda.Nombre, búsqueda.Apellido);
251

```

Store Procedure:


```

ALTER PROCEDURE [dbo].[sp_FiltrarDictamenes]
    @NroGDE nvarchar(30) = null,
    @NroExpediente nvarchar(30) = null,
    @FechaInicio datetime = null,
    @FechaFinal datetime = null,
    @Detalle nvarchar(100) = null,
    @Contenido nvarchar(100) = null,
    @IdAsunto int = null,
    @IdTipoDictamen int = null,
    @IdSujetoObligado int = null,
    @IdTipoSujetoObligado int = null,
    @CuilCuit bigint = null,
    @Nombre nvarchar(50) = null,
    @Apellido nvarchar(50) = null
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;
    -- Insert statements for procedure here
    SELECT * FROM Dictamenes D
    LEFT JOIN SujetosObligados SJ ON SJ.Id = D.IdSujetoObligado
    LEFT JOIN ArchivosPDF Ar ON Ar.Id = D.IdArchivoPDF
    INNER JOIN Asuntos A ON A.Id = D.IdAsunto
    WHERE (@CuilCuit IS NULL OR SJ.CuilCuit = @CuilCuit)
    AND (@IdAsunto IS NULL OR D.IdAsunto = @IdAsunto)
    AND (@IdTipoDictamen IS NULL OR D.IdTipoDictamen = @IdTipoDictamen)
    AND (@IdSujetoObligado IS NULL OR D.IdSujetoObligado = @IdSujetoObligado)
    AND (@IdTipoSujetoObligado IS NULL OR SJ.IdTipoSujetoObligado = @IdTipoSujetoObligado)
    AND (@FechaInicio IS NULL OR D.FechaCarga >= @FechaInicio)
    AND (@FechaFinal IS NULL OR D.FechaCarga < @FechaFinal)
    AND (@NroGDE IS NULL OR D.NroGDE like '%' + @NroGDE + '%')
    AND (@NroExpediente IS NULL OR D.NroExpediente like '%' + @NroExpediente + '%')
    AND (@Contenido IS NULL OR Ar.Contenido like '%' + @Contenido + '%')
    AND (@Detalle IS NULL OR D.Detalle like '%' + @Detalle + '%')
END

```

La lista de dictámenes se pasa a la vista “Index” de “Dictamenes” con los valores de la búsqueda realizada.

/* /edit – POST

Para la edición de objetos dentro de la aplicación se utiliza el mismo proceso, en este caso representado con la entidad Asunto:

1. Guarda el objeto con la información sin actualizar en una variable aparte, llamado nombre de entidad + “Viejo”. En este caso es nombrado (nombre de la entidad = “asunto” + “Viejo”). El ID de que objeto debo traer de la base de datos viene del objeto que viene por parámetro con la información que hay que modificar.

```

147 | | | | |
148 | | | | | Asunto asuntoViejo = db.Asuntos.AsNoTracking().First(d => d.Id == asunto.Id);
149 | | | | |

```

2. Luego con la información del objeto sin actualizar (este caso “asuntoViejo”) se crea un objeto LOG de la misma entidad (en este caso “AsuntoLog”).

```

150 AsuntoLog asuntoLog = new AsuntoLog
151 {
152     IdOriginal = asuntoViejo.Id,
153     Descripcion = asuntoViejo.Descripcion,
154     EstaHabilitado = asuntoViejo.EstaHabilitado,
155     FechaModificacion = asuntoViejo.FechaModificacion,
156     IdUsuarioModificacion = asuntoViejo.IdUsuarioModificacion
157 };

```

3. A continuación, actualizo el idUsuarioModificacion y la FechaModificacion del objeto “asunto” que vino por parámetro con la información modificada.

```

158 asunto.IdUsuarioModificacion = LoginController.GetUserDataIdentity(User.Identity).Id;
159 asunto.FechaModificacion = DateTime.Now;
160
161

```

4. Después se modifica el registro en la tabla principal de esa entidad de la base de datos (en este caso la tabla sería “Asuntos”) y se agrega el objeto LOG a la tabla LOG de la base de datos (en este caso la tabla sería “AsuntosLog”).

```

162 db.Entry(asunto).State = EntityState.Modified;
163 db.AsuntosLog.Add(asuntoLog);
164 db.SaveChanges();
165

```

La consecuencia de este proceso es que hay un registro de todas las modificaciones hechas sobre todos los objetos de la aplicación: “AsuntosLog” y hay una tabla única con las últimas versiones de los objetos: “Asuntos”.

Claves del Web.Config

connectionStrings

```

<connectionStrings>
  <!-- CONEXION A LA BASE DE DATOS DESARROLLO -->
  <add name="SQLServer" connectionString="Data Source=172.16.2.112;Initial Catalog=DICTIONARIO" providerName="System.Data.SqlClient" />
  <!-- CONEXION A LA BASE DE DATOS PRODUCCION -->
  <!--<add name="SQLServer" connectionString="" providerName="System.Data.SqlClient" />-->
</connectionStrings>

```

Se debe dejar solo uno descomentado, según en qué ambiente se encuentre la aplicación.

Authentication

Valores:

- name
 - Uso: Nombre de la Cookie de Login
 - Valor: “.DICTIONARIO”
- loginUrl
 - Uso: URL donde se va a redireccionar para loguear
 - Valor: “login/login”
- timeout
 - Uso: Tiempo en minutos de duración de sesión
 - Valor: 30

En formato RAW:

```
<authentication mode="Forms">
```

```
  <forms name=".DICTAMENES" loginUrl="login/login" timeout="30" />
```

```
</authentication>
```

Implementación de Login Universal

La implementación del Login Universal está hecha con una adaptación de la librería FrameworkSSN. Se tuvo que realizar la adaptación ya que la original no es compatible con el patrón MVC, para consultas de dicha librería puede consultar la documentación de la misma.

Framework.config:

Es el archivo que configura la librería FrameworkMVC:

```
<?xml version="1.0" encoding="utf-8" ?>
<framework>
  <!-- ID de Aplicacion en Desarrollo -->
  <!--<AppID>261</AppID>-->

  <!-- ID de Aplicacion en Produccion -->
  <AppID>259</AppID>

  <!-- URL para redireccionar en el Login en Desarrollo -->
  <!--<LoginUrl>http://segurodes40.ssn.gob.ar/loginuniversal/indexframe.aspx?vinoapp={0}&local={1}</LoginUrl>-->

  <!-- URL para redireccionar en el Login en Produccion -->
  <LoginUrl>http://seguro2.ssn.gob.ar/loginuniversal/indexframe.aspx?vinoapp={0}&local={1} </LoginUrl>

  <Roles>
    <!-- ID de Roles en Desarrollo -->
    <add Name="CARGAR" Id="212"/>
    <add Name="CONSULTAR" Id="211"/>

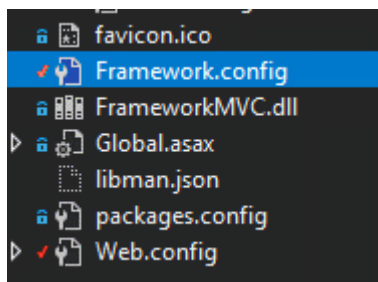
    <!-- ID de Roles en Produccion -->
    <!--<add Name="CARGAR" Id="211"/>-->
    <!--<add Name="CONSULTAR" Id="212"/>-->
  </Roles>

  <WebReferences>
    <!-- URL del servicio del Login Universal en Desarrollo -->
    <!--<add Name="LoginUniversal" Url="http://segurodes40.ssn.gob.ar/LoginUniversal/WCFLoginUniversal.LogLoginService.svc"/>-->

    <!-- URL del servicio del Login Universal en Produccion -->
    <add Name="LoginUniversal" Url="http://seguro2.ssn.gob.ar/LoginUniversal/WCFLoginUniversal.LogLoginService.svc"/>
  </WebReferences>
</framework>
```

El archivo cuenta con los valores duplicados, un campo para Desarrollo y otro para Producción.

Y está ubicado en la raíz del proyecto:



Login:

El proceso de login se realiza en el LoginController en el método Login:

```

[AllowAnonymous]
0 referencias
public ActionResult Login(string ID)
{
    //comprueba si el usuario esta logueado o si tiene cookie en su sistema
    if (!User.Identity.IsAuthenticated)
    {
        //si no esta logueado se envia el ID al metodo para loguearnos
        if (!LoginService.LoginUniversalCallbackCookie(ID, menuNavBarBootstrap: true))
        {
            //en caso de que el ID sea nulo o haya fallado el logueo, se redirige a la URL del login universal
            return Redirect(LoginService.GetURLLoginUniversal());
        }
    }
    //si ya estaba logueado o se completo correctamente en logueo, se redirige al index de Dictamenes
    return RedirectToAction("Index", "Dictamenes");
}

```

Dicho endpoint es el que recibe el callback del Login Universal con el ID de sesión del usuario que se logueo, por lo que en la base de datos de aplicaciones debe ponerse este endpoint.

Logout:

Para cerrar sesión dentro de la aplicación se debe ingresar a este método:

```

[Authorize]
0 referencias
public ActionResult Logout()
{
    //borra la cookie en el usuario
    FormsAuthentication.SignOut();
    //redirige al index de Dictamenes
    return RedirectToAction("Index", "Dictamenes");
}

```