



Manual Técnico de SIFEI Signature

CFDI v3.3

octubre 2017

Control de Versiones

Nombre	No. Versión	Modificaciones	Fecha
Daniel Jesús Hernández Francisco	1.0	Generación del documento	19/oct/2017

Control de Revisiones

Nombre	No. Versión	Modificaciones	Fecha
Mary Quiahua Iztacua	1.0	Revisión técnica del documento.	19/oct/2017

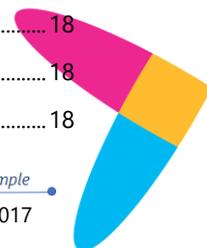
Control de Autorizaciones

Nombre	No. Versión	Modificaciones	Fecha
Asley Alberto Cristales Pavón	1.0	Revisión, correcciones menores técnicas y de redacción y estilo, y autorización de la publicación.	30/oct/2017

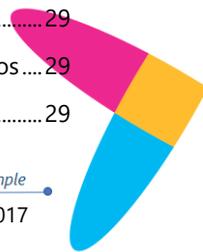


Contenido

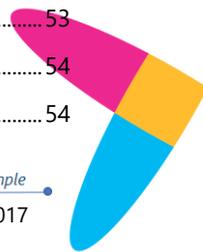
Listado de Imágenes.....	viii
1. Generalidades.....	1
1.1 Introducción.....	1
1.2 Objetivo.....	1
1.3 Alcance.....	1
1.4 Requerimientos Mínimos.....	1
1.5 Términos y Definiciones.....	1
1.6 Términos y definiciones técnicas.....	2
1.7 Referencias.....	2
1.8 Recursos.....	2
1.9 Versiones de CFDI que se timbran.....	2
1.9.1 Métodos de SIFEI Signature.....	2
2. Convenciones de documentación.....	3
2.1 Atributos opcionales.....	3
3. Primeros pasos (getting started).....	4
3.1 Agregar recursos.....	5
3.1.1 Copiar la carpeta SAT y las librerías.....	5
3.1.2 Incluir la carpeta SAT y dll en el proyecto.....	5
3.1.3 Configurar los archivos para ser copiados en el directorio de salida.....	6
3.2 Agregar la biblioteca al proyecto.....	7
3.3 Agregar información del servicio web.....	9
3.4 Crear base de datos.....	11
3.4.1 Comandos usados para crear la base de datos.....	11
3.4.2 Cambiar max_allowed_packet (Opcional).....	12
3.4.3 Ejecutar el script.....	14
4. Ejemplos.....	15
4.1 Instanciar un Objeto tipo CFDI33.....	15
4.2 Establecer CfdiRelacionados.....	16
4.3 Agregar un CfdiRelacionado.....	16
4.4 Agregar el Emisor.....	16
4.5 Agregar el Receptor.....	17
4.6 Agregar un concepto.....	17
4.7 Agregar Información aduanera a un concepto.....	18
4.8 Agregar un impuesto traslado a un concepto.....	18
4.9 Agregar total impuestos de comprobante.....	18



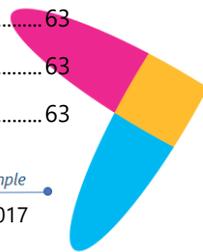
- 4.10 Agregar un impuesto traslado a impuestos de comprobante..... 19
- 4.11 Generar el XML..... 19
- 4.12 Sellar el comprobante 21
- 4.13 Timbrar el comprobante..... 22
- 4.14 Crear Complementos..... 23
- 4.15 Definir complemento Comercio Exterior..... 23
 - 4.15.1 Establecer Emisor de Comercio Exterior 24
 - 4.15.2 Definir Receptor de Comercio Exterior 24
 - 4.15.3 Agregar mercancía de Comercio Exterior 24
 - 4.15.4 Generar el Complemento Comercio Exterior..... 24
- 4.16 Definir complemento Nómina 25
 - 4.16.1 Agregar un emisor a Nómina 25
 - 4.16.2 Agregar un receptor a Nómina..... 25
 - 4.16.3 Establecer que Nómina tiene percepciones 25
 - 4.16.4 Agregar una percepción a percepciones de Nómina..... 26
 - 4.16.5 Agregar una Separación indemnización a percepciones de Nómina 26
 - 4.16.6 Establer deducciones a Nómina 26
 - 4.16.7 Agregar una deducción a deducciones de Nómina 26
 - 4.16.8 Generar el Complemento Nómina 26
- 4.17 Definir complemento Detallista 26
 - 4.17.1 Establecer el requestForPaymentIdentification a Detallista..... 27
 - 4.17.2 Agregar un specialInstruction a Detallista..... 27
 - 4.17.3 Establecer el orderIdentification a Detallista..... 27
 - 4.17.4 Agregar un referenceIdentification al AdditionalInformation de Detallista 27
 - 4.17.5 Establecer un DeliveryNote a Detallista 27
 - 4.17.6 Definir el Buyer (comprador) de Detallista 27
 - 4.17.7 Definir el Seller (vendedor) a Detallista 27
 - 4.17.8 Establer el shipTo (lugar de envío) de Detallista 28
 - 4.17.9 Establecer el InvoiceCreator a Detallista..... 28
 - 4.17.10 Agregar un Customs a Detallista..... 28
 - 4.17.11 Agregar un currency a Detallista 28
 - 4.17.12 Establecer el paymentTerms a Detallista 28
 - 4.17.13 Establecer un allowanceCharge a Detallista para definir cargos o descuentos 28
 - 4.17.14 Definir el totalAmount a Detallista 29
 - 4.17.15 Definir el TotalAllowanceChargea Detallista para definir el monto total de cargos o descuentos..... 29
 - 4.17.16 Generar el complemento Detallista 29



- 4.18 Definir Complemento de Recepción de Pagos.....29
 - 4.18.1 Agregar un DoctoRelacionado a Pago.....30
 - 4.18.2 Generar el Complemento Pago30
- 4.19 Definir y generar el complemento IEDU (a un concepto)30
- 4.20 Recuperar timbre.....30
- 4.21 Cancelar un CFDI31
- 5. Referencia32
 - 5.1 Namespaces32
 - 5.2 Clases principales32
 - 5.3 Namespace SifeiSignature3_3.Version3_3.....32
 - 5.3.1 Clase CFDI33.....32
 - 5.3.2 Métodos33
 - 5.3.3 Enumeraciones:.....37
 - 5.4 Namespace SifeiSignature3_3.Timbre11.....37
 - 5.4.1 Clases.....37
 - 5.4.2 Clase TimbreBE37
 - 5.5 Namespace SifeiSignature3_3.Complementos.....37
 - 5.5.1 Clase Complemento37
 - 5.6 Namespace SifeiSignature3_3>Returns.....50
 - 5.6.1 Clase ReturnXML.....50
 - 5.6.2 Clase ReturnTimbre.....50
 - 5.6.3 Clase ReturnXmlComplemento50
 - 5.6.4 Clase ReturnCancelacion.....51
 - 5.7 Namespace SifeiSignature3_3.Complementos.Detallista131.....51
 - 5.7.1 Enumeraciones de complemento Detallista versión 1.3.1.....51
 - 5.8 Namespace SifeiSignature3_3.Complementos.Iedu10.....52
 - 5.8.1 Enumeraciones de complemento Instituciones Educativas versión 1.052
 - 5.8.2 Enumeraciones de complemento Nómina versión 1.2.....52
 - 5.9 Referencia del Namespace SifeiSignature3_3.Cancelacion.....52
 - 5.9.1 Clase Acuse.....52
 - 5.9.2 Clase AcuseDetalle.....52
 - 5.9.3 Clase Cancelacion53
 - 5.10 Referencia del Namespace SifeiSignature3_3.Complementos.Cce11.....53
 - 5.10.1 Clases.....53
 - 5.10.2 Clase ComercioExterior.....54
 - 5.10.3 Clase ComercioExteriorDestinatario.....54



- 5.10.4 Clase ComercioExteriorDestinatarioDomicilio55
- 5.10.5 Clase ComercioExteriorEmisor55
- 5.10.6 Clase ComercioExteriorEmisor55
- 5.10.7 Clase ComercioExteriorMercancia56
- 5.10.8 Clase ComercioExteriorMercanciaDescripcionesEspecificas.....56
- 5.10.9 Clase ComercioExteriorPropietario56
- 5.10.10 Clase ComercioExteriorReceptor57
- 5.10.11 Clase ComercioExteriorReceptorDomicilio.....57
- 5.11 Referencia del Namespace SifeiSignature3_3.Complementos.ImpLocal1057
 - 5.11.1 Clase ImpuestosLocales.....57
 - 5.11.2 Clase ImpuestosLocalesRetencionesLocales58
 - 5.11.3 Clase ImpuestosLocalesTrasladosLocales58
- 5.12 Referencia del Namespace SifeiSignature3_3.Complementos.Nomina1258
 - 5.12.1 Clases.....58
 - 5.12.2 Enumeración NominaReceptorSindicalizado59
 - 5.12.3 Clase Nomina59
 - 5.12.4 Clase NominaDeducciones59
 - 5.12.5 Clase NominaDeduccionesDeduccion.....59
 - 5.12.6 Clase NominaEmisor.....60
 - 5.12.7 Clase NominaEmisorEntidadSNCF.....60
 - 5.12.8 Propiedades60
 - 5.12.9 Clase NominaIncapacidad.....60
 - 5.12.10 Clase NominaOtroPago.....60
 - 5.12.11 Clase NominaOtroPagoCompensacionSaldosAFavor60
 - 5.12.12 Clase NominaOtroPagoSubsidioAlEmpleo.....61
 - 5.12.13 Clase NominaPercepciones.....61
 - 5.12.14 Clase NominaPercepcionesJubilacionPensionRetiro61
 - 5.12.15 Clase NominaPercepcionesPercepcion.....61
 - 5.12.16 Clase NominaPercepcionesPercepcionAccionesOTitulos.....62
 - 5.12.17 Clase NominaPercepcionesPercepcionHorasExtra62
 - 5.12.18 Clase NominaPercepcionesSeparacionIndemnizacion62
 - 5.12.19 Clase NominaReceptor62
 - 5.12.20 Clase NominaReceptorSubContratacion63
- 5.13 Referencia del Namespace SifeiSignature3_3.Complementos.Pagos10.....63
 - 5.13.1 Clases63
 - 5.13.2 Clase Pagos63



5.13.3	Clase PagosPago.....	63
5.13.4	Clase PagosPagoDoctoRelacionado	64
5.13.5	Clase PagosPagoImpuestos	64
5.13.6	Clase PagosPagoImpuestosRetencion	64
5.13.7	Clase PagosPagoImpuestosTraslado.....	64
5.14	Referencia del Namespace SifeiSignature3_3.Complementos.Iedu10	65
5.14.1	Clases.....	65
5.14.2	Enumeración instEducativasNivelEducativo.....	65
5.14.3	Clase instEducativas.....	65
5.15	Codigos de respuesta de cancelación.....	65
5.16	Códigos de Error de la Matriz de validaciones para CFDI versión 3.3	65
6.	Información de Contacto con SIFEI.....	70



Listado de Imágenes

Imagen 1 Creación de proyecto .Net Framework 3.5	4
Imagen 2 Proyecto recién creado.....	4
Imagen 3 Copiando los recursos al proyecto.....	5
Imagen 4 Inclusión de carpeta SAT en el proyecto	6
Imagen 5 Inclusión del resto de bibliotecas	6
Imagen 6 Modificando la propiedad "Copiar al directorio de Salida"	7
Imagen 7 Modificando la propiedad "Copiar al directorio de salida" a "Copiar siempre"	7
Imagen 8 Agregar referencia al proyecto	8
Imagen 9 Carpeta donde está SifeiSignature.....	8
Imagen 10 Referencia de biblioteca agregada	9
Imagen 11 Configuración de aplicación requerida para SIFEI Signature	9
Imagen 12 Seleccionado Archivo de Configuración de Aplicación.....	10
Imagen 13 Archivo de configuración limpio.....	10
Imagen 14 Archivo de configuración con la configuración necesaria para SIFEI Signature	11
Imagen 15 Ejecuta el script para la creación de la base de datos	11
Imagen 16 Consulta para mostrar el valor de max_allowed_packet.....	11
Imagen 17 Error 2006 MySQL server has gone away.....	12
Imagen 18 Variable max_allowed_packet = 1MB: causará "Error 2006: MySQL server has gone away"	12
Imagen 19 Comando para cambiar de forma global el límite que define max_allowed_packet a 32 MB.....	12
Imagen 20 Cambiando variable Max_allowed_packet temporalmente (este efecto no cambia la sesión actual y dura solo mientras el servicio esté activo -hasta que el servicio sea reiniciado-)	13
Imagen 21 Archivo de configuración del servidor de MariaDB, cambio de 1mb a 16 MB (se debe reiniciar el servicio)	13
Imagen 22 Procedimiento para la ejecución del script, comenzando por el listado de la base de datos, la ejecución del script y finalizando por la comprobación de su creación	14
Imagen 23 Atributos de CFDI agregados	15
Imagen 24 Nodo CfdiRelacionados agregado.....	16
Imagen 25 CFDIRelacionado agregado	16
Imagen 26 Nodo Emisor agregado	16
Imagen 27 Nodo Receptor agregado	17
Imagen 28 Nodo Concepto agregado	17
Imagen 29 InformacionAduanera agregada a conceptos	18
Imagen 30 Resultado de agregar un impuesto traslado al concepto	18
Imagen 31 Nodo Impuestos agregado a comprobante.....	19
Imagen 32 Impuesto traslado agregado.....	19
Imagen 33 Comprobante Sellado	22
Imagen 34 Comprobante timbrado.....	23



1. Generalidades

1.1 Introducción

El presente documento es el *Manual Técnico* para el uso e integración de la biblioteca *SIFEI Signature* versión 3.3 a proyectos bajo plataforma .Net, además de ofrecer la referencia de la biblioteca y ejemplos para emitir, sellar, timbrar y cancelar CFDIs.

Solución Integral de Facturación Electrónica e Informática SIFEI S. A. de C. V. es un Proveedor Autorizado de Certificación de CFDI (PCCFDI antes PAC), con número de autorización 58355 y cumplimos con los requerimientos funcionales, servicios generales y niveles de servicio (SLA) exigidos por el SAT.

1.2 Objetivo

Orientar a usuario de la biblioteca *SIFEI Signature* para emitir y certificar CFDIs válidos siguiendo las normas establecidas para tipo de documento fiscal.

1.3 Alcance

- Emitir CFDIs en versión 3.3.
- La certificación se realiza con base al Anexo 20 y a la *Matriz de Validaciones* de los CFDI y complementos.
- Entrega respuestas sobre los códigos de error de la *Matriz de Validaciones* vigente.

1.4 Requerimientos Mínimos

- Visual Studio 2012 o superior
- Tener instalado como mínimo .Net Framework 3.5
- Tener instalado MariaDB o tener acceso remoto a este gestor
 - Tener las credenciales de un usuario con privilegios para crear base de datos
 - Contar con un cliente para el gestor (para la creación de la base de datos)
 - El servicio MariaDB debe tener la variable "max_allowed_packet" establecida para paquetes mayores a 8 MB.

1.5 Términos y Definiciones

Acrónimo	Definición
CFDI	Comprobante Fiscal Digital por Internet
NDA	<i>Non-Disclosure Agreement</i> , documento de Acuerdo de No Divulgación
PAC	Proveedor Autorizado de Certificación
PCCFDI	Proveedor de Certificación de Comprobantes Fiscales Digitales por Internet (antes PAC)
PCECFDI	Proveedor de Certificación de Expedición de Comprobantes Fiscales Digitales por Internet, que aplica a los contribuyentes que expiden CFDIs al ser Adquirientes de Bienes y Servicios, y Sector Primario (simplificado "Sector Primario")
SAT	Servicio de Administración Tributaria
SIFEI	Solución Integral de Facturación Electrónica e Informática SIFEI S. A. de C. V.
SLA	<i>Service Level Agreement</i> , Acuerdo de Niveles de Servicios



1.6 Términos y definiciones técnicas

Acrónimo	Definición
Biblioteca (Library)	Es una colección de recursos e implementaciones funcionales codificadas que son utilizadas por programas, en ellas se incluyen datos, configuraciones y código (generalmente pre compilado). Una biblioteca ofrece interfaces por las cuales un programa puede invocar funcionalidades.
DLL	Es una biblioteca (de enlace dinámico) que contiene código y datos que pueden utilizarse por varios programas al mismo tiempo. Las DLL son ligadas dinámicamente al programa que los usa durante su ejecución en vez de ser compiladas con el programa principal.
Microsoft Visual Studio (VS)	Es un entorno de desarrollo integrado que soporta múltiples lenguajes de programación, tales como C++, C#, Visual Basic .NET, F#, Java, Python, Ruby y PHP, al igual que entornos de desarrollo web, como ASP.NET MVC.
MariaDB Server	Es un potente servidor de base de datos derivado de MySQL y por tanto comparte muchas características de interfaces, comandos, sentencias y nomenclatura de programas.

1.7 Referencias

Información	Fuente
Formato de factura (Anexo 20)	http://www.sat.gob.mx/informacion_fiscal/factura_electronica/Paginas/Anexo_20_version3.3.aspx

1.8 Recursos

Los recursos listados a continuación son necesarios para poder emitir y timbrar comprobantes.

Archivo/Carpeta	Descripción
SifeiSignature3_3.dll	Biblioteca principal la cual debe ser referenciada en el proyecto para su uso e integración.
SAT	Carpeta que contiene los XSLT y XSD del SAT, además de incluir openssl.
bd_catálogoSAT.sql	Base de datos que contiene los catálogos del SAT.
MySQL.Data.dll	Biblioteca para el acceso a datos a MySQL y MariaDB.
zxing.dll	Biblioteca para el procesamiento de imágenes.
ICSharpCode.SharpZipLib	Biblioteca para que permite varios formatos de compresión como ZIP y TAR entre otros.

1.9 Versiones de CFDI que se timbran

SIFEI Signature 3.3 permite:

- Comprobante Fiscal Digital por Internet versión 3.3

1.9.1 Métodos de SIFEI Signature

- Emitir CFDI 3.3
- Emite CFDI 3.3 Comercio Exterior 1.1
- Emite CFDI 3.3 Nómina 1.2
- Emite CFDI 3.3 Detallista
- Emite CFDI 3.3 Pagos
- Recupera Timbre



- Cancela CFDI

Además, se pueden emitir facturas con complementos:

- Impuestos Locales
- Instituciones Educativas Privadas

2. Convenciones de documentación

- Los tipos de clase, clase, y enumeraciones están en color azul claro.
- Los tipos de datos primitivos están en azul marino.
- Cuando en la descripción de un método se haga referencia a un parámetro se le antepondrá el símbolo '@' al nombre del parámetro para su fácil reconocimiento.
- Los comentarios en las firmas de métodos y constructores van en color verde.
- Los métodos que alteren un nodo, en la sección *descripción* existirá el apartado *XPath* con la ruta del o los nodos modificados.
- Un nodo de tipo any, es un nodo de cualquier tipo, es decir, no hay restricciones como Addenda.
- A los parámetros que son opcionales (definidos así por el SAT), se les coloca el operador de nulificación '?' para pasar null a tipos de datos primitivos como decimal. Ejemplo: "`decimal? descuento`" donde descuento es un atributo opcional y el comprobante no tiene un descuento se debe pasar `null`.
- Cada ejemplo estará encerrado en un rectángulo.
- Para representar los comandos, se usará el nombre de la herramienta de línea de comandos seguido del símbolo mayor que ">", por ejemplo, si se está en la línea de comandos de Windows: "`shell>`", si se está dentro de MariaDB "`MariaDB>`", esto para dar un contexto explícito de donde debe ejecutarse el comando.

2.1 Atributos opcionales

Los atributos opcionales están representados con el operador "?", para el caso de las cadenas si se pasa como argumento `null`, o "" -cadena vacía- no se incluirá en el comprobante dicho atributo. Para el caso de los decimales, si se pasa `null`, tampoco se incluirá dicho atributo.

```
public void setTotalImpuestos(
    decimal? pTotImpRetenidos,
    decimal? pTotImpTrasladados
);
```

Crea y establece un nodo Impuestos (ver XPath abajo) como hijo directo al comprobante, los parametros @pTotImpRetenidos y @pTotImpTrasladados son opcionales, pero si se pasan deben cumplir con el tipo `t_Importe`, en caso contrario se lanza una excepción.

XPath:
 Cfdi:Impuestos

Comentado [CM3]: @pTotImpRetenidos y @pTotImpTrasladados son opcionales ya que llevan el operador de nulificación ? en su tipo primitivo



3. Primeros pasos (*getting started*)

El primer paso es crear nuestro proyecto, el cual para este ejemplo será de tipo *WinForms*:

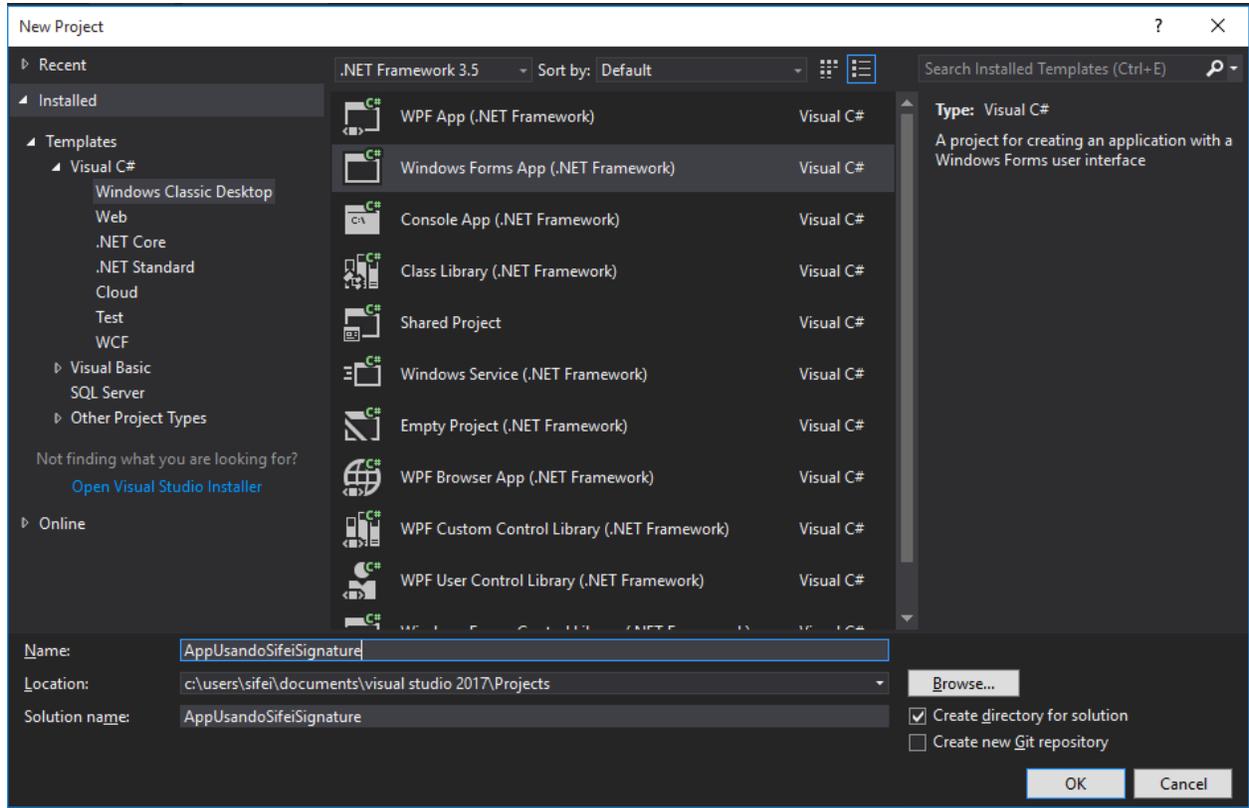


Imagen 1 Creación de proyecto .Net Framework 3.5

Lo cual debe generar algo similar a la siguiente imagen:

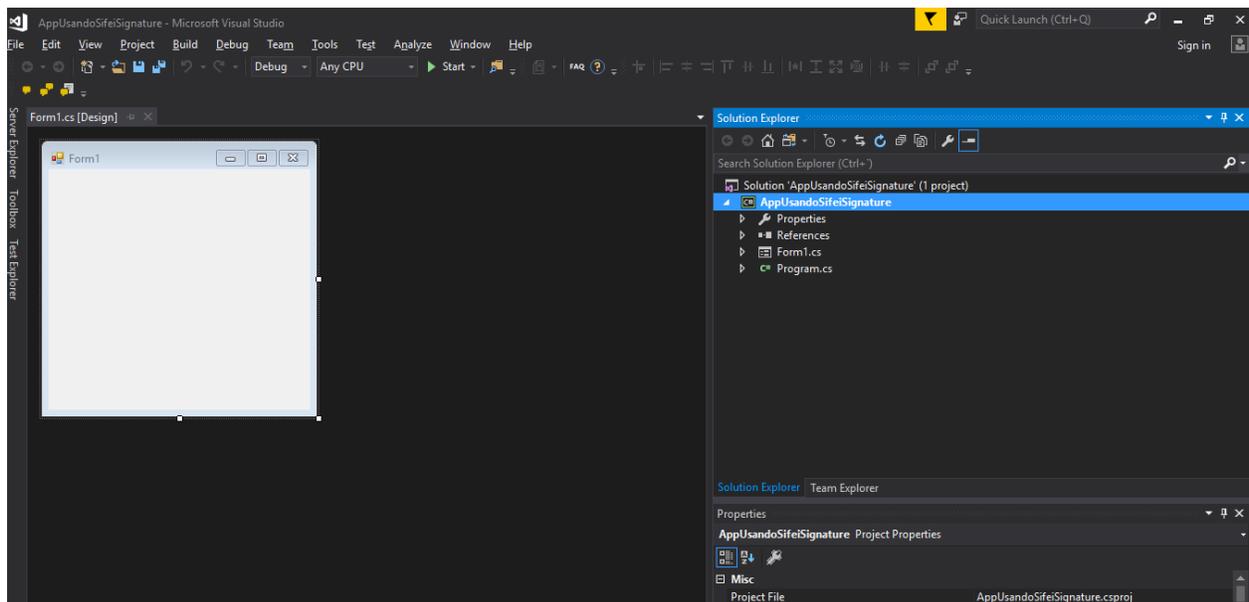


Imagen 2 Proyecto recién creado

3.1 Agregar recursos

Es necesario que los recursos estén en la misma carpeta que el programa (ejecutable del proyecto), para hacerlo debemos seguir los siguientes pasos:

1. Copiar la carpeta SAT y d11 en la misma carpeta que el proyecto
2. Incluir la carpeta SAT y d11 en el proyecto
3. Seleccionar todos los archivos y configurar para que sean copiados en el directorio de salida.

3.1.1 Copiar la carpeta SAT y las librerías

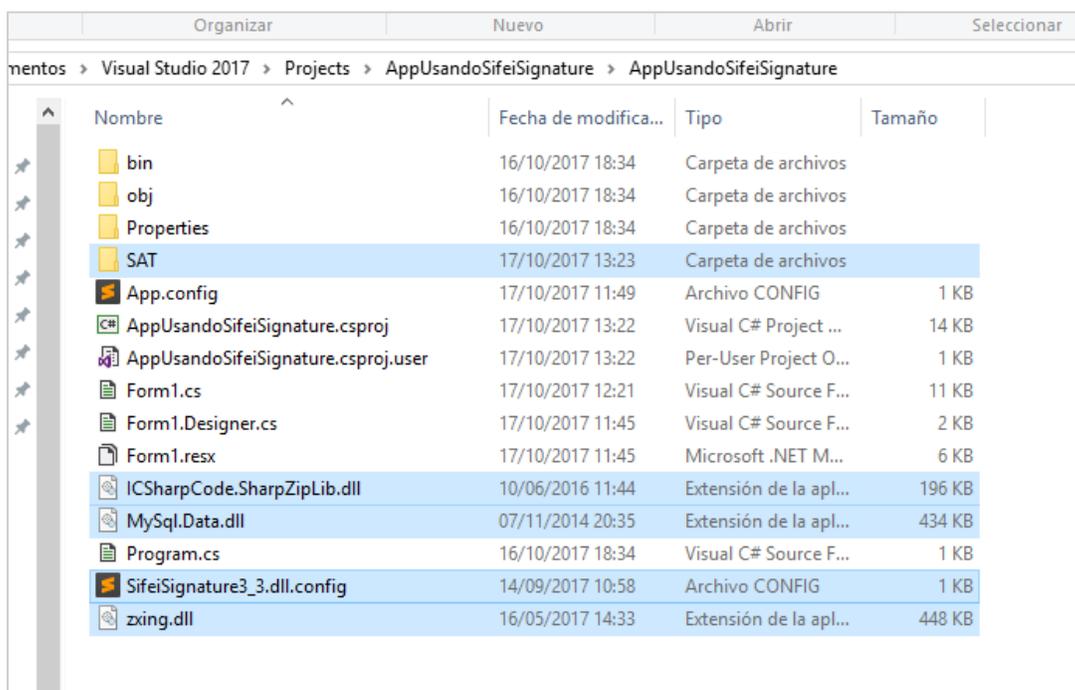


Imagen 3 Copiando los recursos al proyecto

3.1.2 Incluir la carpeta SAT y d11 en el proyecto

Por defecto los archivos no serán visibles, así que se debe presionar el icono “Mostrar todos los archivos” y después seleccionar Imagen 1 la carpeta SAT (Imagen 4) y los d11 para ser incluidos en el proyecto (Imagen 5).



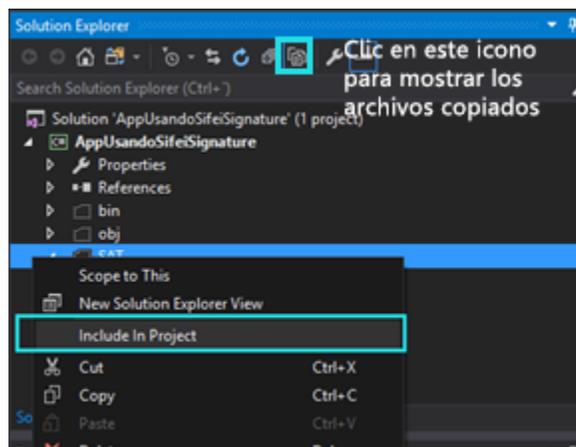


Imagen 4 Inclusión de carpeta SAT en el proyecto

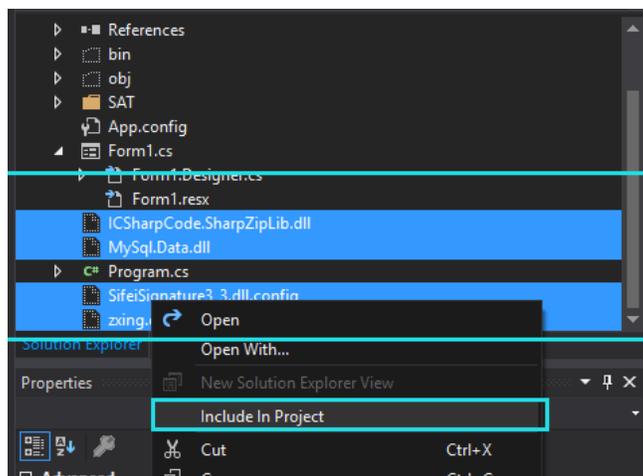


Imagen 5 Inclusión del resto de bibliotecas

3.1.3 Configurar los archivos para ser copiados en el directorio de salida

A pesar de haber incluido los recursos al proyecto, estos no son copiados al momento de compilación, por ello se deben de seleccionar e indicar la propiedad "Copiar al directorio de Salida" el valor "Copiar siempre", esto debe hacerse para los XSLT, XSD y openssl.



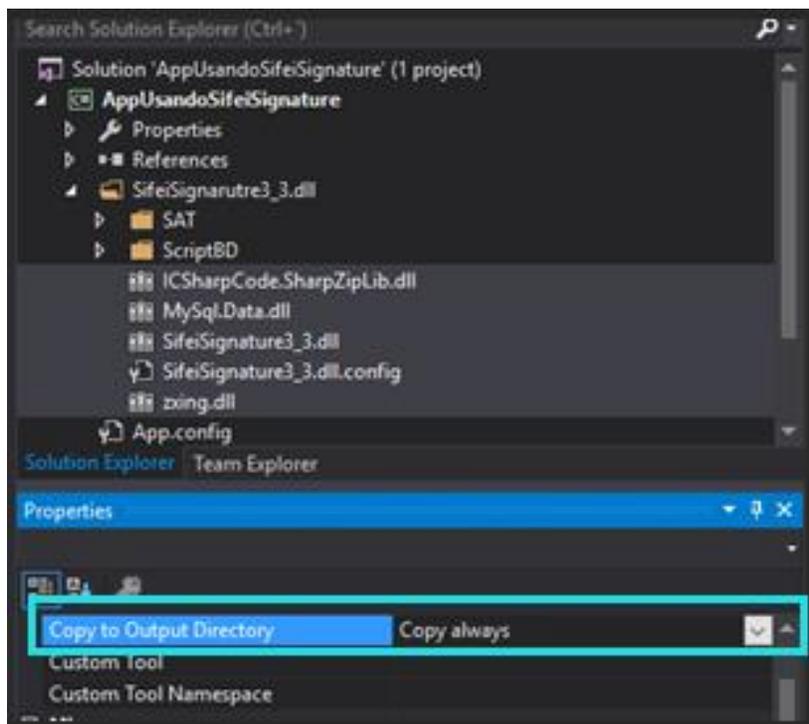


Imagen 6 Modificando la propiedad "Copiar al directorio de Salida"

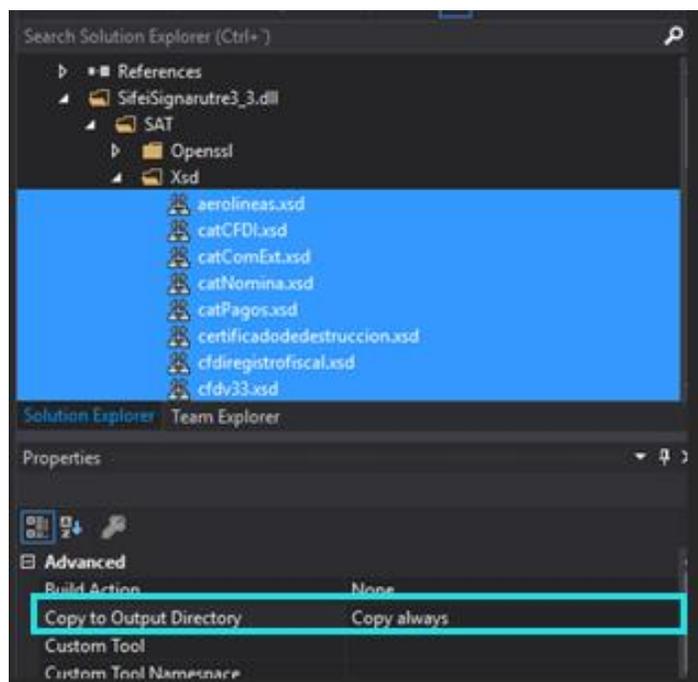
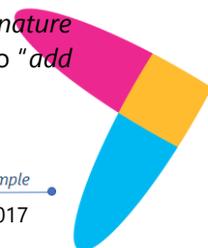


Imagen 7 Modificando la propiedad "Copiar al directorio de salida" a "Copiar siempre"

3.2 Agregar la biblioteca al proyecto

El método para agregar una DLL a un proyecto es a través de Referencias, para agregar *SIFEI Signature* primero se debe desplegar el menú secundario en el proyecto y seleccionar en "agregar referencia" o "add reference" (si Visual Studio está en idioma inglés).



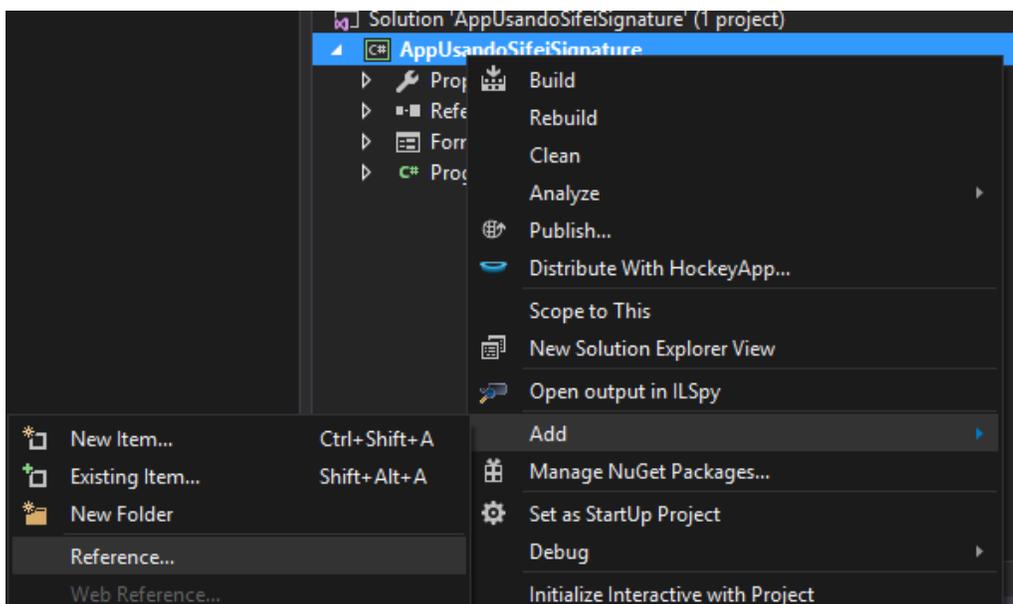


Imagen 8 Agregar referencia al proyecto

Se debe seleccionar la biblioteca "SifeiSignature3_3.dll" y agregar.

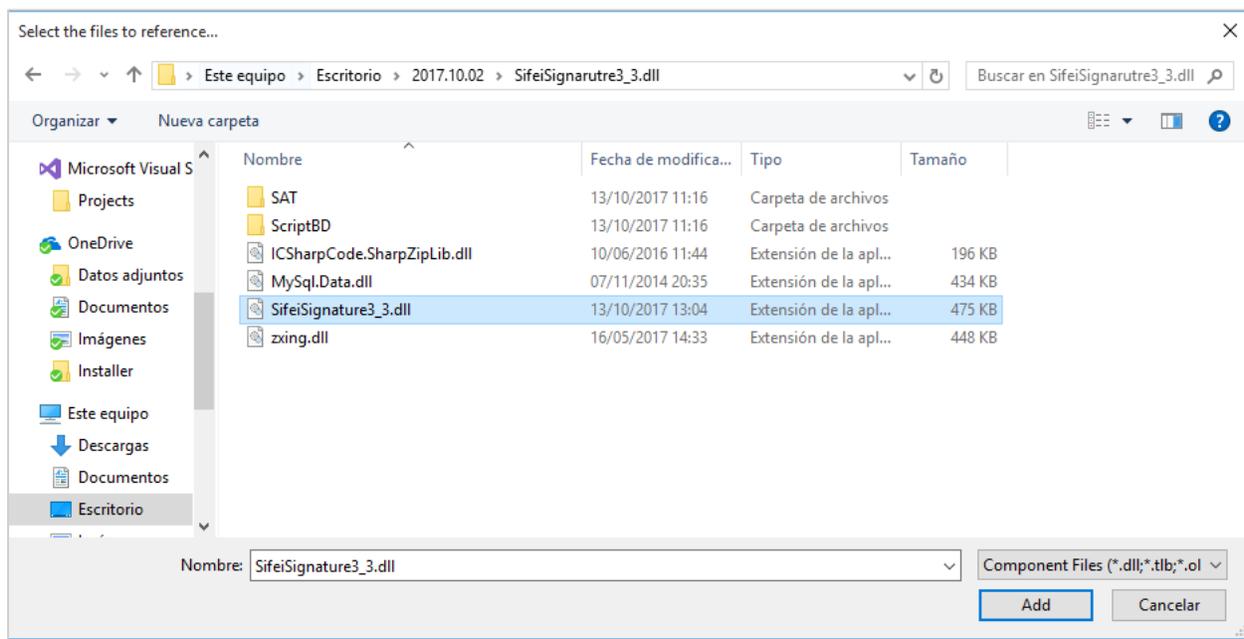


Imagen 9 Carpeta donde está SifeiSignature



Se recomienda que la ruta sea relativa al proyecto, es decir, que esté dentro de la carpeta del proyecto, esto en caso de que copie el área de trabajo.

SifeiSignature3_3.dll tiene dependencias sobre otras librerías y recursos para funcionar, es por ello que los demás archivos dll y carpeta SAT deben estar localizados en la misma carpeta y nivel que SifeiSignature3_3.dll al momento de ejecución y como se muestra en la imagen

Ya marcada la biblioteca, se debe dar clic en *Aceptar* –o en *Ok* si VS está en idioma inglés–.

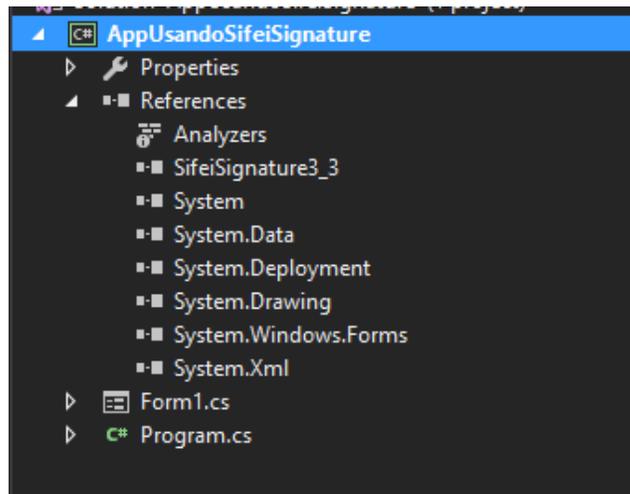


Imagen 10 Referencia de biblioteca agregada

3.3 Agregar información del servicio web

Además, es necesario el siguiente nodo "system.serviceModel" al archivo de configuración App.config (como hijo de configuration).

```
<system.serviceModel>
  <bindings>
    <basicHttpBinding>
      <binding name="SIFEIPortBinding" />
    </basicHttpBinding>
  </bindings>
  <client>
    <endpoint address="http://192.168.16.251:8080/DESKTOP33/SIFEI"
      binding="basicHttpBinding" bindingConfiguration="SIFEIPortBinding"
      contract="srvSifeiTimbrador.SIFEI" name="SIFEIPort" />
  </client>
</system.serviceModel>
```

Imagen 11 Configuración de aplicación requerida para SIFEI Signature

Para hacerlo se debe crear el archivo de configuración (si no existe):

- Seleccionar el proyecto (no a la solución)
- Dar clic secundario y selecciona "Agregar" -> "nuevo elemento"



- Seleccionar en *Archivo de configuración de aplicación* y dar clic en "Agregar".

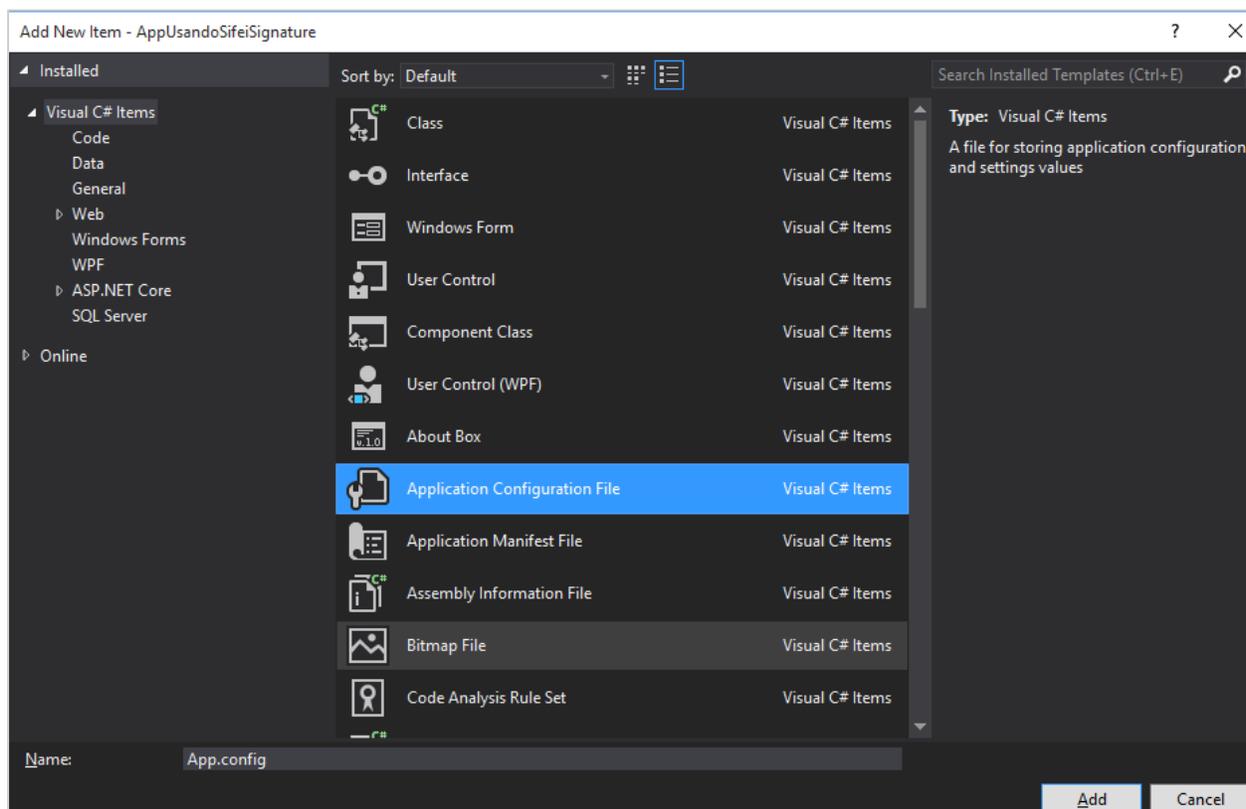


Imagen 12 Seleccionado Archivo de Configuración de Aplicación

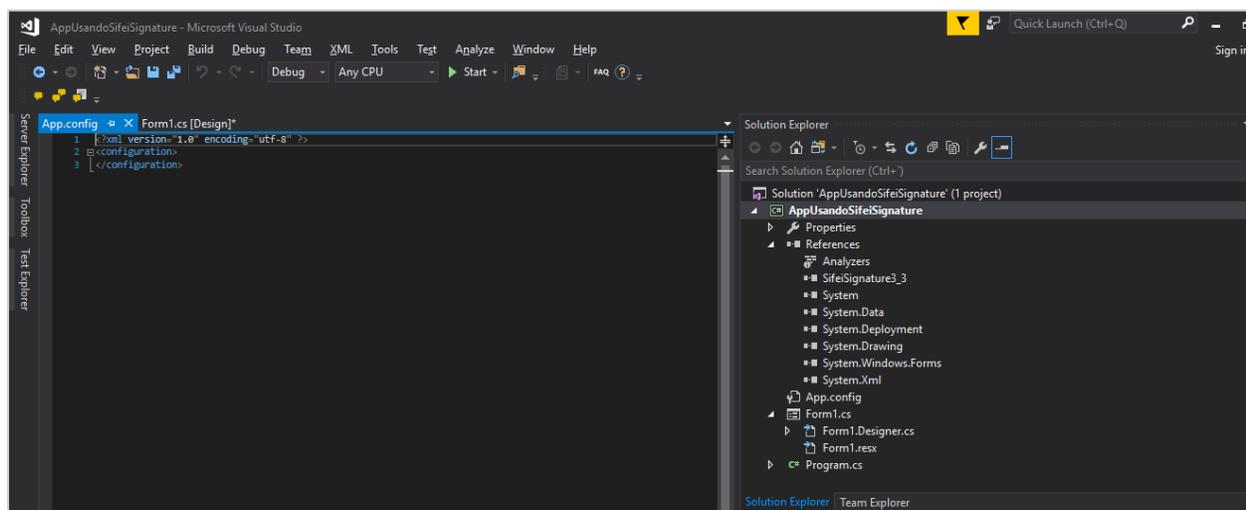
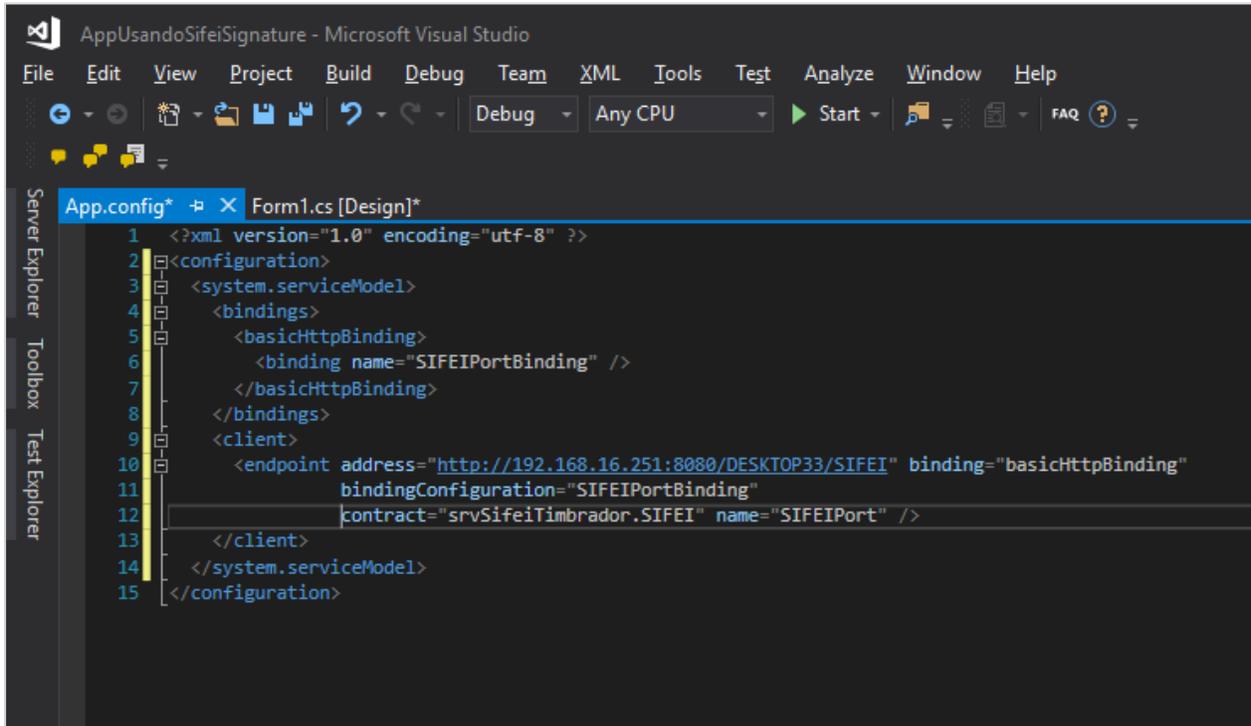


Imagen 13 Archivo de configuración limpio

Copiar el `system.servicemodel`.

Para resultar en algo similar a como se muestra en la Imagen 14.





3.4 Crear base de datos

Dentro de los recursos ofrecidos por SIFEI Signature se incluye un script con los catálogos del SAT listo para ser usado en MySQL o MariaDB, el nombre del script es "bd_catálogosat.sql".

Existen varias formas de ejecutar el script, la forma que se cubrirá en este apartado es a través de la herramienta de línea de comando de MySQL suponiendo que el gestor está instalado de forma local y que la carpeta bin de MariaDB está definida en las variables de entorno. Además, la cuenta de MariaDB debe tener privilegios para crear base de datos y tablas.

Nota: Para ver los requisitos ir al apartado requerimientos.

3.4.1 Comandos usados para crear la base de datos

```
Shell> mysql -U root -p catalogosat< bd_catalogossat.sql
```

Imagen 15 Ejecuta el script para la creación de la base de datos

```
MariaDb> show variables where variable_name='max_allowed_packet';
```

Imagen 16 Consulta para mostrar el valor de max_allowed_packet



3.4.2 Cambiar max_allowed_packet (Opcional)

Este paso es opcional y debe realizarse cuando la variable `max_allowed_packet` sea menor a 8mb y aparezca el siguiente error "Error: 2006", ya que el *script* cuenta con sentencias que superan este límite de tamaño de paquete.

```
C:\Users\sifei>mysql -uroot <C:\Workspace\SifeiSignature\bd_catalogossat.sql  
ERROR 2006 (HY000) at line 86: MySQL server has gone away
```

Imagen 17 Error 2006 MySQL server has gone away

```
MariaDB [(none)]> show variables where variable_name='max_allowed_packet';  
+-----+-----+  
| Variable_name | Value |  
+-----+-----+  
| max_allowed_packet | 1048576 |  
+-----+-----+  
1 row in set (0.00 sec)  
  
MariaDB [(none)]>
```

Imagen 18 Variable `max_allowed_packet` = 1MB: causará "Error 2006: MySQL server has gone away"

La forma más fácil y **temporal** para hacer el cambio es ejecutar el comando inferior, mismo que tendrá efecto a partir de la **siguiente conexión** (no la actual) y solo hasta que el **servicio sea reiniciado**:

```
MariaDb>set global max_allowed_packet=32*1024*1024;
```

Imagen 19 Comando para cambiar de forma global el límite que define `max_allowed_packet` a 32 MB



```

MariaDB [(none)]> set global max_allowed_packet=32*1024*1024;
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> show variables where variable_name='max_allowed_packet';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| max_allowed_packet | 1048576 |
+-----+-----+
1 row in set (0.00 sec)
    
```

NO cambio para esta sesion el valor de la variable

```

MariaDB [(none)]> Ctrl-C -- exit!
Bye

C:\Users\sifei>mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 16
Server version: 10.1.25-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show variables where variable_name='max_allowed_packet';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| max_allowed_packet | 33554432 |
+-----+-----+
1 row in set (0.00 sec)
    
```

Si salimos de la sesion y entramos de nuevo el cambio habra surtido efecto

Imagen 20 Cambiando variable Max_allowed_packet temporalmente (este efecto no cambia la sesión actual y dura solo mientras el servicio esté activo -hasta que el servicio sea reiniciado-)

Para hacerlo de forma permanente se debe cambiar el archivo de configuración de MariaDB:

```

-# The MySQL server
] [mysqld]
port= 3306
socket = "C:/xampp5/mysql/mysql.sock"
basedir = "C:/xampp5/mysql"
tmpdir = "C:/xampp5/tmp"
datadir = "C:/xampp5/mysql/data"
pid_file = "mysql.pid"
# enable-named-pipe
key_buffer = 16M
max_allowed_packet = 1M
sort_buffer_size = 512K
net_buffer_length = 8K
read_buffer_size = 256K
read_rnd_buffer_size = 512K
myisam_sort_buffer_size = 8M
log_error = "mysql_error.log"
    
```

```

-# The MySQL server
] [mysqld]
port= 3306
socket = "C:/xampp5/mysql/mysql.sock"
basedir = "C:/xampp5/mysql"
tmpdir = "C:/xampp5/tmp"
datadir = "C:/xampp5/mysql/data"
pid_file = "mysql.pid"
# enable-named-pipe
key_buffer = 16M
max_allowed_packet = 16M
sort_buffer_size = 512K
net_buffer_length = 8K
read_buffer_size = 256K
read_rnd_buffer_size = 512K
myisam_sort_buffer_size = 8M
log_error = "mysql_error.log"
    
```

Imagen 21 Archivo de configuración del servidor de MariaDB, cambio de 1mb a 16 MB (se debe reiniciar el servicio)



3.4.3 Ejecutar el *script*

```
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| phpmyadmin |
| test |
+-----+
5 rows in set (0.00 sec)

MariaDB [(none)]> set global max_allowed_packet=32*1024*1024;
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> ByeCtrl-C -- exit!

C:\Users\sifei>mysql -uroot <C:\Workspace\SifeiSignature\bd_catalogossat.sql

C:\Users\sifei>mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 23
Server version: 10.1.25-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| catalogossat |
| information_schema |
| mysql |
| performance_schema |
| phpmyadmin |
+-----+
```

Imagen 22 Procedimiento para la ejecución del *script*, comenzando por el listado de la base de datos, la ejecución del *script* y finalizando por la comprobación de su creación



4. Ejemplos

A continuación, se presentan los ejemplos para emitir, sellar, timbrar y cancelar CFDIs además de recuperar timbres. Para mayor detalle sobre las clases, métodos y enumeraciones ver la sección 5 Referencia.

4.1 Instanciar un Objeto tipo CFDI33

La clase [CFDI33](#) es la clase principal para comenzar a emitir CFDIs, en este ejemplo se le pasa como primer argumento la cadena de conexión a la base de datos CatálogoSAT, el resto de los argumentos son los atributos que poseerá el CFDI.

```

01 cfdi33 = new CFDI33(
02 "Server=localhost;Port=3306;Database=catálogoSAT;Uid=root;pwd=123456;", //connString
03 "DEMO", //pSerie
04 "6", //pFolio
05 DateTime.Now, //pFecha
06 "02", //pFormaPago
07 "30 DIAS", //pCondPago
08 2228.80m, //pSubtotal
09 220.51m, //pDescuento
10 "MXN", //pMoneda
11 1m, //pTipoCambio
12 2329.62m, //pTotal
13 "I", //pTipoCompr
14 "PUE", //pMetodoPago
15 "76246", //pLugarEped
16 null, //pConfirmacion
17
18 @"C:\Workspace\DatosBuzonMariaDB\Cert\DEM020170725123153.cer"//rutaCert
19 );
    
```

Lo cual produce en el XML lo siguiente:

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <cfdi:Comprobante xmlns:cfdi="http://www.sat.gob.mx/cfd/3"
3 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4 xsi:schemaLocation="http://www.sat.gob.mx/cfd/3 http://www.sat.gob.mx/sitio_internet/cfd/3/cfdv33.xsd"
5 Version="3.3"
6 Serie="DEMO"
7 Folio="6"
8 Fecha="2017-10-18T16:25:14"
9 Sello=""
10 FormaPago="02"
11 NoCertificado="20001000000300022756"
12 Certificado="MIB5LT6fQyzWhdYKAZ/pdMHS4gN24v5rC3ad9AIGDVoMNSb2gDeL1m+1L..."
13 CondicionesDePago="30 DIAS"
14 SubTotal="2228.80"
15 Descuento="220.51"
16 Moneda="MXN"
17 TipoCambio="1"
18 Total="2329.62"
19 TipoDeComprobante="I"
20 MetodoPago="PUE"
21 LugarExpedicion="76246">
    
```

Imagen 23 Atributos de CFDI agregados



4.2 Establecer CfdiRelacionados

Utilizando la instancia de [CFDI33](#) anterior invocamos el método "setCfdirelacionados()" y le pasamos el tipo de relación y el UUID del CFDI Relacionado, lo cual genera un primer CFDI Relacionado, con el UUID indicado.

```
01 //Estableciendo el nodo CfdiRelacionados
02 cfdi33.setCfdirelacionados("04", //pTipoRelacion
03 "2214C542-2FE6-45C5-BDA0-4C8EFF94CCFE" //pUUID
04 );
```

Lo cual produce el nodo CfdiRelacionados y un nodo CfdiRelacionado (ya que si existe CfdiRelacionados debe existir al menos un CfdiRelacionado):

```
22 <cfdi:CfdiRelacionados TipoRelacion="04">
23   <cfdi:CfdiRelacionado UUID="2214C542-2FE6-45C5-BDA0-4C8EFF94CCFE" />
24 </cfdi:CfdiRelacionados>
```

Imagen 24 Nodo CfdiRelacionados agregado

4.3 Agregar un CfdiRelacionado

Si se desea agregar un CfdiRelacionado adicional basta con invocar al método "addCfdiRelacionado()".

```
01 //agregando un CfdiRelacionado a CfdiRelacionados
02 cfdi33.addCfdiRelacionado(
03 "806EA594-A850-48C3-B56E-1EB0C4D570C8" //pUUID
04 );
```

El resultado sería (incluyendo el paso anterior) la línea 24 (ver rojo).

```
22 <cfdi:CfdiRelacionados TipoRelacion="04">
23   <cfdi:CfdiRelacionado UUID="2214C542-2FE6-45C5-BDA0-4C8EFF94CCFE" />
24   <cfdi:CfdiRelacionado UUID="806EA594-A850-48C3-B56E-1EB0C4D570C8" />
25 </cfdi:CfdiRelacionados>
```

Imagen 25 CFDIRelacionado agregado

4.4 Agregar el Emisor

Para agregar el emisor del CFDI se debe invocar al método setEmisor().

```
01 // Estableciendo el Emisor
02 cfdi33.setEmisor("MAR980114GQA", //pRfc
03 "UNION NACIONAL DE RESINEROS SA DE CV", //pNombre
04 "601"
05 );
```

El nodo agregado es Emisor línea 26 (ver verde):

```
22 <cfdi:CfdiRelacionados TipoRelacion="04">
23   <cfdi:CfdiRelacionado UUID="2214C542-2FE6-45C5-BDA0-4C8EFF94CCFE" />
24   <cfdi:CfdiRelacionado UUID="806EA594-A850-48C3-B56E-1EB0C4D570C8" />
25 </cfdi:CfdiRelacionados>
26 <cfdi:Emisor Rfc="MAR980114GQA" Nombre="UNION NACIONAL DE RESINEROS SA DE CV" RegimenFiscal="601" />
```

Imagen 26 Nodo Emisor agregado

4.5 Agregar el Receptor

Para agregar el Receptor del CFDI se debe invocar al método setReceptor().

```

01 // Estableciendo el Receptor
02         cfdi33.setReceptor("CTR801211M15",           //pRfc
03                             "MARGARITA RODRIGUEZ TELLEZ", //pNombre
04                             null,                       //pResidenciaFiscal
05                             null,                       //pNumRegIdTrib
06                             "G01"                      //pUsoCfdi
07     );
  
```

El nodo agregado es (ver naranja, línea 27), el cual tiene los atributos con los valores que se les paso al método setReceptor():

```

22 <cfdi:CfdiRelacionados TipoRelacion="04">
23   <cfdi:CfdiRelacionado UUID="2214C542-2FE6-45C5-BDA0-4C8EFF94CCFE" />
24   <cfdi:CfdiRelacionado UUID="806EA594-A850-48C3-B56E-1EB0C4D570C8" />
25 </cfdi:CfdiRelacionados>
26 <cfdi:Emisor Rfc="MAR980114GQA" Nombre="UNION NACIONAL DE RESINEROS SA DE CV" RegimenFiscal="601" />
27 <cfdi:Receptor Rfc="CTR801211M15" Nombre="MARGARITA RODRIGUEZ TELLEZ" UsoCFDI="G01" />
  
```

Imagen 27 Nodo Receptor agregado

4.6 Agregar un concepto

Para agregar un concepto se debe invocar al método addConcepto() el cual devuelve un entero con el índice del concepto que se creó, **mismo que servirá para en posteriores métodos para agregar información adicional al concepto (como impuestos, información aduanera, etc.)**.

El parámetro @indexConc es de tipo entero y recibe el índice del nodo concepto.

```

01 indexConc = cfdi33.addConcepto("01010101",           //pCveProdServ
02                                 null,                 //pNoIdentif
03                                 8m,                    //pCantidad
04                                 "H87",                 //pCveUnidad
05                                 "PZA",                 //pUnidad
06                                 "CALCIFLAVON (Tablets)", //pDescripcion
07                                 278.60m,              //pValorUnit
08                                 2228.80m,             //pImporte
09                                 220.51m,              //pDescuento
10                                 null);                 //pNumCuentaPredial
  
```

El nodo agregado es concepto (en morado) con los atributos que se le paso al método:

```

22 <cfdi:CfdiRelacionados TipoRelacion="04">
23   <cfdi:CfdiRelacionado UUID="2214C542-2FE6-45C5-BDA0-4C8EFF94CCFE" />
24   <cfdi:CfdiRelacionado UUID="806EA594-A850-48C3-B56E-1EB0C4D570C8" />
25 </cfdi:CfdiRelacionados>
26 <cfdi:Emisor Rfc="MAR980114GQA" Nombre="UNION NACIONAL DE RESINEROS SA DE CV" RegimenFiscal="601" />
27 <cfdi:Receptor Rfc="CTR801211M15" Nombre="MARGARITA RODRIGUEZ TELLEZ" UsoCFDI="G01" />
28 <cfdi:Conceptos>
29   <cfdi:Concepto ClaveProdServ="01010101" Cantidad="8" ClaveUnidad="H87"
30     Unidad="PZA" Descripcion="CALCIFLAVON (Tablets)" ValorUnitario="278.60"
31     Importe="2228.80" Descuento="220.51">
32 </cfdi:Concepto>
33 </cfdi:Conceptos>
  
```

Imagen 28 Nodo Concepto agregado



4.7 Agregar Información aduanera a un concepto

Cuando se ha definido un concepto al CFDI, es posible agregar "información aduanera", para ello es necesario indicar el índice (posición) del concepto y pasar los parámetros al método `addInformacionAduaneraToConcepto()`;

```

01 // Agregando Información aduanera al concepto con posición indexConc,
02     cfdi33.addInformacionAduaneraToConcepto(
03 indexConc, //pIndexConc seria= 1
04 "15 43 3740 6000564" //pNumPedimento
05 );
06
  
```

Modifica el nodo concepto apuntado por `@indexConc` agregándole como hijo el nodo información Aduanera (en amarillo):

```

28 <cfdi:Conceptos>
29   <cfdi:Concepto ClaveProdServ="01010101" Cantidad="8" ClaveUnidad="H87"
30     Unidad="PZA" Descripcion="CALCIFLAVON (Tablets)" ValorUnitario="278.60"
31     Importe="2228.80" Descuento="220.51">
32     <cfdi:InformacionAduanera NumeroPedimento="15 43 3740 6000564" />
33   </cfdi:Concepto>
34 </cfdi:Conceptos>
  
```

Imagen 29 InformacionAduanera agregada a conceptos

4.8 Agregar un impuesto traslado a un concepto

Cuando se ha definido un concepto al CFDi, es posible indicar los impuestos traslados por medio del método `addImpTrasladoToConcepto()`. El concepto se indica con el parámetro `@indexConc`.

```

01 // agregando impuesto traslado al concepto
02     cfdi33.addImpTrasladoToConcepto(indexConc, //pIndexConc
03     2008.29m, //pBase
04     "002", //pImpuesto
05     "Tasa", //pTipoFactor
06     0.16m, //pTasaOCuota
07     321.33m); //pImporte
  
```

Agrega el traslado (en azul) al concepto:

```

28 <cfdi:Conceptos>
29   <cfdi:Concepto ClaveProdServ="01010101" Cantidad="8" ClaveUnidad="H87"
30     Unidad="PZA" Descripcion="CALCIFLAVON (Tablets)" ValorUnitario="278.60"
31     Importe="2228.80" Descuento="220.51">
32     <cfdi:Impuestos>
33       <cfdi:Traslados>
34         <cfdi:Traslado Base="2008.29" Impuesto="002" TipoFactor="Tasa" TasaOCuota="0.16" Importe="321.33" />
35       </cfdi:Traslados>
36     </cfdi:Impuestos>
37     <cfdi:InformacionAduanera NumeroPedimento="15 43 3740 6000564" />
38   </cfdi:Concepto>
39 </cfdi:Conceptos>
  
```

Imagen 30 Resultado de agregar un impuesto traslado al concepto

4.9 Agregar total impuestos de comprobante

Definiendo el total impuestos con el método `setTotalImpuestos()`.



```

01 // estableciendo el total de impuestos
02         cfdi33.setTotalImpuestos(0.00m,           //pTotImpRetenidos
03                                     321.33m);     //pTotImpTrasladados
04
05
    
```

Lo cual agrega el nodo Impuestos (en color morado):

```

28 <cfdi:Conceptos>
29   <cfdi:Concepto ClaveProdServ="01010101" Cantidad="8" ClaveUnidad="H87"
30     Unidad="PZA" Descripcion="CALCIFLAVON (Tablets)" ValorUnitario="278.60"
31     Importe="2228.80" Descuento="220.51">
32     <cfdi:Impuestos>
33       <cfdi:Traslados>
34         <cfdi:Traslado Base="2008.29" Impuesto="002" TipoFactor="Tasa" TasaOCuota="0.16" Importe="321.33" />
35       </cfdi:Traslados>
36     </cfdi:Impuestos>
37   <cfdi:InformacionAduanera NumeroPedimento="15 43 3740 6000564" />
38 </cfdi:Concepto>
39 </cfdi:Conceptos>
40 <cfdi:Impuestos TotalImpuestosRetenidos="0.00" TotalImpuestosTrasladados="321.33">
41 </cfdi:Impuestos>
42 </cfdi:Comprobante>
    
```

Imagen 31 Nodo Impuestos agregado a comprobante

4.10 Agregar un impuesto traslado a impuestos de comprobante

Solo si se ha definido el nodo Impuestos al comprobante (Xpath: `cfdi:Comprobante/cfdi:Impuestos`), es posible indicar el impuesto Traslado por medio del método `addImpTraslado()`.

```

01 //estableciendo un impuesto traslado
02         cfdi33.addImpTraslado("002",           //pImpuesto
03                                 "Tasa",         //pTipoFactor
04                                 0.16m,         //pTasaCuota
05                                 321.33m);     //pImporte
    
```

Este agrega los nodos traslados y traslado al nodo "impuestos" (marcados en color azul)

```

40 <cfdi:Impuestos TotalImpuestosRetenidos="0.00" TotalImpuestosTrasladados="321.33">
41   <cfdi:Traslados>
42     <cfdi:Traslado Impuesto="002" TipoFactor="Tasa" TasaOCuota="0.16" Importe="321.33" />
43   </cfdi:Traslados>
44 </cfdi:Impuestos>
45 </cfdi:Comprobante>
    
```

Imagen 32 Impuesto traslado agregado

4.11 Generar el XML

Para generar el `XmlDocument` debemos invocar al método `generaXml()` y pasarle el `string` de conexión, el objeto `XmlDocument` no es retornado de forma directa, sino que es devuelto en un objeto `Wrapper` de tipo `ReturnXML`, este objeto nos permite averiguar si se generó el XML o no. (Para más información ver la referencia [ReturnXML](#)).

```

01 //generamos el XML, el objeto de tipo ReturnXML, contiene propiedades que indican si se creó
02 el XML, y si es así en su propiedad XML, el objeto XmlDocument
03 ReturnXML rtnCfdi = cfdi33.generaXml(ConnString);
04 //comprobamos que se ha creado con la propiedad IsXMLCreated
05 if (rtnCfdi.IsXMLCreated) {
06     // si se creó, obtenemos de la propiedad XML para guardarlo
07     rtnCfdi.XML.Save(@"C:\Workspace\SifeiSignature\CfdiGeneradosTest\cfdi_creado.xml");
08     if (!File.Exists(@"C:\Workspace\SifeiSignature\CfdiGeneradosTest\cfdi_creado.xml")) {
    
```

```

09         MessageBox.Show("No se escribió el archivo");
10     }
11 }
12
  
```

Tras seguir los ejemplos de código anteriores el XML producido es el siguiente:

```

<?xml version="1.0" encoding="utf-8"?>
<cfdi:Comprobante xmlns:cfdi="http://www.sat.gob.mx/cfd/3"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.sat.gob.mx/cfd/3
http://www.sat.gob.mx/sitio_internet/cfd/3/cfdv33.xsd"
Version="3.3"
Serie="DEMO"
Folio="6"
Fecha="2017-10-18T16:25:14"
Sello="@ "
FormaPago="02"
NoCertificado="20001000000300022756"

Certificado="MIb5LT6fQyzWhdYKAZ/pdMhs4gN24v5rC3ad9AlGDVoMNs b2gDeL1m+1L...."
CondicionesDePago="30 DIAS"
SubTotal="2228.80"
Descuento="220.51"
Moneda="MXN"
TipoCambio="1"
Total="2329.62"
TipoDeComprobante="I"
MetodoPago="PUE"
LugarExpedicion="76246">
<cfdi:CfdiRelacionados TipoRelacion="04">
<cfdi:CfdiRelacionado UUID="2214C542-2FE6-45C5-BDA0-4C8EFF94CCFE" />
<cfdi:CfdiRelacionado UUID="806EA594-A850-48C3-B56E-1EB0C4D570C8" />
</cfdi:CfdiRelacionados>
<cfdi:Emisor Rfc="MAR980114GQA" Nombre="UNION NACIONAL DE RESINEROS SA DE CV"
RegimenFiscal="601" />
<cfdi:Receptor Rfc="CTR801211M15" Nombre="MARGARITA RODRIGUEZ TELLEZ" UsoCFDI="G01" />
<cfdi:Conceptos>
<cfdi:Concepto ClaveProdServ="01010101" Cantidad="8" ClaveUnidad="H87"
Unidad="PZA" Descripcion="CALCIFLAVON (Tablets)" ValorUnitario="278.60"
Importe="2228.80" Descuento="220.51">
<cfdi:Impuestos>
<cfdi:Traslados>
<cfdi:Traslado Base="2008.29" Impuesto="002" TipoFactor="Tasa" TasaOCuota="0.16"
Importe="321.33" />
</cfdi:Traslados>
</cfdi:Impuestos>
<cfdi:InformacionAduanera NumeroPedimento="15 43 3740 6000564" />
</cfdi:Concepto>
</cfdi:Conceptos>
<cfdi:Impuestos TotalImpuestosRetenidos="0.00" TotalImpuestosTrasladados="321.33">
<cfdi:Traslados>
<cfdi:Traslado Impuesto="002" TipoFactor="Tasa" TasaOCuota="0.16" Importe="321.33" />
</cfdi:Traslados>
</cfdi:Impuestos>
</cfdi:Comprobante>
  
```

En caso de error se debe verificar la propiedad *ErrorDetalle* del objeto *rtnCfdi* de tipo *ReturnXML* (línea 07)



```

01 //
02 //codigo
03 //
04 if (rtnCfdi.IsXMLCreated) {
05 }
06 else {
07     MessageBox.Show("Error al sellar el CFDI: " +rtnCfdi.ErrorDetalle);
08 }
  
```

Para más saber sobre los valores que contiene la propiedad `rtnCfdi.ErrorDetalle` ver [Códigos de Error de la Matriz de validaciones para CFDI versión 3.3.](#)

4.12 Sellar el comprobante

Es posible sellar un comprobante a través del método `sellaComprobante()` el cual retorna un objeto tipo `ReturnXML`, mismo que nos sirve para saber si el comprobante fue sellado y recuperarlo, el código siguiente muestra como sellar el comprobante.

```

01 //sellamos el comprobante pasándole el path del certificado y de la llave
02 ReturnXML rtnCfdi = cfdi33.sellaComprobante(rtnCfdi.XML,
03     @"C:\Workspace\DatosBuzonMariaDB\Cert\DEM020170725123153.cer",
04     @"C:\Workspace\DatosBuzonMariaDB\Cert\DEM020170725123153.key",
05     "12345678a",
06     false,
07     true);
08
09 //comprobamos que se halla sellado
10 if (rtnCfdi.EstaSellado) {
11     //escribimos el Comprobante sellado:
12
13     rtnCfdi.XML.Save(@"C:\Workspace\SifeiSignature\CfdiGeneradosTest\cfdi_sellado.xml");
14 }
15
  
```

Comprobante sellado atributo sello (línea 5, marcado en púrpura):



```

1 <?xml version="1.0" encoding="utf-8"?>
2 <cfdi:Comprobante xmlns:cfdi="http://www.sat.gob.mx/cfd/3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3   xsi:schemaLocation="http://www.sat.gob.mx/cfd/3 http://www.sat.gob.mx/sitio_internet/cfd/3/cfdv33.xsd"
4   Version="3.3" Serie="DEMO" Folio="6" Fecha="2017-10-18T16:51:55"
5   Sello="f8rA2gcjnykVocsF17BIQPjOXIbFDd0UFA988EcAX1U+yu36hmIRlpDPElFt8AINZ7uaJVeDUu1HF0kqLiB9mx/mCwVTTDF1/1YZPk95YoCuErT
6   FormaPago="02" NoCertificado="2000100000300022756"
7   Certificado="MIIF/zCCA+egAwIBAgIUMjAwDEWMDAwMDAzMDAwMjI3NTYwDQYJKoZIhvcNAQELBQAwggFmMSAwHgYDVQ0DD8dBLkMuIDIgZGUGcH31Z
8   CondicionesDePago="30 DIAS"
9   SubTotal="2228.80" Descuento="220.51" Moneda="MXN" TipoCambio="1"
10  Total="2329.62" TipoDeComprobante="I" MetodoPago="PUE" LugarExpedicion="76246">
11   <cfdi:CfdiRelacionados TipoRelacion="04">
12     <cfdi:CfdiRelacionado UUID="2214C542-2FE6-45C5-BDA0-4C8EFF94CCFE" />
13     <cfdi:CfdiRelacionado UUID="806EA594-A850-48C3-B56E-1EB0C4D570C8" />
14   </cfdi:CfdiRelacionados>
15   <cfdi:Emisor Rfc="MAR980114GQA" Nombre="UNION NACIONAL DE RESINEROS SA DE CV" RegimenFiscal="601" />
16   <cfdi:Receptor Rfc="CTR801211M15" Nombre="MARGARITA RODRIGUEZ TELLEZ" UsoCFDI="601" />
17   <cfdi:Conceptos>
18     <cfdi:Concepto ClaveProdServ="01010101" Cantidad="8" ClaveUnidad="H87" Unidad="PZA" Descripcion="CALCIFLAVON (Tablets)"
19       ValorUnitario="278.60" Importe="2228.80" Descuento="220.51">
20       <cfdi:Impuestos>
21         <cfdi:Traslados>
22           <cfdi:Traslado Base="2008.29" Impuesto="002" TipoFactor="Tasa" TasaOCuota="0.16" Importe="321.33" />
23         </cfdi:Traslados>
24       </cfdi:Impuestos>
25       <cfdi:InformacionAduanera NumeroPedimento="15 43 3740 6000564" />
26     </cfdi:Concepto>
27   </cfdi:Conceptos>
28   <cfdi:Impuestos TotalImpuestosRetenidos="0.00" TotalImpuestosTrasladados="321.33">
29     <cfdi:Traslados>
30       <cfdi:Traslado Impuesto="002" TipoFactor="Tasa" TasaOCuota="0.16" Importe="321.33" />
31     </cfdi:Traslados>
32   </cfdi:Impuestos>
33 </cfdi:Comprobante>
  
```

Imagen 33 Comprobante Sellado

4.13 Timbrar el comprobante

Es posible timbrar un comprobante a través del método `timbraComprobante()` el cual retorna un objeto tipo `ReturnTimbre`, mismo que sirve para saber si el comprobante fue timbrado y recuperarlo, tal y como se ve en el siguiente ejemplo:

```

01 //timbramos el comprobante
02 ReturnTimbre rtnTimbre = cfdi33.timbraComprobante("JGE080430LN8", "0e84ec92", "A",
03
04 "VPjftCMG1beV+/wkNwc8lkaFETxTrC/rt20oZgbnl30S6SD6kPAmLozt7CMkqzZK3L7+m1xZtXYDvu9oauJ1Wh
05 EWqSieNh96KCDnJdIXezM=",
06         rtnCfdi.XML, CFDI33.tipoTimbrado.Pruebas);
07         //verificamos que haya sido timbrado:
08     if (rtnTimbre.IsTimbrado) {
09         //escribimos el cfdi timbrado
10     rtnTimbre.Xml.Save(@"C:\Workspace\SifeiSignature\cfdi_timbrado.xml");
11
  
```

Al guardar el CFDI contendrá el nodo `TimbreFiscalDigital` (líneas 25 a 29 marcado en amarillo):



4.15.1 Establecer Emisor de Comercio Exterior

```

01 complCfdi.Cce11_setEmisor(
02     "", //pCurp
03     "1a. Privada de Oriente 17", //pDomCalle
04     "32", //pDomNumExt
05     null, //pDomNumInt
06     "2805", //pDomColonia
07     "12", //pDomLocalidad
08     "Entre Norte 2 y 4", //pDomReferencia
09     "118", //pDomMunicipio
10     "VER", //pDomEstado
11     "MEX", //pDomPais
12     "94300"); //pDomCP
    
```

4.15.2 Definir Receptor de Comercio Exterior

```

01 complCfdi.Cce11_setReceptor(
02     "", //pCurp
03     "Domicilio Conocido", //pDomCalle
04     "", //pDomNumExt
05     null, //pDomNumInt
06     "", //pDomColonia
07     "", //pDomLocalidad
08     "", //pDomReferencia
09     "MPIO", //pDomMunicipio
10     "CA", //pDomEstado
11     "USA", //pDomPais
12     "99999"); //pDomCP
    
```

4.15.3 Agregar mercancía de Comercio Exterior

```

01 ReturnXmlComplemento rtnXmlCompl = complCfdi.Cce11_generaComplemento();
02 Console.WriteLine(string.Format("{0} - Termina crear complemento
03 cce11", DateTime.Now)); //log
04 if (rtnXmlCompl.IsXmlComplementoCreated) {
05     cfdi33.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
06     cfdi33.addNamespace(complCfdi.NsComercioExterior1_1);
07 }
    
```

4.15.4 Generar el Complemento Comercio Exterior

```

01 // generamos el complemento
02 ReturnXmlComplemento rtnXmlCompl = complCfdi.Cce11_generaComplemento();
03 //verificamos que se haya creado.
04 if (rtnXmlCompl.IsXmlComplementoCreated) {
05     // le pasamos a al objeto CFDI33 el complemento y el namespace de comercio exterior
06     cfdi33.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
07     cfdi33.addNamespace(complCfdi.NsComercioExterior1_1);
08 }
09 // ahora para generar el comprobante, sellar y timbrar ver la clase CFDI33
10
11
    
```

Para generar el comprobante, sellar y timbrar ver la clase [CFDI33](#)



4.16 Definir complemento Nómina

```

01 //complCfdi es de tipo Complemento
02     complCfdi.Nomina12_DefineNomina(
03         "E", //pTipoNomina
04         new DateTime(2017, 01, 31), //pFechaPago
05         new DateTime(2017, 01, 31), //fechaInicPago
06         new DateTime(2017, 01, 31), //fechaFinPago
07         1m, //numDiasPag
08         75215.68m, //totPercepciones
09         3781.05m, //totDeducciones
10         null);
11
    
```

4.16.1 Agregar un emisor a Nómina

```

01 complCfdi.Nomina12_setEmisor(
02     null, //pCurp
03     "E52-19201-10-3", //pRegPatronal
04     null); //pRfcPatronOrigen
05
    
```

4.16.2 Agregar un receptor a Nómina

```

01 //establecemos el receptor
02     complCfdi.Nomina12_setReceptor(
03         "BAPJ570702HSLLLN02", //pCurp
04         "23755735414", //pNumSegSocial
05         new DateTime(2013, 12, 09), //pFechaInicRelLab
06         "P164W", //pAntiguedad
07         "01", //pTipoContrato
08         Complemento.Nomina12_pSiNoNoEspecificado.NoEspecificado,
09         //esSindicalizado
10         "01", //pTipoJornada
11         "02", //pTipoRegimen
12         "49", //pNumEmpleado
13         "1", //pDepartamento
14         "ENC ALMACEN DE MAQUI", //pPuesto
15         "5", //pRiesgo
16         "99", //pPeriodiciadPago
17         "072", //pBanco
18         null, //pCuentaBanco
19         210.00m, //pSalarioBaseCotApor
20         231.52m, //pSalarioDiario
21         "SIN"); //pEstado
22
    
```

4.16.3 Establecer que Nómina tiene percepciones

```

01     complCfdi.Nomina12_setPercepciones(
02         5054.74m, //pTotalSueldos
03         70160.94m, //pTotalSeparacionIndemn
04         null, //pTotalJubilacionPensionRetiro
05         50212.88m, //pTotalGravado
06         25002.80m); //pTotalExento
07
    
```



4.17.1 Establecer el requestForPaymentIdentification a Detallista

```
01 complCfdi.Det131_Agrega_RequestForPaymentIdentification_Detallista(
02     detallistaRequestForPaymentIdentificationEntityType.DEBIT_NOTE
03     //EntityType
04     );
05
```

4.17.2 Agregar un specialInstruction a Detallista

```
01 complCfdi.Det131_AgregaSpecialInstruction(
02     detallistaSpecialInstructionCode.AAB, //Code
03     "TEXT"); //text
04
```

4.17.3 Establecer el orderIdentification a Detallista

```
01 complCfdi.Det131_SetOrderIdentification(new DateTime(2016, 02, 07),
02     //referenceDate
03     "81283939"); //ReferenceIdentification
04
05
```

4.17.4 Agregar un referenceIdentification al AdditionalInformation de Detallista

```
01 complCfdi.Det131_AddReferenceIdentificationToAdditionalInformation(
02     detallistaReferenceIdentificationType.AAE, //tipo
03     "000001"); //valor
04
05
06
```

4.17.5 Establecer un DeliveryNote a Detallista

```
01 complCfdi.Det131_SetDeliveryNote(
02     new DateTime(2016, 04, 07), // reference date
03     "000001" //referenceIdIndetification
04     );
05
```

4.17.6 Definir el Buyer (comprador) de Detallista

```
01 complCfdi.Det131_SetBuyer(
02     "7504000107903", //gln
03     "0001" //personOrDeparmentName
04     );
05
```

4.17.7 Definir el Seller (vendedor) a Detallista

```
01 complCfdi.Det131_SetSeller(
02     "7244020982223", //gln
03     detallistaSellerAlternatePartyIdentificationType.IEPS_REFERENCE,
04     "12382317" //alternatePartyIdentification
05     );
06
```



4.17.8 Establecer el shipTo (lugar de envío) de Detallista

```

01 complCfdi.Det131_SetShipTo(
02     "8724098266218", //gln
03     "EMPRESA DEMO", //name
04     "Calle Peru num 029", //street address one
05     "Nuevo León", //city
06     "64590" //cp
07 );
    
```

4.17.9 Establecer el InvoiceCreator a Detallista

```

01 complCfdi.Det131_SetInvoiceCreator(
02     "5529910871321", //gln
03     detallistaInvoiceCreatorAlternatePartyIdentificationType.IA,
04     "1281282",
05     "EMPRESA DEMO", //name
06     "Calle Niños Heres num 1293", //street address one
07     "Nuevo León", //city
08     "64590"); //cp
09
    
```

4.17.10 Agregar un Customs a Detallista

```

01 complCfdi.Det131_AddCustomsGln("0129829934772");
02
03
    
```

4.17.11 Agregar un currency a Detallista

```

01 complCfdi.Det131_addCurrency(detallistaCurrencyCurrencyISOCODE.MXN,
02     detallistaCurrencyCurrencyFunction.PRICE_CURRENCY,
03     null,
04     null,
05     17.10m
06 );
07
    
```

4.17.12 Establecer el paymentTerms a Detallista

```

01
02 complCfdi.Det131_SetPaymentTerms(detallistaPaymentTermsPaymentTermsEvent.DATE_OF_INVOIC
03 E,
04     detallistaPaymentTermsPaymentTermsRelationTime.REFERENCE_AFTER,
05     detallistaPaymentTermsNetPaymentNetPaymentTermsType.BASIC_NET,
06     detallistaPaymentTermsNetPaymentPaymentTimePeriodTimePeriodDueTimePeriod.DAYS,
07     "10",
08     detallistaPaymentTermsDiscountPaymentDiscountType.ALLOWANCE_BY_PAYMENT_ON_TIME,
09     "0.00");
10
11
    
```

4.17.13 Establecer un allowanceCharge a Detallista para definir cargos o descuentos

```

01 complCfdi.Det131_AddAllowanceCharge(
02     detallistaAllowanceChargeAllowanceChargeType.ALLOWANCE_GLOBAL,
03     detallistaAllowanceChargeSettlementType.BILL_BACK,
04     "12717262", // numConsec
05     null, //detallista servetype
06     1.00m
07 );
    
```


4.18.1 Agregar un DoctoRelacionado a Pago

```

01 complCfdi.Pagos10_addDoctoRelacionadoToPago(
02   indexPago, //posicion del pago al cual agregarle el DoctoRelacionado
03   "BD837DE6-A09D-42D9-8849-FE576BD875C0", //IdDocumento
04   "Serie", //serie
05   "1", //folio
06   "USD",
07   null, //tipo de cambio
08   "PPD", //método de pago
09   5, //número de parcialidad
10   3222.34m,
11   3222.34m,
12   0m); //impuesto saldo Insoluto
  
```

4.18.2 Generar el Complemento Pago

```

01 //generamos el complemento pago
02   ReturnXmlComplemento rtnXmlCompl =
03   complCfdi.Pagos10_generaComplemento();
04   //verificamos que se haya creado.
05   if (rtnXmlCompl.IsXmlComplementoCreated) {
06     // agregamos el complemento al comprobante
07     cfdi33.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
08     //agregamos el namespace del complemento pago al comprobante
09     cfdi33.addNamespace(complCfdi.NsPagos10);
10   }
11
  
```

4.19 Definir y generar el complemento IEDU (a un concepto)

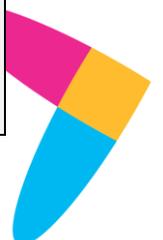
```

01 //debido a que el complemento IEDU es un solo nodo, el método Iedu10_generaComplemento
02 define y genera el complemento a la vez.
03 ReturnXmlComplemento rtnXmlCompl = complCfdi.Iedu10_generaComplemento(
04   "SOLIS HERRERA VALERIA", //nombre del alumno
05   "SOHV031025MHGLRLA4", //curp
06   instEducativasNivelEducativo.Primaria,
07   "13PPR0034X", //autRVOE
08   "AALE5706229F6" //rfcPago
09   );
10 //comprobamos que se creó el complemento
11 if (rtnXmlCompl.IsXmlComplementoCreated) {
12   //agregamos el complemento al concepto indicado por @indexConc
13   cfdi33.addComplementoConcepto(indexConc, rtnXmlCompl.XMLComplemento);
14   //agregamos el namespace de Iedu al comprobante
15   cfdi33.addNamespace(complCfdi.NsIedu10);
16 }
  
```

4.20 Recuperar timbre

```

01 //creamos el XmlDocument y cargamos el comprobante
02 System.Xml.XmlDocument cfdi = new System.Xml.XmlDocument();
03 cfdi.Load(@"C:\Workspace\comprobante.xml");
04 ReturnTimbre recuperado = CFDI33.recuperaTimbre(
05   "JGE080430LN8", //usuario
06   "0e84ec92", //password
07   cfdi, //objeto XmlDocument
08   CFDI33.tipoTimbrado.Pruebas
09   );
10 //verificamos si esta timbrado (si existe un timbre)
11 if (recuperado.IsTimbrado) {
  
```



```

12 //escribimos el comprobante
13 recuperado.Xml.Save(@"C:\Workspace\hash_cfdi_timbrado.xml");
14
15 // escribimos el código de barra bidimensional en un archivo jpg
16 ByteArrayToFile(@"C:\Workspace\hash_cbb.jpg", recuperado.Cbb);
17 Console.WriteLine(string.Format("{0} - Proceso finalizado", DateTime.Now));
18 }
19
  
```

4.21 Cancelar un CFDI

```

01 ReturnCancelacion rtnCanc = null;
02 Cancelacion sifCanc = new Cancelacion();
03 string msg = "";
04 List<string> uuids = new List<string>();
05 //el uuid del comprobante a cancelar
06 uuids.Add("B7A5628D-447C-4F10-BCD7-D1949CDF636C");
07 try {
08     rtnCanc = sifCanc.cancela(ref msg,
09         uuids.ToArray(), //arreglo de UUID a cancelar
10         "JGE080430LN8",
11         "0e84ec92",
12         "MAR980114GQA",
13         CERTIFICADO_PATH,
14         CERTIFICADO_PASS,
15         "12345678a", //pass del certificado
16         SifeiSignature3_3.Utills.Definiciones.TipoTimbrado.Pruebas
17     );
18     //verificamos que se haya cancelado
19     if (rtnCanc.IsCancelado) {
20         //GUARDAMOS EL ACUSE DE CANCELACIÓN
21         String acuse = rtnCanc.AcuseBE.AcuseXml;
22         XmlDocument xml = new XmlDocument();
23         xml.LoadXml(acuse);
24         xml.Save(@"C:\Workspace\xmlSaveTest\acuseC.xml");
25         MessageBox.Show("Cancelación terminada");
26     }
27     else {
28         MessageBox.Show("Error al cancelar: " + rtnCanc.ErrorDetalle);
29     }
30 }
31 catch (Exception ex) {
32     MessageBox.Show("Error al cancelar: " + ex.Message);
33 }
  
```

Para mayor detalle ir a la sección 5 Referencia y a la sección 5.15 Códigos de Respuesta de Cancelación.



5. Referencia

5.1 Namespaces

A continuación, se listan los *namespaces* de la biblioteca:

- Namespace [Cancelacion](#)
- Namespace [Complementos](#)
- Namespace [Returns](#)
- Namespace [Version3_3](#)
- Namespace [Timbre11](#)

5.2 Clases principales

Archivo	Descripción
CFDI33	Clase que representa el comprobante CFDI versión 3.3, en esta clase se agregan todos los valores que el comprobante a generar tendrá en sus atributos, así como sus respectivos nodos.
Complemento	Clase que representa al nodo complemento, en esta clase se contendrán todos los valores que el comprobante a generar tendrá en sus atributos, así como sus respectivos nodos.
ReturnXML	Clase que representa el resultado de generar el CFDI tras invocar al método CFDI33.generaXml() , por tanto, esta clase no contiene métodos. Si algún error ocurre al generar el comprobante cuando no es válido, esta clase contendrá los códigos de error y mensajes correspondientes a la matriz de errores, en caso de éxito contiene el comprobante.
ReturnTimbre	Clase que representa el resultado de obtener el timbre CFDI33.recuperaTimbre() , por tanto, no contiene métodos.
ReturnXmlComplemento	Clase que representa el resultado de generar un complemento a través del método Pagos10.generaComplemento de clase Complemento , por tanto, esta clase no tiene métodos.

5.3 Namespace SifeiSignature3_3.Version3_3

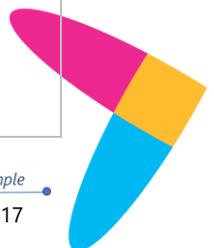
Clases:

- [CFDI33](#)

5.3.1 Clase CFDI33

5.3.1.1 Constructor

Constructor	Descripción
<pre>public CFDI33(string connString, string pSerie, string pFolio, DateTime pFecha, string pFormaPago, string pCondPago, decimal pSubtotal, decimal? pDescuento, string pMoneda, decimal? pTipoCambio,</pre>	Inicializa el objeto CFDI33 con los atributos principales del comprobante



Constructor	Descripción
<pre>decimal pTotal, string pTipoCompr, string pMetodoPago, string pLugarEped, string pConfirmacion, string rutaCert);</pre>	

5.3.2 Métodos

Método de CFDI33	Descripción
<pre>public static ReturnTimbre recuperaTimbre(string usuario, string password, XmlDocument xml, tipoTimbrado TipoTimbrado);</pre>	Recupera un timbre del servicio web de sifei, el cual si es correcto es devuelto en el objeto de tipo <code>ReturnTimbre</code> . Para más detalle ver la clase <code>ReturnTimbre</code> .
<pre>public void addAddenda(XmlElement pAddenda);</pre>	Agrega un nodo hijo addenda de tipo any (cualquiera) al nodo <code>addenda</code> del comprobante. XPath: <code>Cfdi:Addenda/cfdi:any</code>
<pre>public void addCfdiRelacionado(string pUUID);</pre>	Crea y agrega un nodo <code>CfdiRelacionado</code> con el UUID especificado. XPath: <code>Cfdi:CfdiRelacionados/cfdi:Cfdirelacionado</code> Excepciones: Exception Si el nodo <code>CfdiRelacionados</code> no ha sido establecido (es null). Si <code>@pUUID</code> no cumple con el formato y restricciones establecidas por el SAT.
<pre>public void addComplementoComprobante(XmlEle ment pComplemento);</pre>	Agrega un complemento al nodo complemento.
<pre>public bool addComplementoConcepto(int indexConcepto, XmlElement complemento);</pre>	Agrega el nodo <code>@complemento</code> como hijo al Concepto apuntado por <code>@indexConcepto</code> . XPath: <code>Cfdi:Conceptos/cfdi:concepto[@indexConcepto]/ComplementoC oncepto</code> Excepciones: Exception Si el concepto apuntado por <code>@indexConcepto</code> no existe.
<pre>public int addConcepto(string pCveProdServ, string pNoIdentif, decimal pCantidad, string pCveUnidad, string pUnidad, string pDescripcion,</pre>	Crea y agrega un nodo hijo <code>concepto</code> al nodo <code>conceptos</code> (<code>Cfdi:Conceptos</code>). XPath: <code>Cfdi:Conceptos/cfdi:concepto</code>



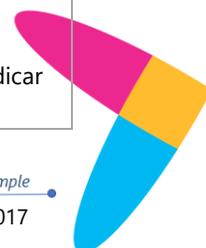
Método de CFDI33	Descripción
<pre>decimal pValorUnit, decimal pImporte, decimal? pDescuento, string pNumCuentaPredial);</pre>	<p>Returns Índice del nodo concepto que se agregó</p> <p>Excepciones: Exception Si los parámetros no cumplen el formato y restricciones definidas por el SAT</p> <p>Returns Índice del nodo Concepto</p>
<pre>public void addImpRetencion(string pImpuesto, decimal pImporte);</pre>	<p>Crea y agrega un nodo Retencion con el impuesto e importe indicado al nodo Impuestos del comprobante, verifica que exista el impuesto en el catálogo impuesto y que el importe corresponda con tipo importe (definido por el estándar del SAT).</p> <p>XPath: cfdi:Impuestos/cfdi:Retenciones/cfdi:Retencion</p> <p>Excepciones: Exception Si el nodo Impuestos no existe (es null), no ha sido definida. Lanza excepciones los parámetros no cumplen el formato y restricciones definidas por el SAT.</p> <p>ArgumentException Si alguno de los parámetros: @pImpuesto no existe en su catálogo.</p>
<pre>public void addImpRetencionToConcepto(int pIndexConc, decimal pBase, string pImpuesto, string pTipoFactor, decimal pTasaOCuota, decimal pImporte);</pre>	<p>Crea y agrega un nodo Retencion al Concepto indicado por el valor @pIndexConc, el nodo retencion tendrá los valores: pBase, pImpuesto, pTipoFactor, pTasaOCuota, pImporte.</p> <p>XPath: Cfdi:Conceptos/cfdi:concepto[@pIndexConc]/cfdi: Impuestos/cfdi:Retenciones/cfdi:Retencion</p> <p>Excepciones: Exception Si el concepto al que apunta @pIndexConc no existe (es null o está fuera de rango). Lanza excepciones los parámetros no cumplen el formato y restricciones definidas por el SAT.</p> <p>ArgumentException Si alguno de los parámetros: @pImpuesto, @pTipoFactor no existe en su catálogo.</p>
<pre>public void addImpTraslado(string pImpuesto, string pTipoFactor, decimal pTasaCuota, decimal pImporte);</pre>	<p>Crea y agrega un nodo Traslado con el @pImpuesto, @ptipofactor, @pTasaCuota e @pimporte indicado al nodo Impuestos del comprobante, verifica que exista el impuesto en el catálogo impuesto y que el importe corresponda con tipo importe (definido por el estándar del SAT).</p> <p>XPath: cfdi:Impuestos/cfdi:Traslados/cfdi:Traslado</p>



Método de CFDI33	Descripción
	<p>Excepciones: Exception Lanza excepciones los parámetros no cumplen el formato y restricciones definidas por el SAT. ArgumentException Si alguno de los parámetros: @pImpuesto, @pTipoFactor no existe en su catálogo.</p>
<pre>public void addImpTrasladoToConcepto(int pIndexConc, decimal pBase, string pImpuesto, string pTipoFactor, decimal? pTasaOCuota, decimal? pImporte);</pre>	<p>Crea y agrega un nodo Traslado al Concepto indicado por el valor @pIndexConc, el nodo traslado tendrá los valores: @pBase, @pImpuesto, @pTipoFactor, @pTasaOCuota, @pImporte.</p> <p>XPath: Cfdi:Conceptos/cfdi:concepto[@pIndexConc]/cfdi:Impuestos/cfdi:Traslados/cfdi:Traslado</p> <p>Excepciones: Exception Si el concepto apuntado por @pIndexConc no existe. Lanza excepciones si no existen los valores en el catálogo o si no cumplen los tipos.</p>
<pre>public void addInformacionAduaneraToConcepto (int pIndexConc, string pNumPedimento);</pre>	<p>Crea y agrega un nodo InformacionAduanera al Concepto indicado por el valor @pIndexConc, el nodo InformacionAduanera tendrá los valores: @pNumPedimento.</p> <p>Lanza excepciones si no existen los valores en el catálogo o si no cumplen los tipos.</p> <p>XPath: Cfdi:Conceptos/cfdi:concepto[@pIndexConc]/cfdi:InformacionAduanera</p> <p>Excepciones: Exception Lanza excepciones si no existen los valores en el catálogo o si no cumplen los tipos.</p>
<pre>public void addNamespace(namespaces ns);</pre>	<p>Agrega un <i>namespace</i> a la lista de <i>namespace</i> que contendrá el comprobante</p>
<pre>public ReturnXML generaXml(string pCadConex, bool aplicaValidacionesMatrizSat = true);</pre>	<p>Genera el comprobante con los datos de la instancia, y retorna un objeto tipo ReturnXML, el cual contiene información acerca del resultado de generar el XML, contiene una referencia al objeto xmlDocument.</p>
<pre>public List<namespaces> getNamespaces();</pre>	<p>Obtiene la lista de <i>namespaces</i> del comprobante.</p>
<pre>public ReturnXML sellaComprobante(XmlDocument xml, string rutaCert, string rutaKey, string pwdCert, bool asignaFechaEmisionActual, bool validaXsds);</pre>	<p>Recibe una referencia del de tipo XmlDocument (que es el comprobante por sellar). La ruta del certificado y llave, además de la fecha actual y una bandera para validar contra los XSD.</p>
<pre>public void setCfdirelacionados(string pTipoRelacion, string pUUID);</pre>	<p>Crea y agrega el nodo CfdiRelacionados como hijo directo al comprobante (ver XPath abajo). Verifica que el tipo de relación existe en el catálogo y lanza una excepción sino existe. Así mismo crea un primer hijo nodo CfdiRelacionado con el UUID especificado.</p>



Método de CFDI33	Descripción
	<p>XPath: Cfdi:CfdiRelacionados/cfdi:Relacionado</p> <p>Excepciones: Exception Si los parámetros no cumplen con el formato y restricciones establecidas por el SAT. ArgumentException Si alguno de los parámetros: @pTipoRelacion no existe en su catálogo.</p>
<pre>public void setEmisor(string pRfc, string pNombre, string pRegFiscal);</pre>	<p>Crea y establece un nodo Emisor como hijo directo al comprobante, verifica que existe el régimen fiscal en el catálogo y lanza una excepción sino existe.</p> <p>XPath: Cfdi:Emisor</p> <p>Excepciones: Exception Si los parámetros no cumplen con el formato y restricciones establecidas por el SAT. ArgumentException Si alguno de los parámetros: @pRegFiscal no existe en su catálogo.</p>
<pre>public void setReceptor(string pRfc, string pNombre, string pResidenciaFiscal, string pNumRegIdTrib, string pUsoCfdi);</pre>	<p>Crea y establece un nodo Receptor como hijo directo al comprobante.</p> <p>XPath: Cfdi:Receptor</p> <p>Excepciones: Exception Si los parámetros no cumplen con el formato y restricciones establecidas por el SAT. ArgumentException Si alguno de los parámetros: @pUsoCfdi, @pResidenciaFiscal no existe en su catálogo.</p>
<pre>public void setTotalImpuestos(decimal? pTotImpRetenidos, decimal? pTotImpTrasladados);</pre>	<p>Crea y establece un nodo Impuestos (ver XPath abajo) como hijo directo al comprobante, los parámetros @pTotImpRetenidos y @pTotImpTrasladados son opcionales, pero si se pasan deben cumplir con el tipo t_Importe, en caso contrario se lanza una excepción.</p> <p>XPath: Cfdi:Impuestos</p> <p>Excepciones: Exception Si los parámetros son establecidos (no null) y no cumplen los formatos y restricciones del SAT.</p>
<pre>public ReturnTimbre timbraComprobante(string usuario, string password, string serie, string idEquipo, XmlDocument cfdi, tipoTimbrado TipoTimbrado);</pre>	<p>Genera el timbre del comprobante a partir de XML. Retorna un objeto ReturnTimbre con el Comprobante timbrado y el cbb.</p> <p>@usuario asignado para usar <i>SifeiSignature</i> @password asignado para usar <i>SifeiSignature</i></p> <p>@cfdi XmlDocument con el comprobante a timbrar. @TipoTimbrado es un valor de enumerado tipoTimbrado para indicar si es de tipo pruebas o productivo.</p>



5.3.3 Enumeraciones:

5.3.3.1 Enumeracion `tipoTimbrado`

Valor	Descripción
Productivo	Indica que se debe usar el ambiente de producción.
Pruebas	Valor que indicar que se están haciendo pruebas con <i>SIFEI Signature</i> .

5.4 Namespace `SifeiSignature3_3.Timbre11`

5.4.1 Clases

- `TimbreBE`

5.4.2 Clase `TimbreBE`

Clase que contiene los datos del Timbre Fiscal versión 1.1.

5.4.2.1 Propiedades de clase

Propiedades
<code>public string UUID { get; set; }</code>
<code>public string SelloPAC { get; set; }</code>
<code>public string NoCertificado { get; set; }</code>
<code>public string FechaTimbre { get; set; }</code>
<code>public string CadenaOriginalTimbre { get; set; }</code>
<code>public string RfcProvCertif { get; set; }</code>
<code>public string Version { get; set; }</code>
<code>public string Leyenda { get; set; }</code>
<code>public string SelloCFD { get; set; }</code>

5.5 Namespace `SifeiSignature3_3.Complementos`

5.5.1 Clase `Complemento`

5.5.1.1 Enumeraciones

Nomina12_pSiNoNoEspecificado
SI = 0,
NO = 1,
NoEspecificado = 2

5.5.1.2 Propiedades de clase

Propiedades de clase	Descripción
<code>public namespaces NsComercioExterior1_1 { get }</code>	Obtiene el <i>namespace</i> de Comercio Exterior versión 1.2
<code>public namespaces NsNomina12 { get }</code>	Obtiene el <i>namespace</i> de Nómina versión 1.1
<code>public namespaces NsDetallista131 { get }</code>	Obtiene el <i>namespace</i> de Detallista versión 1.3.1
<code>public namespaces NsIedu10 { get }</code>	Obtiene el <i>namespace</i> de Instituciones Educativas Privadas versión 1.0
<code>public namespaces NsImpLocal10 { get }</code>	Obtiene el <i>namespace</i> de Impuestos locales versión 1.0
<code>public namespaces NsPagos10 { get }</code>	Obtiene el <i>namespace</i> de Complemento pagos versión 1.0



Constructor	Descripción
<pre>public Complemento(string connString, DateTime pFecha);</pre>	<p>Inicializar la clase.</p> <p>@connString representa la cadena de conexión a la base de datos, ejemplo: "Server=localhost;Port=3306;Database=catálogosat;Uid=root;Pwd=123456;"</p>

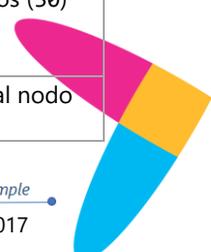
5.5.1.3 Métodos

Métodos complemento	Descripción
<pre>public void Cce11_addDescripcionEspecific(int pIndexMercancia, string pMarca, string pModelo, string pSubmodelo, string pNumSerie);</pre>	<p>Agrega la descripción al complemento de Comercio Exterior.</p> <p>xPath: Cfdi:Complemento/cce11:ComercioExterior/Mercancias/Mercancia/DescripcionesEspecificas</p> <p>Excepciones: Exception Esta excepción ocurre si los parámetros no cumplen con el formato válido establecido por el SAT. También cuando no se ha definido la instancia Comercio Exterior.</p>
<pre>public int Cce11_addDestinatario(string pNumRegIdTrib, // número de identificación o registro fiscal del país string pNombre, //nombre del destinatario string pDomCalle, string pDomNumExt, string pDomNumInt, string pDomColonia, string pDomLocalidad, string pDomReferencia, string pDomMunicipio, string pDomEstado, string pDomPais, string pDomCP)</pre>	<p>Creación del nodo Destinatario y su hijo Domicilio. Se hacen las validaciones correspondientes.</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior/Destinatario</p> <p>Excepciones: Exception Esta excepción ocurre si no cumple con el formato válido establecido por el SAT. ArgumentException Ocurre cuando pDomPais no existe en el catálogo.</p> <p>Returns: Índice del nodo Destinatario que se agregó.</p>
<pre>public void Cce11_addDestinatarioDomicilio(int pIndexDestinatario, string pDomCalle, string pDomNumExt, string pDomNumInt, string pDomColonia, string pDomLocalidad, string pDomReferencia, string pDomMunicipio, string pDomEstado, string pDomPais, string pDomCP);</pre>	<p>Creación y establece un nodo hijo Domicio al nodo Destinatario.</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior/Destinatario/Domicilio</p>
<pre>public int Cce11_addMercancia(string pNoIdentificacion, string pFraccionAranc, decimal? pCantidadAduana, string pUnidadAduana, decimal? pValorUnitAduana, decimal pValorDolares)</pre>	<p>Creación y establece un nodo hijo Mercancia a Mercancias.</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior/Mercancias/Mercancia</p> <p>Returns: Índice del nodo Mercancia que se agregó.</p>

Métodos complemento	Descripción
<pre>public void Cce11_addPropietario(string pNumRegIdTrib, string pResidenciaFiscal)</pre>	<p>Crea y establece un nodo Propietario como hijo a ComercioExterior.</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior/Propietario</p>
<pre>public void Cce11_Define_ComercioExterior(string pMotivoTrasl, string pTipoOperacion, string pClavePedim, int? pCertifOrigen, string pNumCertifOrigen, string pNumExportadorConf, string pIncoterm, int? pSubdivision, string pObservaciones, decimal? pTipoCambioUSD, decimal? pTotalUSD)</pre>	<p>Define el complemento ComercioExterior.</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior</p> <p>Excepciones: Exception Esta excepción ocurre si no cumple con el formato válido establecido por el SAT. ArgumentException Ocurre cuando un parámetro no existe en el catálogo.</p>
<pre>public ReturnXmlComplemento Cce11_generaComplemento()</pre>	<p>Genera el complemento de comercio exterior, lanza una excepción si no se ha definido la instancia comercio exterior (con el método Cce11_Define_ComercioExterior())</p>
<pre>public void Cce11_setEmisor(string pCurp, string pDomCalle, string pDomNumExt, string pDomNumInt, string pDomColonia, string pDomLocalidad, string pDomReferencia, string pDomMunicipio, string pDomEstado, string pDomPais, string pDomCP)</pre>	<p>Crea y establece un nodo Emisor (como hijo de ComercioExterior).</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior/Emisor</p> <p>Excepciones: Exception Si algún atributo no cumple con el formato definido por el SAT, o si no se ha definido el complemento ComercioExterior11.</p>
<pre>public void Cce11_setReceptor(string pNumRegIdTrib, string pDomCalle, string pDomNumExt, string pDomNumInt, string pDomColonia, string pDomLocalidad, string pDomReferencia, string pDomMunicipio, string pDomEstado, string pDomPais, string pDomCP)</pre>	<p>Crea y establece un nodo Receptor (como hijo de ComercioExterior).</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior/Emisor</p> <p>Excepciones: Exception Si algún atributo no cumple con el formato definido por el SAT, o si no se ha definido el complemento ComercioExterior11</p>
<pre>public void Det131_AddAllowanceCharge(detalli staAllowanceChargeAllowanceCharge Type allChargeType, detallistaAllowanceChargeSettleme ntType settType, string numConsec, detallistaAllowanceChargeSpecialS ervicesType? specialServType, decimal? percent)</pre>	<p>Crea y establece un nodo allowanceCharge (como hijo de Detallista).</p> <p>Xpath: Cfdi:Complemento/Detallista/ allowanceCharge</p> <p>Excepciones: Exception Si algún atributo no cumple con el formato definido por el SAT, o si no se ha definido el complemento Detallista (ver Det131_Define_Detallista()), o si se ha sobrepasado el límite de 99.</p>
<pre>public void Det131_addCurrency(detallistaCurrencyCurrencyISOCode isoCode, /*enumeración */ detallistaCurrencyCurrencyFuncio n funcion1, /*enumeración */</pre>	<p>Crea y agrega un nodo currency como hijo al nodo detallista.</p> <p>Xpath: Cfdi:Complemento/Detallista/ currency[]</p> <p>Excepciones:</p>



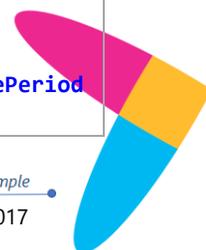
Métodos complemento	Descripción
<pre> detallistaCurrencyCurrencyFuncio n? funcion2, /*enumeración */ detallistaCurrencyCurrencyFuncio n? funcion3, /*enumeración */ decimal? rateOfChange) </pre>	<p>Exception Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos Currency.</p>
<pre> public void Det131_AddCustomsGln(string gln/* número global de localización (GLN) de la aduana */) </pre>	<p>Crea y agrega un nodo Customs como hijo al nodo detallista.</p> <p>Xpath: Cfdi:Complemento/Detallista/ Customs[]</p> <p>Excepciones: Exception Esta excepción ocurre si no si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos Customs.</p>
<pre> public void Det131_AddReferenceIdentification ToAdditionalInformation(detallistaReferenceIdentification Type tipo, /* el número de referencia adicional*/ string valor) </pre>	<p>Crea y agrega un nodo referenceIdentification como hijo al nodo AdditionalInformation.</p> <p>Xpath: Cfdi:Complemento/Detallista/ AdditionalInformation/ referenceIdentification</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos (99) referenceIdentification.</p>
<pre> public void Det131_AddReferenceIdentification ToDeliveryNote(string referenceIdentification /* folio de folio de recepción de mercancia*/) </pre>	<p>Crea y agrega un nodo referenceIdentification como hijo al nodo DeliveryNote.</p> <p>Xpath: Cfdi:Complemento/Detallista/ DeliveryNote/ referenceIdentification</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos (30) referenceIdentification. Por último, si no se ha establecido DeliveryNote (que es opcional).</p>
<pre> public void Det131_AddReferenceIdentification ToOrderIdentification(string referenceIdentification /* número de orden de compra (comprador) a la que hace referencia la factura */) </pre>	<p>Crea y agrega un nodo referenceIdentification como hijo al nodo orderIdentification.</p> <p>Xpath: Cfdi:Complemento/Detallista/ orderIdentification / referenceIdentification</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos (30) referenceIdentification. Por último, si no se ha establecido orderIdentification (que es requerido).</p>
<pre> public void Det131_AddTotalAllowanceCharge(</pre>	<p>Crea y agrega un nodo TotalAllowanceCharge como hermano al nodo TotalAllowanceCharge.</p>



Métodos complemento	Descripción
<pre> detaillistaTotalAllowanceChargeAll owanceOrChargeType allOrChrgType, /*enum para indicar si es cargo o descuento*/ detaillistaTotalAllowanceChargeSpe cialServicesType? specServType, /* enum para indicar el tipo de servicio*/ decimal? Amnt /* total de los cargos o descuentos*/) </pre>	<p>Se debe indicar si es descuento o cargo; es opcional indicar el tipo de servicio y el total de los cargos o descuentos.</p> <p>Xpath: Cfdi:Complemento/Detallista/TotalAllowanceCharge[]</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos (30) referenceIdentification. Por último, si no se ha establecido orderIdentification (que es requerido).</p>
<pre> public int Det131_AgregaSpecialInstruction(detaillistaSpecialInstructionCode, /* tipo de instrucciones comerciales*/ string text) </pre>	<p>Crea y agrega un nodo specialInstruction como hermano al nodo specialInstruction.</p> <p>Xpath: Cfdi:Complemento/Detallista/specialInstruction[]</p> <p>Returns: Índice del nodo specialInstruction que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos (4) specialInstruction.</p>
<pre> public void Det131_AgregaTextASpecialInstru ction(int indexSpecialInstr, /*index del nodo specialInstruction */ string text) </pre>	<p>Crea y agrega un nodo text a specialInstruction (como hijo) dado un índice (el cual indica a qué nodo specialInstruction será el padre).</p> <p>Xpath: Cfdi:Complemento/Detallista/specialInstruction[indexSpecialInstr]/text</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista, si no existe el nodo specialInstruction o si se superó el número máximo de nodos (15) text.</p>
<pre> public void Det131_Agrega_RequestForPaymentId entification_Detallista(detaillistaRequestForPaymentIdenti ficationEntityType EntityType /* tipo de transacción*/) </pre>	<p>Crea y agrega (o reemplaza si existe) un requestForPaymentIdentification a detaillista (como hijo) para indicar la transacción.</p> <p>Xpath: Cfdi:Complemento/Detallista/requestForPaymentIdentification/ entityType</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista.</p>
<pre> public void Det131_Define_Detallista(detaillis taDocumentStatus DocumentStatus) </pre>	<p>Define el complemento detaillista del comprobante.</p> <p>Xpath: Cfdi:Complemento/Detallista</p>
<pre> public ReturnXmlComplemento Det131_generaComplemento() </pre>	<p>Retorna el complemento detaillista en el objeto ReturnXmlComplemento si todo fue bien.</p>



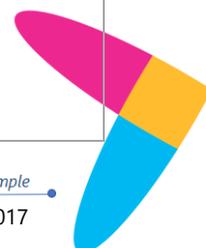
Métodos complemento	Descripción
	<p>Excepciones: Exception Ocurre si no se ha establecido en complemento Detallista.</p>
<pre>public void Det131_SetBuyer(string gln, /* el número global de localización */ string personOrDepartmentName /* contacto de compras */)</pre>	<p>Crea y agrega (o reemplaza si existe) un buyer a detallista (como hijo) para indicar la información del comprador.</p> <p>Xpath: Cfdi:Complemento/Detallista/ buyer/gln, buyer/contactInformation/ personOrDepartmentName/text</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista.</p>
<pre>public void Det131_SetDeliveryNote(DateTime? referenceDate, /*opcional, pasar null para no agregar atributo*/ string ReferenceIdentification)</pre>	<p>Crea y establece un DeliveryNote a detallista (como hijo) para indicar la información recepción de mercancía.</p> <p>Xpath: Cfdi:Complemento/Detallista/ DeliveryNote</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista.</p>
<pre>public void Det131_SetInvoiceCreator(string gln, detallistaInvoiceCreatorAlternate PartyIdentificationType? tipo, /* tipo de identificación secundaria se le asigno al proveedor */ string value, string name, string streetAddressOne,/* domicilio del emisor de la factura */ string city, string cp /* codigo postal del domicilio del emisor de la factura */)</pre>	<p>Crea y establece el InvoiceCreator a detallista (como hijo).</p> <p>Xpath: Cfdi:Complemento/Detallista/ InvoiceCreator/nameAndAddress, InvoiceCreator/ alternatePartyIdentification</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Detallista.</p>
<pre>public void Det131_SetOrderIdentification(DateTime? referenceDate, string ReferenceIdentification)</pre>	<p>Crea y establece el orderIdentification a detallista (como hijo).</p> <p>Xpath: Cfdi:Complemento/Detallista/ orderIdentification/referenceIdentification</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Detallista.</p>
<pre>public void Det131_SetPaymentTerms(detallistaPaymentTermsPaymentTerm sEvent? termsEvent, detallistaPaymentTermsPaymentTerm sRelationTime? termsRelationTime, detallistaPaymentTermsNetPaymentN etPaymentTermsType?</pre>	<p>Crea y establece el paymentTerms a detallista (como hijo) para especificar los términos de pago de la factura</p> <p>Xpath: Cfdi:Complemento/Detallista/ paymentTerms/ y/o Cfdi:Complemento/Detallista/ paymentTerms/paymentTimePeriod y/o</p>



Métodos complemento	Descripción
<pre>netPaymntTermsType, /*para el nodo paymentTimePeriod*/ detallistaPaymentTermsNetPaymentP aymentTimePeriodTimePeriodDueTime Period? pTimePeriod, /*para el nodo paymentTimePeriod*/ string valueTimePeriod, detallistaPaymentTermsDiscountPay mentDiscountType? descType, /*para el nodo discountPayment */ string percent /*para el nodo discountPayment */)</pre>	<p>Cfdi:Complemento/Detallista/ paymentTerms/discountPayment</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Detallista o si se estableció @pTimePeriod o valueTimePeriod cuando se pasó @netPaymntTermsType como null (ya que estos pertenecen a dicho nodo).</p>
<pre>public void Det131_SetSeller(string strGln, detallistaSellerAlternatePartyIde ntificationType tipo, string alternatePartyIdentification)</pre>	<p>Creación y establece el seller a detallista (como hijo) para especificar la información del vendedor.</p> <p>Xpath: Cfdi:Complemento/Detallista/ seller</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Detallista.</p>
<pre>public void Det131_SetShipTo(string gln, string name, string streetAddressOne, string city, string cp)</pre>	<p>Creación y establece el shipTo a detallista (como hijo) para especificar la ubicación donde debe realizarse la entrega de la mercancía.</p> <p>Xpath: Cfdi:Complemento/Detallista/ shipTo</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Detallista.</p>
<pre>public void Det131_setTotalAmount(decimal amnt)</pre>	<p>Creación y establece el totalAmount a detallista (como hijo).</p> <p>Xpath: Cfdi:Complemento/Detallista/ totalAmount</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Detallista.</p>
<pre>public ReturnXmlComplemento Iedu10_generaComplemento(string nombreAlumno, string curp, instEducativasNivelEducativo nivelEducativo, string autRVOE, string RfcPago)</pre>	<p>Genera el complemento concepto de instituciones educativas privadas. El complemento viene la propiedad XMLComplemento de ReturnXmlComplemento.</p> <p>Excepciones: Exception Esta excepción ocurre si los parámetros no cumplen la especificación del SAT en formato o restricción: @curp, @r-fcPago, @autRVOE.</p>
<pre>public void ImpLocal10_Agrega_RetencionImpues toLocal(string nombre,/*Nombre del impuesto local retenido*/ decimal importe, decimal tasa)</pre>	<p>Creación y establece el RetencionesLocales a ImpuestosLocales (como hijo).</p> <p>Xpath: Cfdi:Complemento/ImpuestosLocales/RetencionesLocales</p> <p>Excepciones:</p>



Métodos complemento	Descripción
	<p>Exception Esta excepción ocurre si no se ha establecido el complemento ImpuestosLocales. Si @nombre es null o cadena vacía.</p>
<pre>public void ImpLocal10_Agrega_TrasladoImpuest oLocal(string nombre, decimal importe, decimal tasa)</pre>	<p>Crea y establece el RetencionesLocales a ImpuestosLocales (como hijo).</p> <p>Xpath: Cfdi:Complemento/ImpuestosLocales/RetencionesLocales</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento ImpuestosLocales. Si @nombre es null o cadena vacía.</p>
<pre>public void ImpLocal10_Define_ImpuestoLocal(decimal totalTraslado, decimal totalRetencion)</pre>	<p>Crea y establece el complemento ImpuestosLocales.</p> <p>Xpath: Cfdi:Complemento/ImpuestosLocales</p> <p>Excepciones: Exception Ninguna.</p>
<pre>public ReturnXmlComplemento ImpLocal10_generaComplemento()</pre>	<p>Genera el complemento impuestos locales que es devuelto en el objeto ReturnXmlComplemento, si existe un error no fatal es incluido en el mismo objeto. Ver propiedades:</p> <ul style="list-style-type: none"> • ReturnXmlComplemento.IsXmlComplementoCreated • ReturnXmlComplemento.ErrorDetalle <p>Excepciones: Exception Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación. Si no se pudo formar el nodo IEDU.</p>
<pre>public void Nomina12_addDeduccion(string pTipoDeducc, string pClave, string pConcepto, decimal pImporte)</pre>	<p>Crea y establece una Deduccion a Deducciones (como hijo).</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12: Deducciones/Deduccion[]</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Nómina 1.2. Si no se ha establecido el nodo deducciones. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p> <p>ArgumentException si @pTipoDeducc no existe en el catálogo definido por el SAT.</p>
<pre>public void Nomina12_addHorasExtra(int indexPercepcion, int pDias, string pTipoHoras, int pHorasExtra,</pre>	<p>Crea y establece un nodo HorasExtra (como hijo) a la Percepcion apuntada por @indexPercepcion.</p> <p>Xpath:</p>



Métodos complemento	Descripción
<pre>decimal pImportePagado)</pre>	<p>Cfdi:Complemento/nomina12: Nomina/nomina12:Percepciones/Percepcion[@indexPercepcion]/HorasExtra[]</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Nómina 1.2. Si no se ha establecido el nodo Percepcion. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p> <p>ArgumentException si @pTipoHoras no existe en el catálogo definido por el SAT.</p>
<pre>public void Nomina12_addIncapacidad(int pDias, string pTipo, decimal? pImporteMon)</pre>	<p>Crea y establece un nodo Incapacidad como hijo a Incapacidades.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:Incapacidades/Incapacidad[]</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Nómina 1.2. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p> <p>ArgumentException si @pTipo no existe en el catálogo definido por el SAT.</p>
<pre>public int Nomina12_addOtroPago(string pTipo, /*Tipo de otro pago(catálogo)*/ string pClave, string pConcepto, decimal pImporte)</pre>	<p>Crea y establece un nodo OtroPago como hijo a OtrosPagos para expresar otros pagos aplicables.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:OtrosPagos/OtroPago[]</p> <p>Returns: Índice del nodo OtroPago que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Nómina 1.2. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p> <p>ArgumentException si @pTipo no existe en el catálogo definido por el SAT</p>
<pre>public int Nomina12_addPercepcion(string pTipoPercepcion, string pClave, string pConcepto, decimal pImporteGravado, decimal pImporteExento)</pre>	<p>Crea y establece un nodo Percepcion como hijo a Percepciones.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:Percepciones/Percepcion[]</p> <p>Returns: Índice del nodo Percepcion que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Nómina 1.2.</p>



Métodos complemento	Descripción
	<p>Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p> <p>ArgumentException si @pTipoPercepcion no existe en el catálogo definido por el SAT</p>
<pre>public void Nomina12_addSubContratacion(string pRfcLabora, decimal pPorcentajeTiempo)</pre>	<p>Crea y establece un nodo SubContratacion como hijo a Receptor.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:Receptor/SubContratacion[]</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void Nomina12_DefineNomina(string pTipoNomina, DateTime pFechaPago, DateTime fechaInicPago, DateTime fechaFinPago, decimal numDiasPag, decimal? totPercepciones, decimal? totDeducciones, decimal? totOtrosPagos)</pre>	<p>Define (establece) el complemento Nómina versión 1.2.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/</p> <p>Excepciones: Exception Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación. ArgumentException Si @pTipoNomina no existe en el catálogo definido por el SAT.</p>
<pre>public ReturnXmlComplemento Nomina12_generaComplemento()</pre>	<p>Genera el complemento Nómina versión 1.2 que es devuelto en el objeto ReturnXmlComplemento, si existe un error no fatal es incluido en el mismo objeto. Ver propiedades:</p> <ul style="list-style-type: none"> • ReturnXmlComplemento.IsXmlComplementoCreated • ReturnXmlComplemento.ErrorDetalle <p>Excepciones: Exception Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación. Si la instancia complemento Nómina no está definida.</p>
<pre>public void Nomina12_setAccionesOTitulos(int indexPercepcion, decimal pValorMercado, decimal pPrecioAlOtorgarse)</pre>	<p>Crea y establece el nodo AccionesOTitulos como hijo a la Percepcion apuntada por @indexPercepcion.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:Percepcion[@indexPercepcion]/AccionesOTitulos</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si la Percepcion padre apuntada no existe. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>



Métodos complemento	Descripción
<pre>public void Nomina12_setCompensacionSalDOSAFa vor(int indexOtroPago, decimal pSaldoAFavor, short pAño, decimal pRemanenteAFavor)</pre>	<p>Crea y define la CompensacionSalDOSAFavor al OtroPago apuntado por @indexOtroPago.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/OtrosPagos/OtroPago [@indexOtroPago]/ CompensacionSalDOSAFavor</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si OtrosPagos no está definida. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void Nomina12_setDeducciones(decimal? pTotalOtrasDeducc, decimal? pTotalImpuestosReten)</pre>	<p>Crea y define el nodo Deducciones como hijo a nomima.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Deducciones</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.</p>
<pre>public void Nomina12_setEmisor(string pCurp, string pRegPatronal, string pRfcPatronOrigen)</pre>	<p>Crea y define el nodo Emisor como hijo a nomima.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Emisor</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void Nomina12_setEntidadSNCF(string pOrigenRecurso, decimal pMontoRecursoPropio, bool aplicaMontoRecursoPropio)</pre>	<p>Crea y define el nodo EntidadSNCF como hijo a Emisor.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Emisor/EntidadSNCF</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si no se ha definido la instancia Emisor. ArgumentException Si @pOrigenRecurso no existe en el catálogo definido por el SAT.</p>
<pre>public void Nomina12_setJubilacionPensionReti ro(decimal? pTotalUnaExhibicion, decimal? pTotalParcialidad, decimal? pMontoDiario, decimal pIngresoAcumulable, decimal pIngresoNoAcumulable)</pre>	<p>Crea y define el nodo JubilacionPensionRetiro como hijo a Percepciones.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Percepciones/JubilacionPensionRetiro</p> <p>Excepciones:</p>



Métodos complemento	Descripción
	<p>Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.</p> <p>Si no se ha definido la instancia nodo Percepciones.</p> <p>ArgumentException</p> <p>Ninguna.</p>
<pre>public void Nomina12_setPercepciones(decimal? pTotalSueudos, decimal? pTotalSeparacionIndemn, decimal? pTotalJubilacionPensionRetiro, decimal pTotalGravado, decimal pTotalExento)</pre>	<p>Crea y define el nodo Percepciones como hijo a nomina.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Percepciones</p> <p>Excepciones:</p> <p>Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.</p> <p>ArgumentException</p> <p>Ninguna.</p>
<pre>public void Nomina12_setReceptor(string pCurp, string pNumSegSocial, DateTime? pFechaInicRelLab, string pAntiguedad, string pTipoContrato, Nomina12_pSiNoEspecificado esSindicalizado, string pTipoJornada, string pTipoRegimen, string pNumEmpleado, string pDepartamento, string pPuesto, string pRiesgo, string pPeriodiciadPago, string pBanco, string pCuentaBanco, decimal? pSalarioBaseCotApor, decimal? pSalarioDiario, string pEstado)</pre>	<p>Crea y define el nodo Emisor como hijo a Receptor.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Receptor</p> <p>Excepciones:</p> <p>Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.</p> <p>Si los parámetros no cumplen con las especificiones definidas por el SAT en formato o combinación.</p> <p>ArgumentException</p> <p>Si @pTipoContrato, @pTipoRegimen, @pTipoJornada, @pRiesgo, @pEstado, @pPeriodiciadPago o @pBanco no existe en el catálogo del SAT.</p>
<pre>public void Nomina12_setSeparacionIndemnizaci on(decimal pTotalPagado, int pNumAñosServicio, decimal pUltimoSueldoMensOrd, decimal pIngresoAcumulable, decimal pIngresoNoAcumulable)</pre>	<p>Crea y define el nodo SeparacionIndemnizacion como hijo a Percepciones.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Percepciones/ SeparacionIndemnizacion</p> <p>Excepciones:</p> <p>Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.</p> <p>Si no se ha definido la instancia nodo Percepciones.</p> <p>Si @pNumAñosServicio es mayor a 99.</p> <p>ArgumentException</p> <p>Ninguna.</p>
<pre>public void Nomina12_setSubsidioAlEmpleo(int indexOtroPago, decimal pSubsidioCausado)</pre>	<p>Crea y define el nodo SubsidioAlEmpleo como hijo a OtroPago.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/OtrosPagos/OtroPago/SubsidioAlEmpleo</p>



Métodos complemento	Descripción
	<p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si no se ha definido la instancia nodo OtrosPagos. Si @pNumAñosServicio es mayor a 99. ArgumentException Ninguna.</p>
<pre>public void Pagos10_addDoctoRelacionadoToPago (int indexPago, string pIdDocumento, string pSerie, string pFolio, string pMonedaDR, decimal? pTipoCambioDR, string pMetodoDePagoDR, int? pNumParcialidad, decimal? pImpSaldoAnt, decimal? pImpPagado, decimal? pImpSaldoInsoluto)</pre>	<p>Crea y define el nodo DoctoRelacionado como hijo de Pago.</p> <p>Xpath: Cfdi:Complemento/pagos/pago/ DoctoRelacionado[]</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Pago 1.0. Si no se ha definido la instancia nodo pago. Si los parámetros no cumplen con las especificiones definidas por el SAT en formato o combinación.</p> <p>ArgumentException Si @pMonedaDRes ó @pMetodoDePagoDR no existen en el catálogo del SAT.</p>
<pre>public int Pagos10_addPago(DateTime pFechaPago, string pFormaDePago, string pMoneda, decimal? pTipoCambio, decimal pMonto, string pNumOperacion, string pRfcEmisorCtaOrd, string pNomBancoOrdExt, string pCtaOrdenante, string pRfcEmisorCtaBen, string pCtaBeneficiario, string pTipoCadPago, byte[] pCertPago, string pCadPago, byte[] pSelloPago)</pre>	<p>Crea y define el nodo Pago como hijo de Pagos.</p> <p>Xpath: Cfdi:Complemento/pagos/pago[]</p> <p>Returns: Índice del nodo pago que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Pago 1.0. Si los parámetros no cumplen con las especificiones definidas por el SAT en formato o combinación.</p> <p>ArgumentException Si @pFormaDePago, @pMoneda ó @pTipoCadPago no existen en el catálogo del SAT.</p>
<pre>public ReturnXmlComplemento Pagos10_generaComplemento()</pre>	<p>Genera el complemento pago 1.0 que es devuelto en el objeto ReturnXmlComplemento, si existe un error no fatal es incluido en el mismo objeto. Ver propiedades:</p> <ul style="list-style-type: none"> • ReturnXmlComplemento.IsXmlComplementoCreated • ReturnXmlComplemento.ErrorDetalle <p>Excepciones: Exception Si los parámetros no cumplen con las especificiones definidas por el SAT en formato o combinación. Si no se tiene el mínimo de elementos pagos (1). Si se tiene establecido el nodo impuestos (no soportado para esta versión de librería).</p>

5.6 Namespace SifeiSignature3_3.Returns

5.6.1 Clase ReturnXML

5.6.1.1 Constructores

Constructor	Descripción
<code>public ReturnXML();</code>	Constructor vacío.

5.6.1.2 Propiedades

Nombre de propiedad	Descripción
<code>public bool IsXMLCreated { get; set; }</code>	<i>Booleano</i> que indica si el XML fue creado.
<code>public string ErrorCodigo { get; set; }</code>	Indica el código de error.
<code>public string ErrorDetalle { get; set; }</code>	Propiedad que indicará el error detallado, esto incluye los errores de la " Matriz de Errores " del SAT.
<code>public XmlDocument XML { get; set; }</code>	Contendrá el comprobante en XML generado o un complemento. Esto varía dependiendo del método que lo retorne.
<code>public bool EstaSellado { get; set; }</code>	<i>Booleano</i> que indicará si el comprobante está sellado.
<code>public string CadenaOriginal { get; set; }</code>	<i>String</i> que alberga la cadena original del comprobante.
<code>public string Sello { get; set; }</code>	Sello del comprobante.
<code>public string NoCert { get; set; }</code>	Número de certificado.

5.6.2 Clase ReturnTimbre

5.6.2.1 Constructores

Constructor	Descripción
<code>public ReturnTimbre();</code>	Constructor vacío.

5.6.2.2 Propiedades

Nombre de propiedad	Descripción
<code>public bool IsTimbrado { get; set; }</code>	Indica si se se timbró el CFDI.
<code>public string ErrorDetalle { get; set; }</code>	Indica el detalle del error (si es que hubo) durante el proceso de timbrado.
<code>public string CodigoError { get; set; }</code>	Código de error (si es que hubo) durante el proceso de timbrado.
<code>public XmlDocument Xml { get; set; }</code>	Objeto <code>XmlElement</code> que contiene Timbre en XML.
<code>public byte[] Cbb { get; set; }</code>	Arreglo de bytes del archivo imagen jpg del código de barras bidimensional.
<code>public TimbreBE Timbre { get; set; }</code>	Datos del timbre fiscal versión 1.1.
<code>public XmlNode NodoTimbre { get; set; }</code>	Nodo del complemento timbre.

5.6.3 Clase ReturnXmlComplemento

5.6.3.1 Constructores

Constructor	Descripción
<code>public ReturnXmlComplemento();</code>	Constructor vacío.

5.6.3.2 Propiedades

Constructor	Descripción
<code>public bool IsXmlComplementoCreated { get; set; }</code>	Boleano que indica si se creó el complemento en XML.
<code>public string ErrorDetalle { get; set; }</code>	Incluye el detalle del error cuando <code>IsXmlComplementoCreated</code> es <code>false</code> .
<code>public XmlElement XMLComplemento { get; set; }</code>	Objeto <code>XmlElement</code> que contiene el Complemento.

5.6.4 Clase ReturnCancelacion

5.6.4.1 Constructores

Constructor	Descripción
<code>public ReturnCancelacion();</code>	Constructor vacío.

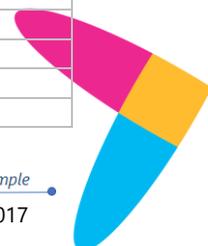
5.6.4.2 Propiedades

Constructor	Descripción
<code>public bool IsCancelado { get; set; }</code>	Indica si fue cancelado(s) los CFDI de forma correcta.
<code>public Acuse AcuseBE { get; set; }</code>	Objeto de tipo <code>Acuse</code> , indica el estado de la operación "Cancelacion" ver clase <code>Acuse</code> .
<code>public string CodigoError { get; set; }</code>	Indica el código de error (si es que hubo).
<code>public string ErrorDetalle { get; set; }</code>	<code>String</code> con el detalle del error.

5.7 Namespace SifeiSignature3_3.Complementos.Detallista131

5.7.1 Enumeraciones de complemento Detallista versión 1.3.1

Enumeraciones de Detallista
<code>detallistaAllowanceChargeAllowanceChargeType</code>
<code>detallistaAllowanceChargeMonetaryAmountOrPercentageRateBase</code>
<code>detallistaAllowanceChargeSettlementType</code>
<code>detallistaAllowanceChargeSpecialServicesType</code>
<code>detallistaCurrencyCurrencyFunction</code>
<code>detallistaCurrencyCurrencyISOCODE</code>
<code>detallistaDocumentStatus</code>
<code>detallistaInvoiceCreatorAlternatePartyIdentificationType</code>
<code>detallistaLineItemAdditionalInformationReferenceIdentificationType</code>
<code>detallistaLineItemAdditionalQuantityQuantityType</code>
<code>detallistaLineItemAllowanceChargeAllowanceChargeType</code>
<code>detallistaLineItemAllowanceChargeSettlementType</code>
<code>detallistaLineItemAllowanceChargeSpecialServicesType</code>
<code>detallistaLineItemAlternateTradeItemIdentificationType</code>
<code>detallistaLineItemCustomsAlternatePartyIdentificationType</code>
<code>detallistaLineItemLogisticUnitsSerialShippingContainerCodeType</code>
<code>detallistaLineItemPalletInformationDescriptionType</code>
<code>detallistaLineItemPalletInformationTransportMethodOfPayment</code>
<code>detallistaLineItemTradeItemDescriptionInformationLanguage</code>
<code>detallistaLineItemTradeItemTaxInformationTaxCategory</code>
<code>detallistaLineItemTradeItemTaxInformationTaxTypeDescription</code>
<code>detallistaOrderIdentificationReferenceIdentificationType</code>
<code>detallistaPaymentTermsDiscountPaymentDiscountType</code>



Enumeraciones de Detallista
<code>detallistaPaymentTermsNetPaymentNetPaymentTermsType</code>
<code>detallistaPaymentTermsNetPaymentPaymentTimePeriodTimePeriodDueTimePeriod</code>
<code>detallistaPaymentTermsPaymentTermsEvent</code>
<code>detallistaPaymentTermsPaymentTermsRelationTime</code>
<code>detallistaReferenceIdentificationType</code>
<code>detallistaRequestForPaymentIdentificationEntityType</code>
<code>detallistaSellerAlternatePartyIdentificationType</code>
<code>detallistaSpecialInstructionCode</code>
<code>detallistaTotalAllowanceChargeAllowanceOrChargeType</code>
<code>detallistaTotalAllowanceChargeSpecialServicesType</code>

5.8 Namespace SifeiSignature3_3.Complementos.Iedu10

5.8.1 Enumeraciones de complemento Instituciones Educativas versión 1.0

Namespace: SifeiSignature3_3.Complementos.Iedu10
<code>instEducativasNivelEducativo</code>

5.8.2 Enumeraciones de complemento Nómina versión 1.2

Namespace: SifeiSignature3_3.Complementos.Nomina12
<code>NominaReceptorSindicalizado</code>

5.9 Referencia del Namespace SifeiSignature3_3.Cancelacion

5.9.1 Clase Acuse

5.9.1.1 Constructores

Constructor	Descripción
<code>public Acuse();</code>	Constructor vacío.

5.9.1.2 Propiedades

Constructor	Descripción
<code>public string AcuseXml { get; set; }</code>	<i>String</i> que contiene el XML del acuse de cancelación.
<code>public DateTime Fecha { get; set; }</code>	Fecha en que se generó el acuse.
<code>public string Rfc { get; set; }</code>	RFC del emisor.
<code>public string Status { get; set; }</code>	Código del estatus de la operación cancelar SAT.
<code>public string Sello { get; set; }</code>	Sello.
<code>public string Mensaje { get; set; }</code>	Mensaje que contiene la descripción del <i>status</i> .
<code>public List<AcuseDetalle> AcuseDetalle { get; set; }</code>	Lista de detalles de acuse (indicando en UUID y estatus) ver clase AcuseDetalle .

5.9.2 Clase AcuseDetalle

5.9.2.1 Constructores

Constructor	Descripción
<code>public AcuseDetalle();</code>	Constructor vacío.



5.9.2.2 Propiedades

Propiedades	Descripción
<code>public string UUID { get; set; }</code>	UUID del CFDI.
<code>public string Status { get; set; }</code>	Código del estatus de la operacion (ver numeral 5.15 Códigos de Respuesta de Cancelación).
<code>public string Mensaje { get; set; }</code>	Mensaje relacionado al estatus.

5.9.3 Clase Cancelacion

5.9.3.1 Constructores

Constructor	Descripción
<code>public Cancelacion ();</code>	Constructor vacío.

5.9.3.2 Métodos

Metodos de Cancelacion	Descripción
<pre>public ReturnCancelacion cancela(ref string msg, string[] listaUUIDS, /*arreglo de UUID a cancelar*/ string rfcSifei, string password, string rfcEmisor, string rutaCert, /*ruta del certificado*/ string rutaKey, /*ruta del key*/ string pwdCert, /*Contraseña del certificado*/ Definiciones.TipoTimbrado pTipoTimbrado);</pre>	<p>Cancela todos los CFDI pasados en el parámetro @listaUUIDS</p> <p>Excepciones: Exception Si no existe el certificado indicado por el path @rutaCert. Si el certificado es inválido. Si hubo un error al crear el PFX.</p> <p>ArgumentException Ninguna.</p> <p>Returns: Objeto de tipo <code>ReturnCancelacion</code> que indica el resultado de la operación cancelar (ver clase <code>ReturnCancelacion</code>).</p>

5.10 Referencia del *Namespace* SifeiSignature3_3.Complementos.Cce11

5.10.1 Clases

- `class` ComercioExterior
- `class` ComercioExteriorDestinatario
- `class` ComercioExteriorDestinatarioDomicilio
- `class` ComercioExteriorEmisor
- `class` ComercioExteriorEmisorDomicilio
- `class` ComercioExteriorMercancia
- `class` ComercioExteriorMercanciaDescripcionesEspecificas
- `class` ComercioExteriorPropietario
- `class` ComercioExteriorReceptor
- `class` ComercioExteriorReceptorDomicilio



5.10.2 Clase ComercioExterior

5.10.2.1 Constructores

Constructor	Descripción
<code>public ComercioExterior();</code>	Constructor vacío.

5.10.2.2 Propiedades

Propiedades
<code>public bool TipoCambioUSDSpecified { get; set; }</code>
<code>public decimal TipoCambioUSD { get; set; }</code>
<code>public string Observaciones { get; set; }</code>
<code>public bool SubdivisionSpecified { get; set; }</code>
<code>public int Subdivision { get; set; }</code>
<code>public bool IncotermSpecified { get; set; }</code>
<code>public string Incoterm { get; set; }</code>
<code>public string NumeroExportadorConfiable { get; set; }</code>
<code>public string NumCertificadoOrigen { get; set; }</code>
<code>public bool CertificadoOrigenSpecified { get; set; }</code>
<code>public decimal TotalUSD { get; set; }</code>
<code>public int CertificadoOrigen { get; set; }</code>
<code>public string ClaveDePedimento { get; set; }</code>
<code>public string TipoOperacion { get; set; }</code>
<code>public bool MotivoTrasladoSpecified { get; set; }</code>
<code>public string MotivoTraslado { get; set; }</code>
<code>public string Version { get; set; }</code>
<code>public ComercioExteriorMercancia[] Mercancias { get; set; }</code>
<code>public ComercioExteriorDestinatario[] Destinatario { get; set; }</code>
<code>public ComercioExteriorReceptor Receptor { get; set; }</code>
<code>public ComercioExteriorPropietario[] Propietario { get; set; }</code>
<code>public ComercioExteriorEmisor Emisor { get; set; }</code>
<code>public bool ClaveDePedimentoSpecified { get; set; }</code>
<code>public bool TotalUSDSpecified { get; set; }</code>

5.10.3 Clase ComercioExteriorDestinatario

5.10.3.1 Constructores

Constructor	Descripción
<code>public ComercioExteriorDestinatario();</code>	Constructor vacío.

5.10.3.2 Propiedades

Propiedades
<code>public ComercioExteriorDestinatarioDomicilio[] Domicilio { get; set; }</code>
<code>public string NumRegIdTrib { get; set; }</code>
<code>public string Nombre { get; set; }</code>



5.10.4 Clase ComercioExteriorDestinatarioDomicilio

5.10.4.1 Constructores

Constructor	Descripción
<code>public ComercioExteriorDestinatarioDomicilio ();</code>	Constructor vacío.

5.10.4.2 Propiedades

Propiedades
<code>public string Calle { get; set; }</code>
<code>public string NumeroExterior { get; set; }</code>
<code>public string NumeroInterior { get; set; }</code>
<code>public string Colonia { get; set; }</code>
<code>public string Localidad { get; set; }</code>
<code>public string Referencia { get; set; }</code>
<code>public string Municipio { get; set; }</code>
<code>public string Estado { get; set; }</code>
<code>public string Pais { get; set; }</code>
<code>public stringCodigoPostal { get; set; }</code>

5.10.5 Clase ComercioExteriorEmisor

5.10.5.1 Constructores

Constructor	Descripción
<code>public ComercioExteriorEmisor ();</code>	Constructor vacío.

5.10.5.2 Propiedades

Propiedades
<code>public ComercioExteriorEmisorDomicilio Domicilio { get; set; }</code>
<code>public string Curp { get; set; }</code>

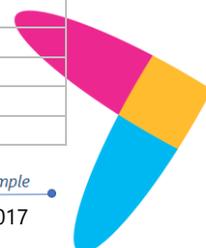
5.10.6 Clase ComercioExteriorEmisor

5.10.6.1 Constructores

Constructor	Descripción
<code>public ComercioExteriorEmisor ();</code>	Constructor vacío.

5.10.6.2 Propiedades

Propiedades
<code>public string Calle { get; set; }</code>
<code>public string NumeroExterior { get; set; }</code>
<code>public string NumeroInterior { get; set; }</code>
<code>public string Colonia { get; set; }</code>
<code>public bool ColoniaSpecified { get; set; }</code>
<code>public string Localidad { get; set; }</code>
<code>public bool LocalidadSpecified { get; set; }</code>
<code>public string Referencia { get; set; }</code>
<code>public string Municipio { get; set; }</code>
<code>public bool MunicipioSpecified { get; set; }</code>



Propiedades
<code>public string Estado { get; set; }</code>
<code>public string Pais { get; set; }</code>
<code>public string CodigoPostal { get; set; }</code>

5.10.7 Clase ComercioExteriorMercancia

5.10.7.1 Constructores

Constructor	Descripción
<code>public ComercioExteriorMercancia();</code>	Constructor vacío.

5.10.7.2 Propiedades

Propiedades
<code>public ComercioExteriorMercanciaDescripcionesEspecificas[] DescripcionesEspecificas { get; set; }</code>
<code>public string NoIdentificacion { get; set; }</code>
<code>public string FraccionArancelaria { get; set; }</code>
<code>public bool FraccionArancelariaSpecified { get; set; }</code>
<code>public decimal CantidadAduana { get; set; }</code>
<code>public bool CantidadAduanaSpecified { get; set; }</code>
<code>public string UnidadAduana { get; set; }</code>
<code>public bool UnidadAduanaSpecified { get; set; }</code>
<code>public decimal ValorUnitarioAduana { get; set; }</code>
<code>public bool ValorUnitarioAduanaSpecified { get; set; }</code>
<code>public decimal ValorDolares { get; set; }</code>

5.10.8 Clase ComercioExteriorMercanciaDescripcionesEspecificas

5.10.8.1 Constructores

Constructor	Descripción
<code>public ComercioExteriorMercanciaDescripcionesEspecificas ();</code>	Constructor vacío.

5.10.8.2 Propiedades

Propiedades
<code>public string Marca { get; set; }</code>
<code>public string Modelo { get; set; }</code>
<code>public string SubModelo { get; set; }</code>
<code>public string NumeroSerie { get; set; }</code>

5.10.9 Clase ComercioExteriorPropietario

5.10.9.1 Constructores

Constructor	Descripción
<code>public ComercioExteriorPropietario ();</code>	Constructor vacío.

5.10.9.2 Propiedades

Propiedades
<code>public string NumRegIdTrib { get; set; }</code>



Propiedades
<code>public string ResidenciaFiscal { get; set; }</code>

5.10.10 Clase ComercioExteriorReceptor

5.10.10.1 Constructores

Constructor	Descripción
<code>public ComercioExteriorReceptor ();</code>	Constructor vacío.

5.10.10.2 Propiedades

Propiedades
<code>public ComercioExteriorReceptorDomicilio Domicilio { get; set; }</code>
<code>public string NumRegIdTrib { get; set; }</code>

5.10.11 Clase ComercioExteriorReceptorDomicilio

5.10.11.1 Constructores

Constructor	Descripción
<code>public ComercioExteriorReceptorDomicilio ();</code>	Constructor vacío.

5.10.11.2 Propiedades

Propiedades
<code>public string Calle { get; set; }</code>
<code>public string NumeroExterior { get; set; }</code>
<code>public string NumeroInterior { get; set; }</code>
<code>public string Colonia { get; set; }</code>
<code>public string Localidad { get; set; }</code>
<code>public string Referencia { get; set; }</code>
<code>public string Municipio { get; set; }</code>
<code>public string Estado { get; set; }</code>
<code>public string Pais { get; set; }</code>
<code>public string CodigoPostal { get; set; }</code>

5.11 Referencia del Namespace SifeiSignature3_3.Complementos.ImpLocal10

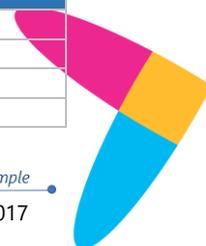
5.11.1 Clase ImpuestosLocales

5.11.1.1 Constructores

Constructor	Descripción
<code>public ImpuestosLocales ();</code>	Constructor vacío.

5.11.1.2 Propiedades

Propiedades
<code>public ImpuestosLocalesRetencionesLocales[] RetencionesLocales { get; set; }</code>
<code>public ImpuestosLocalesTrasladosLocales[] TrasladosLocales { get; set; }</code>
<code>public string version { get; set; }</code>
<code>public decimal TotaldeRetenciones { get; set; }</code>



Propiedades
<code>public decimal TotaldeTraslados { get; set; }</code>

5.11.2 Clase ImpuestosLocalesRetencionesLocales

5.11.2.1 Constructores

Constructor	Descripción
<code>public ImpuestosLocalesRetencionesLocales ();</code>	Constructor vacío.

5.11.2.2 Propiedades

Propiedades
<code>public string ImpLocRetenido { get; set; }</code>
<code>public decimal TasadeRetencion { get; set; }</code>
<code>public decimal Importe { get; set; }</code>

5.11.3 Clase ImpuestosLocalesTrasladosLocales

5.11.3.1 Constructores

Constructor	Descripción
<code>public ImpuestosLocalesTrasladosLocales ();</code>	Constructor vacío.

5.11.3.2 Propiedades

Propiedades
<code>public string ImpLocTrasladado { get; set; }</code>
<code>public decimal TasadeTraslado { get; set; }</code>
<code>public decimal Importe { get; set; }</code>

5.12 Referencia del Namespace SifeiSignature3_3.Complementos.Nomina12

5.12.1 Clases

- class [Nomina](#)
- class [NominaDeducciones](#)
- class [NominaDeduccionesDeduccion](#)
- class [NominaEmisor](#)
- class [NominaEmisorEntidadSNCF](#)
- class [NominaIncapacidad](#)
- class [NominaOtroPago](#)
- class [NominaOtroPagoCompensacionSalDOSAFavor](#)
- class [NominaOtroPagoSubsidioAlEmpleo](#)
- class [NominaPercepciones](#)
- class [NominaPercepcionesJubilacionPensionRetiro](#)
- class [NominaPercepcionesPercepcion](#)
- class [NominaPercepcionesPercepcionAccionesOTitulos](#)
- class [NominaPercepcionesPercepcionHorasExtra](#)
- class [NominaPercepcionesSeparacionIndemnizacion](#)
- class [NominaReceptor](#)



- class [NominaReceptorSubContratacion](#)

5.12.2 Enumeración [NominaReceptorSindicalizado](#)

Enumeraciones de NominaReceptorSindicalizado
NominaReceptorSindicalizado

5.12.3 Clase [Nomina](#)

5.12.3.1 Propiedades

Propiedades
<code>public bool TotalDeduccionesSpecified { get; set; }</code>
<code>public decimal TotalDeducciones { get; set; }</code>
<code>public bool TotalPercepcionesSpecified { get; set; }</code>
<code>public decimal TotalPercepciones { get; set; }</code>
<code>public decimal NumDiasPagados { get; set; }</code>
<code>public DateTime FechaFinalPago { get; set; }</code>
<code>public DateTime FechaInicialPago { get; set; }</code>
<code>public decimal TotalOtrosPagos { get; set; }</code>
<code>public DateTime FechaPago { get; set; }</code>
<code>public string Version { get; set; }</code>
<code>public NominaIncapacidad[] Incapacidades { get; set; }</code>
<code>public NominaOtroPago[] OtrosPagos { get; set; }</code>
<code>public NominaDeducciones Deducciones { get; set; }</code>
<code>public NominaPercepciones Percepciones { get; set; }</code>
<code>public NominaReceptor Receptor { get; set; }</code>
<code>public NominaEmisor Emisor { get; set; }</code>
<code>public string TipoNomina { get; set; }</code>
<code>public bool TotalOtrosPagosSpecified { get; set; }</code>

5.12.4 Clase [NominaDeducciones](#)

5.12.4.1 Propiedades

Propiedades
<code>public NominaDeduccionesDeducccion[] Deducccion { get; set; }</code>
<code>public decimal TotalOtrasDeducciones { get; set; }</code>
<code>public bool TotalOtrasDeduccionesSpecified { get; set; }</code>
<code>public decimal TotalImpuestosRetenidos { get; set; }</code>
<code>public bool TotalImpuestosRetenidosSpecified { get; set; }</code>

5.12.5 Clase [NominaDeduccionesDeducccion](#)

5.12.5.1 Propiedades

Propiedades
<code>public string TipoDeducccion { get; set; }</code>
<code>public string Clave { get; set; }</code>
<code>public string Concepto { get; set; }</code>
<code>public decimal Importe { get; set; }</code>



5.12.6 Clase NominaEmisor

5.12.6.1 Propiedades

Propiedades
<code>public string TipoDeduccion { get; set; }</code>
<code>public string Clave { get; set; }</code>
<code>public string Concepto { get; set; }</code>
<code>public decimal Importe { get; set; }</code>

5.12.7 Clase NominaEmisorEntidadSNCF

5.12.8 Propiedades

Propiedades
<code>public string OrigenRecurso { get; set; }</code>
<code>public decimal MontoRecursoPropio { get; set; }</code>
<code>public bool MontoRecursoPropioSpecified { get; set; }</code>

5.12.9 Clase NominaIncapacidad

5.12.9.1 Propiedades

Propiedades
<code>public int DiasIncapacidad { get; set; }</code>
<code>public string TipoIncapacidad { get; set; }</code>
<code>public decimal ImporteMonetario { get; set; }</code>
<code>public bool ImporteMonetarioSpecified { get; set; }</code>

5.12.10 Clase NominaOtroPago

5.12.10.1 Propiedades

Propiedades
<code>public NominaOtroPagoSubsidioAlEmpleo SubsidioAlEmpleo { get; set; }</code>
<code>public NominaOtroPagoCompensacionSaldosAFavor CompensacionSaldosAFavor { get; set; }</code>
<code>public string TipoOtroPago { get; set; }</code>
<code>public string Clave { get; set; }</code>
<code>public string Concepto { get; set; }</code>
<code>public decimal Importe { get; set; }</code>

5.12.11 Clase NominaOtroPagoCompensacionSaldosAFavor

5.12.11.1 Propiedades

Propiedades
<code>public decimal SaldoAFavor { get; set; }</code>
<code>public short Año { get; set; }</code>
<code>public decimal RemanenteSalFav { get; set; }</code>



5.12.12 Clase NominaOtroPagoSubsidioAlEmpleo

5.12.12.1 Propiedades

Propiedades
<code>public decimal SubsidioCausado { get; set; }</code>

5.12.13 Clase NominaPercepciones

5.12.13.1 Propiedades

Propiedades
<code>public NominaPercepcionesPercepcion[] Percepcion { get; set; }</code>
<code>public NominaPercepcionesJubilacionPensionRetiro JubilacionPensionRetiro { get; set; }</code>
<code>public NominaPercepcionesSeparacionIndemnizacion SeparacionIndemnizacion { get; set; }</code>
<code>public decimal TotalSueldos { get; set; }</code>
<code>public bool TotalSueldosSpecified { get; set; }</code>
<code>public decimal TotalSeparacionIndemnizacion { get; set; }</code>
<code>public bool TotalSeparacionIndemnizacionSpecified { get; set; }</code>
<code>public decimal TotalJubilacionPensionRetiro { get; set; }</code>
<code>public bool TotalJubilacionPensionRetiroSpecified { get; set; }</code>
<code>public decimal TotalGravado { get; set; }</code>
<code>public decimal TotalExento { get; set; }</code>

5.12.14 Clase NominaPercepcionesJubilacionPensionRetiro

5.12.14.1 Propiedades

Propiedades
<code>public decimal TotalUnaExhibicion { get; set; }</code>
<code>public bool TotalUnaExhibicionSpecified { get; set; }</code>
<code>public decimal TotalParcialidad { get; set; }</code>
<code>public bool TotalParcialidadSpecified { get; set; }</code>
<code>public decimal MontoDiario { get; set; }</code>
<code>public bool MontoDiarioSpecified { get; set; }</code>
<code>public decimal IngresoAcumulable { get; set; }</code>
<code>public decimal IngresoNoAcumulable { get; set; }</code>

5.12.15 Clase NominaPercepcionesPercepcion

5.12.15.1 Propiedades

Propiedades
<code>public NominaPercepcionesPercepcionAccionesOTitulos AccionesOTitulos { get; set; }</code>
<code>public NominaPercepcionesPercepcionHorasExtra[] HorasExtra { get; set; }</code>
<code>public string TipoPercepcion { get; set; }</code>
<code>public string Clave { get; set; }</code>
<code>public string Concepto { get; set; }</code>
<code>public decimal ImporteGravado { get; set; }</code>
<code>public decimal ImporteExento { get; set; }</code>



5.12.16 Clase NominaPercepcionesPercepcionAccionesOTitulos

5.12.16.1 Propiedades

Propiedades
<code>public decimal ValorMercado { get; set; }</code>
<code>public decimal PrecioAlOtorgarse { get; set; }</code>

5.12.17 Clase NominaPercepcionesPercepcionHorasExtra

5.12.17.1 Propiedades

Propiedades
<code>public int Dias { get; set; }</code>
<code>public string TipoHoras { get; set; }</code>
<code>public int HorasExtra { get; set; }</code>
<code>public decimal ImportePagado { get; set; }</code>

5.12.18 Clase NominaPercepcionesSeparacionIndemnizacion

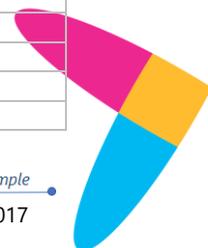
5.12.18.1 Propiedades

Propiedades
<code>public decimal TotalPagado { get; set; }</code>
<code>public int NumAñosServicio { get; set; }</code>
<code>public decimal UltimoSueldoMensOrd { get; set; }</code>
<code>public decimal IngresoAcumulable { get; set; }</code>
<code>public decimal IngresoNoAcumulable { get; set; }</code>

5.12.19 Clase NominaReceptor

5.12.19.1 Propiedades

Propiedades
<code>public decimal SalarioDiarioIntegrado { get; set; }</code>
<code>public bool SalarioBaseCotAporSpecified { get; set; }</code>
<code>public decimal SalarioBaseCotApor { get; set; }</code>
<code>public string CuentaBancaria { get; set; }</code>
<code>public bool BancoSpecified { get; set; }</code>
<code>public string Banco { get; set; }</code>
<code>public string PeriodicidadPago { get; set; }</code>
<code>public bool RiesgoPuestoSpecified { get; set; }</code>
<code>public string RiesgoPuesto { get; set; }</code>
<code>public string Puesto { get; set; }</code>
<code>public string Departamento { get; set; }</code>
<code>public bool SalarioDiarioIntegradoSpecified { get; set; }</code>
<code>public string NumEmpleado { get; set; }</code>
<code>public bool TipoJornadaSpecified { get; set; }</code>
<code>public string TipoJornada { get; set; }</code>
<code>public bool SindicalizadoSpecified { get; set; }</code>
<code>public NominaReceptorSindicalizado Sindicalizado { get; set; }</code>
<code>public string TipoContrato { get; set; }</code>
<code>public string Antigüedad { get; set; }</code>
<code>public bool FechaInicioRelLaboralSpecified { get; set; }</code>
<code>public DateTime FechaInicioRelLaboral { get; set; }</code>
<code>public string NumSeguridadSocial { get; set; }</code>



Propiedades
<code>public string Curp { get; set; }</code>
<code>public NominaReceptorSubContratacion[] SubContratacion { get; set; }</code>
<code>public string TipoRegimen { get; set; }</code>
<code>public string ClaveEntFed { get; set; }</code>

5.12.20 Clase NominaReceptorSubContratacion

5.12.20.1 Propiedades

Propiedades
<code>public string Rfclabora { get; set; }</code>
<code>public decimal PorcentajeTiempo { get; set; }</code>

5.13 Referencia del Namespace SifeiSignature3_3.Complementos.Pagos10

5.13.1 Clases

- class [Pagos](#)
- class [PagosPago](#)
- class [PagosPagoDoctoRelacionado](#)
- class [PagosPagoImpuestos](#)
- class [PagosPagoImpuestosRetencion](#)
- class [PagosPagoImpuestosTraslado](#)

5.13.2 Clase Pagos

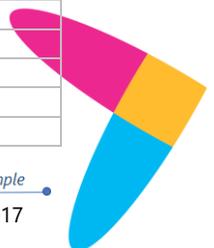
5.13.2.1 Propiedades

Propiedades
<code>public PagosPago[] Pago { get; set; }</code>
<code>public string Version { get; set; }</code>

5.13.3 Clase PagosPago

5.13.3.1 Propiedades

Propiedades
<code>public byte[] CertPago { get; set; }</code>
<code>public bool TipoCadPagoSpecified { get; set; }</code>
<code>public string TipoCadPago { get; set; }</code>
<code>public string CtaBeneficiario { get; set; }</code>
<code>public string RfcEmisorCtaBen { get; set; }</code>
<code>public string CtaOrdenante { get; set; }</code>
<code>public string NomBancoOrdExt { get; set; }</code>
<code>public string RfcEmisorCtaOrd { get; set; }</code>
<code>public string NumOperacion { get; set; }</code>
<code>public decimal Monto { get; set; }</code>
<code>public bool TipoCambioPSpecified { get; set; }</code>
<code>public decimal TipoCambioP { get; set; }</code>
<code>public string MonedaP { get; set; }</code>
<code>public string FormaDePagoP { get; set; }</code>
<code>public DateTime FechaPago { get; set; }</code>
<code>public PagosPagoImpuestos[] Impuestos { get; set; }</code>
<code>public PagosPagoDoctoRelacionado[] DoctoRelacionado { get; set; }</code>



Propiedades
<code>public string CadPago { get; set; }</code>
<code>public byte[] SelloPago { get; set; }</code>

5.13.4 Clase PagosPagoDoctoRelacionado

5.13.4.1 Propiedades

Propiedades
<code>public string IdDocumento { get; set; }</code>
<code>public string Serie { get; set; }</code>
<code>public string Folio { get; set; }</code>
<code>public string MonedaDR { get; set; }</code>
<code>public decimal TipoCambioDR { get; set; }</code>
<code>public bool TipoCambioDRSpecified { get; set; }</code>
<code>public string MetodoDePagoDR { get; set; }</code>
<code>public string NumParcialidad { get; set; }</code>
<code>public decimal ImpSaldoAnt { get; set; }</code>
<code>public bool ImpSaldoAntSpecified { get; set; }</code>
<code>public decimal ImpPagado { get; set; }</code>
<code>public bool ImpPagadoSpecified { get; set; }</code>
<code>public decimal ImpSaldoInsoluto { get; set; }</code>
<code>public bool ImpSaldoInsolutoSpecified { get; set; }</code>

5.13.5 Clase PagosPagoImpuestos

5.13.5.1 Propiedades

Propiedades
<code>public PagosPagoImpuestosRetencion[] Retenciones { get; set; }</code>
<code>public PagosPagoImpuestosTraslado[] Traslados { get; set; }</code>
<code>public decimal TotalImpuestosRetenidos { get; set; }</code>
<code>public bool TotalImpuestosRetenidosSpecified { get; set; }</code>
<code>public decimal TotalImpuestosTrasladados { get; set; }</code>
<code>public bool TotalImpuestosTrasladadosSpecified { get; set; }</code>

5.13.6 Clase PagosPagoImpuestosRetencion

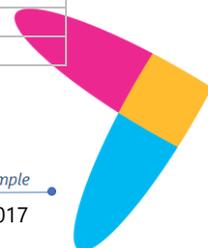
5.13.6.1 Propiedades

Propiedades
<code>public string Impuesto { get; set; }</code>
<code>public decimal Importe { get; set; }</code>

5.13.7 Clase PagosPagoImpuestosTraslado

5.13.7.1 Propiedades

Propiedades
<code>public string Impuesto { get; set; }</code>
<code>public string TipoFactor { get; set; }</code>
<code>public decimal TasaOCuota { get; set; }</code>
<code>public decimal Importe { get; set; }</code>



5.14 Referencia del *Namespace* SifeiSignature3_3.Complementos.Iedu10

5.14.1 Clases

- `class instEducativas`

5.14.2 Enumeración `instEducativasNivelEducativo`

Valores de enumeración
Preescolar = 0,
Primaria = 1,
Secundaria = 2,
Profesionaltécnico = 3,
Bachilleratoosuequivalente = 4

5.14.3 Clase `instEducativas`

5.14.3.1 Propiedades

Propiedades
<code>public string version { get; set; }</code>
<code>public string nombreAlumno { get; set; }</code>
<code>public string CURP { get; set; }</code>
<code>public instEducativasNivelEducativo nivelEducativo { get; set; }</code>
<code>public string autRVOE { get; set; }</code>
<code>public string rfcPago { get; set; }</code>

5.15 Códigos de Respuesta de Cancelación

Estos son los códigos de respuesta que devuelve la operación `cancelar` CFDI.

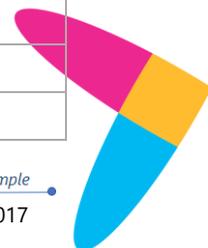
Código	Descripción
201	Cancelado
202	Previamente cancelado
203	UUID no corresponde a Emisor
204	UUID no aplicable para cancelacion
205	UUID no existe
300	Autenticación inválida
301	XML mal formado
302	Sello mal formado o inválido
303	Sello no corresponde a Emisor o caduco
304	Certificado Revocado o Caduco
305	La Fecha de Emision no está dentro de la vigencia del CSD del Emisor
306	El Certificado no es de tipo de CSD
307	El CFDI contiene un timbre previo
308	Certificado no expedido por el SAT

5.16 Códigos de Error de la Matriz de validaciones para CFDI versión 3.3

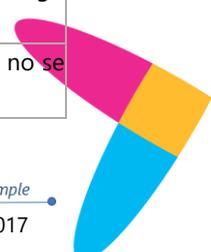
Código	Error
CFDI33101	El campo Fecha no cumple con el patrón requerido
CFDI33102	El resultado de la digestión debe ser igual al resultado de la descripción del sello.



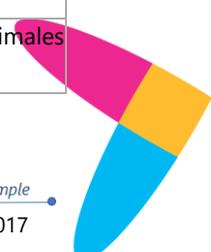
Código	Error
CFDI33103	Si existe el complemento para recepción de pagos el campo FormaPago no debe existir.
CFDI33104	El campo FormaPago no contiene un valor del catálogo c_FormaPago.
CFDI33105	EL certificado no cumple con alguno de los valores permitidos.
CFDI33106	El valor de este campo SubTotal excede la cantidad de decimales que soporta la moneda.
CFDI33107	El TipoDeComprobante es I, E o N, el importe registrado en el campo no es igual a la suma de los importes de los conceptos registrados.
CFDI33108	El TipoDeComprobante es T o P y el importe no es igual a 0, o cero con decimales.
CFDI33109	El valor registrado en el campo Descuento no es menor o igual que el campo Subtotal.
CFDI33110	El TipoDeComprobante no es I, E o N, y un concepto incluye el campo descuento.
CFDI33111	El valor del campo Descuento excede la cantidad de decimales que soporta la moneda.
CFDI33112	El campo Moneda no contiene un valor del catálogo c_Moneda.
CFDI33113	El campo TipoCambio no tiene el valor "1" y la moneda indicada es MXN.
CFDI33114	El campo TipoCambio se debe registrar cuando el campo Moneda tiene un valor distinto de MXN y XXX.
CFDI33115	El campo TipoCambio no se debe registrar cuando el campo Moneda tiene el valor XXX.
CFDI33116	El campo TipoCambio no cumple con el patrón requerido.
CFDI33117	Cuando el valor del campo TipoCambio se encuentre fuera de los límites establecidos, debe existir el campo Confirmacion.
CFDI33118	El campo Total no corresponde con la suma del subtotal, menos los descuentos aplicables, más las contribuciones recibidas (impuestos trasladados - federales o locales, derechos, productos, aprovechamientos, aportaciones de seguridad social, contribuciones de mejoras) menos los impuestos retenidos.
CFDI33119	Cuando el valor del campo Total se encuentre fuera de los límites establecidos, debe existir el campo Confirmacion.
CFDI33120	El campo TipoDeComprobante, no contiene un valor del catálogo c_TipoDeComprobante.
CFDI33121	El campo MetodoPago, no contiene un valor del catálogo c_MetodoPago.
CFDI33122	Cuando se tiene el valor PIP en el campo MetodoPago y el valor en el campo TipoDeComprobante es I o E, el CFDI debe contener un complemento de recibo de pago.
CFDI33123	Se debe omitir el campo MetodoPago cuando el TipoDeComprobante es T o P.
CFDI33124	Si existe el complemento para recepción de pagos en este CFDI el campo MetodoPago no debe existir.
CFDI33125	El campo LugarExpedicion, no contiene un valor del catálogo c_CodigoPostal.
CFDI33126	El campo Confirmacion no debe existir cuando los atributos TipoCambio y/o Total están dentro del rango permitido.
CFDI33127	Número de confirmación inválido.
CFDI33128	Número de confirmación utilizado previamente.
CFDI33129	El campo TipoRelacion, no contiene un valor del catálogo c_TipoRelacion.
CFDI33130	El campo RegimenFiscal, no contiene un valor del catálogo c_RegimenFiscal.



Código	Error
CFDI33131	La clave del campo RegimenFiscal debe corresponder con el tipo de persona (física o moral).
CFDI33132	Este RFC del receptor no existe en la lista de RFC inscritos no cancelados del SAT.
CFDI33133	El campo ResidenciaFiscal, no contiene un valor del catálogo c_Pais.
CFDI33134	El RFC del receptor es de un RFC registrado en el SAT o un RFC genérico nacional y existe el campo ResidenciaFiscal.
CFDI33135	El valor del campo ResidenciaFiscal no puede ser MEX.
CFDI33136	Se debe registrar un valor de acuerdo al catálogo c_Pais en el campo ResidenciaFiscal, cuando en el en el campo NumRegIdTrib se registre información.
CFDI33137	El valor del campo es un RFC inscrito no cancelado en el SAT o un RFC genérico nacional, y se registró el campo NumRegIdTrib.
CFDI33138	Para registrar el campo NumRegIdTrib, el CFDI debe contener el complemento de comercio exterior y el RFC del receptor debe ser un RFC genérico extranjero.
CFDI33139	El campo NumRegIdTrib no cumple con el patrón correspondiente.
CFDI33140	El campo UsoCFDI, no contiene un valor del catálogo c_UsoCFDI.
CFDI33141	La clave del campo UsoCFDI debe corresponder con el tipo de persona (física o moral).
CFDI33142	El campo ClaveProdServ, no contiene un valor del catálogo c_ClaveProdServ.
CFDI33143	No existe el complemento requerido para el valor de ClaveProdServ.
CFDI33144	No está declarado el impuesto relacionado con el valor de ClaveProdServ.
CFDI33145	El campo ClaveUnidad no contiene un valor del catálogo c_ClaveUnidad.
CFDI33146	El valor del campo ValorUnitario debe tener hasta la cantidad de decimales que soporte la moneda.
CFDI33147	El valor del campo ValorUnitario debe ser mayor que cero (0) cuando el tipo de comprobante es Ingreso, Egreso o Nomina.
CFDI33148	El valor del campo Importe debe tener hasta la cantidad de decimales que soporte la moneda.
CFDI33149	El valor del campo Importe no se encuentra entre el límite inferior y superior permitido.
CFDI33150	El valor del campo Descuento debe tener hasta la cantidad de decimales que tenga registrado el atributo importe del concepto.
CFDI33151	El valor del campo Descuento es mayor que el campo Importe.
CFDI33152	En caso de utilizar el nodo Impuestos en un concepto, se deben incluir impuestos de traslado y/o retenciones.
CFDI33153	El valor del campo Base que corresponde a Traslado debe tener hasta la cantidad de decimales que soporte la moneda.
CFDI33154	El valor del campo Base que corresponde a Traslado debe ser mayor que cero.
CFDI33155	El valor del campo Impuesto que corresponde a Traslado no contiene un valor del catálogo c_Impuesto.
CFDI33156	El valor del campo TipoFactor que corresponde a Traslado no contiene un valor del catálogo c_TipoFactor.
CFDI33157	Si el valor registrado en el campo TipoFactor que corresponde a Traslado es Exento no se deben registrar los campos TasaOCuota ni Importe.



Código	Error
CFDI33158	Si el valor registrado en el campo TipoFactor que corresponde a Traslado es Tasa o Cuota, se deben registrar los campos Tasa0Cuota e Importe.
CFDI33159	El valor del campo Tasa0Cuota que corresponde a Traslado no contiene un valor del catálogo c_Tasa0cuota o se encuentra fuera de rango.
CFDI33160	El valor del campo Importe que corresponde a Traslado debe tener hasta la cantidad de decimales que soporte la moneda.
CFDI33161	El valor del campo Importe o que corresponde a Traslado no se encuentra entre el límite inferior y superior permitido.
CFDI33162	El valor del campo Base que corresponde a Retención debe tener hasta la cantidad de decimales que soporte la moneda.
CFDI33163	El valor del campo Base que corresponde a Retención debe ser mayor que cero.
CFDI33164	El valor del campo Impuesto que corresponde a Retención no contiene un valor del catálogo c_Impuesto.
CFDI33165	El valor del campo TipoFactor que corresponde a Retención no contiene un valor del catálogo c_TipoFactor.
CFDI33166	Si el valor registrado en el campo TipoFactor que corresponde a Retención debe ser distinto de Exento.
CFDI33167	El valor del campo Tasa0Cuota que corresponde a Retención, no contiene un valor del catálogo c_Tasa0cuota o se encuentra fuera de rango.
CFDI33168	El valor del campo Importe que corresponde a Retención debe tener hasta la cantidad de decimales que soporte la moneda.
CFDI33169	El valor del campo Importe que corresponde a Retención no se encuentra entre el límite inferior y superior permitido.
CFDI33170	El número de pedimento es inválido.
CFDI33171	El NumeroPedimento no debe existir si se incluye el complemento de comercio exterior.
CFDI33172	El campo ClaveProdServ, no contiene un valor del catálogo c_ClaveProdServ.
CFDI33173	El valor del campo ValorUnitario debe tener hasta la cantidad de decimales que soporte la moneda.
CFDI33174	El valor del campo ValorUnitario debe ser mayor que cero (0).
CFDI33175	El valor del campo Importe debe tener hasta la cantidad de decimales que soporte la moneda.
CFDI33176	El valor del campo Importe no se encuentra entre el límite inferior y superior permitido.
CFDI33177	El número de pedimento es inválido.
CFDI33178	El NumeroPedimento no debe existir si se incluye el complemento de comercio exterior.
CFDI33179	Cuando el TipoDeComprobante sea T o P, el elemento Impuestos no debe existir.
CFDI33180	El valor del campo TotalImpuestosRetenidos debe tener hasta la cantidad de decimales que soporte la moneda.
CFDI33181	El valor del campo TotalImpuestosRetenidos debe ser igual a la suma de los importes registrados en el elemento hijo Retencion.
CFDI33182	El valor del campo TotalImpuestosTrasladados debe tener hasta la cantidad de decimales que soporte la moneda.



Código	Error
CFDI33183	El valor del campo TotalImpuestosTrasladados no es igual a la suma de los importes registrados en el elemento hijo Traslado.
CFDI33184	Debe existir el campo TotalImpuestosRetenidos.
CFDI33185	El campo Impuesto no contiene un valor del catálogo c_Impuesto.
CFDI33186	Debe haber sólo un registro por cada tipo de impuesto retenido.
CFDI33187	Debe existir el campo TotalImpuestosRetenidos.
CFDI33188	El valor del campo Importe correspondiente a Retención debe tener hasta la cantidad de decimales que soporte la moneda.
CFDI33189	El campo Importe correspondiente a Retención no es igual a la suma de los importes de los impuestos retenidos registrados en los conceptos donde el impuesto sea igual al campo Impuesto de este elemento.
CFDI33190	Debe existir el campo TotalImpuestosTrasladados.
CFDI33191	El campo Impuesto no contiene un valor del catálogo c_Impuesto.
CFDI33192	Debe haber sólo un registro con la misma combinación de impuesto, factor y tasa por cada traslado.
CFDI33193	El valor seleccionado debe corresponder a un valor del catálogo donde la columna impuesto corresponda con el campo impuesto y la columna factor corresponda con el campo TipoFactor.
CFDI33194	El valor del campo Importe correspondiente a Traslado debe tener hasta la cantidad de decimales que soporte la moneda.
CFDI33195	El campo Importe correspondiente a Traslado no es igual a la suma de los importes de los impuestos trasladados registrados en los conceptos donde el impuesto del concepto sea igual al campo impuesto de este elemento y la Tasa0Cuota del concepto sea igual al campo Tasa0Cuota de este elemento.
CFDI33196	Error no identificado.



6. Información de Contacto con SIFEI

CENTRO DE SOPORTE TÉCNICO SIFEI

Acceso a recursos de Soporte Técnico de los productos y servicios de SIFEI, Preguntas Frecuentes, Manuales de Usuario, Manuales Técnicos, Notas Técnicas, entre otros.

Dirección electrónica

[Centro de Soporte Técnico SIFEI](#)

TELÉFONOS DE CONTACTO

Orizaba, Ver.	+52 (272) 726 6999
CDMX	(55) 4624 0146
Puebla, Pue.	(222) 211 6156
con 10 líneas	

ATENCIÓN A INCIDENTES

La atención a incidentes se realizará mediante una herramienta de gestión de incidentes y la comunicación se realizará mediante correo electrónico.

Correo Electrónico

helpdesk@sifei.com.mx

HORARIO DE ATENCIÓN

El horario de atención a clientes y de Soporte Técnico para para preguntas, dudas o problemas de la aplicación es:

Lunes a viernes

De 09:00 a 19:00 hrs.

PÁGINAS OFICIALES DE SIFEI

Sitio web	http://www.sifei.com.mx/
Facebook	http://www.facebook.com/sifeimexico
Twitter	http://twitter.com/#!/SIFEIMexico
YouTube	https://www.youtube.com/channel/UCDSLHrFOg2tGvG3AruHNmaw
LinkedIn	https://www.linkedin.com/company/18265037/

UBICACIÓN DE OFICINA MATRIZ

Primera Privada de Oriente 17 No. 32
Col. Centro, Orizaba, Veracruz, México
CP 94300

<Fin del Documento>

