



# Manual Técnico de SIFEI Signature

CFDI v3.3

Agosto 2019

## Control de Versiones

Nombre	No. Versión	Modificaciones	Fecha
Daniel Jesús Hernández Francisco	1.0	Generación del documento	19/oct/2017
Daniel Jesús Hernández Francisco	1.1	Modificación del documento	25/nov/2017
Daniel Jesús Hernández Francisco	1.2	Modificación del documento	09/abril/2019

## Control de Revisiones

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

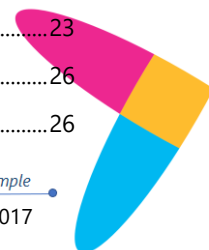
## Control de Autorizaciones

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

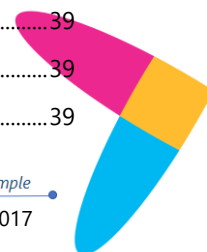


## Contenido

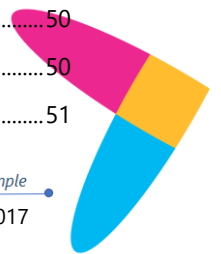
1.	Generalidades.....	1
1.1	Introducción.....	1
1.2	Objetivo.....	1
1.3	Alcance.....	1
1.4	Requerimientos Mínimos.....	1
1.5	Términos y Definiciones.....	1
1.6	Términos y definiciones técnicas.....	2
1.7	Referencias.....	2
1.8	Recursos.....	2
1.9	Versiones de CFDI que se timbran.....	2
1.9.1	Complementos para CFDI 3.3.....	2
1.9.2	Complementos para Comprobante de Retención e Información de Pagos.....	3
1.9.3	Métodos de Timbrado CFDI 3.3.....	3
1.9.4	Métodos de Timbrado Comprobante de Retención e Información de Pagos.....	3
1.9.5	Metodos de proceso de cancelación CFDI 3.3.....	3
2.	Convenciones de documentación.....	3
2.1	Atributos opcionales.....	4
3.	Primeros pasos ( <i>getting started</i> ).....	5
3.1	Agregar recursos.....	6
3.1.1	Copiar la carpeta SAT y las librerías.....	6
3.1.2	Incluir la carpeta SAT y d11 en el proyecto.....	6
3.1.3	Configurar los archivos para ser copiados en el directorio de salida.....	7
3.2	Agregar la biblioteca al proyecto.....	8
3.3	Agregar información de los servicios web (configurar WS).....	10
3.4	Crear base de datos.....	13
3.4.1	Comandos usados para crear la base de datos.....	13
3.4.2	Cambiar max_allowed_packet (Opcional).....	14
3.4.3	Ejecutar el <i>script</i> .....	16
3.4.4	Crear la base de datos en Mysql.....	17
3.5	Mysql y MariaDb en Servidores Linux, Windows y Mac OS X.....	21
3.5.1	<i>Case sensitive</i> (sensibilidad a minúsculas y mayúsculas).....	21
3.5.2	Hacer insensible Mysql y MariaDb en sistemas basados en Unix.....	23
4.	Ejemplos CFDI.....	26
4.1	Instanciar un Objeto tipo CFDI33.....	26



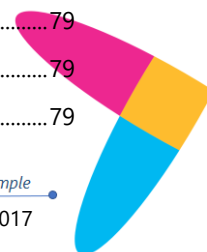
- 4.2 Establecer CfdiRelacionados.....27
- 4.3 Agregar un CfdiRelacionado.....27
- 4.4 Agregar el Emisor .....27
- 4.5 Agregar el Receptor.....28
- 4.6 Agregar un concepto.....28
- 4.7 Agregar Información aduanera a un concepto .....29
- 4.8 Agregar un impuesto traslado a un concepto.....29
- 4.9 Agregar total impuestos de comprobante .....29
- 4.10 Agregar un impuesto traslado a impuestos de comprobante .....30
- 4.11 Autorización para el uso de Sifei Signature .....30
- 4.12 Generar el XML.....31
- 4.13 Sellar el comprobante.....33
- 4.14 Timbrar el comprobante .....34
- 4.15 Recuperar timbre.....35
- 5. Crear Complementos.....35
  - 5.1 Complemento Comercio Exterior.....35
    - 5.1.1 Definir complemento Comercio Exterior.....35
    - 5.1.2 Establecer Emisor de Comercio Exterior.....36
    - 5.1.3 Definir Receptor de Comercio Exterior.....36
    - 5.1.4 Agregar mercancía de Comercio Exterior .....36
    - 5.1.5 Generar el Complemento Comercio Exterior.....36
  - 5.2 Complemento Nómina .....37
    - 5.2.1 Definir complemento Nómina.....37
    - 5.2.2 Agregar un emisor a Nómina .....37
    - 5.2.3 Agregar un receptor a Nómina.....37
    - 5.2.4 Establecer que Nómina tiene percepciones.....37
    - 5.2.5 Agregar una percepción a percepciones de Nómina.....38
    - 5.2.6 Agregar una Separación indemnización a percepciones de Nómina.....38
    - 5.2.7 Establer deducciones a Nómina .....38
    - 5.2.8 Agregar una deducción a deducciones de Nómina .....38
    - 5.2.9 Generar el Complemento Nómina.....38
  - 5.3 Complemento Detallista.....38
    - 5.3.1 Definir complemento Detallista.....38
    - 5.3.2 Establecer el requestForPaymentIdentification a Detallista.....39
    - 5.3.3 Agregar un specialInstruction a Detallista.....39
    - 5.3.4 Establecer el orderIdentification a Detallista.....39



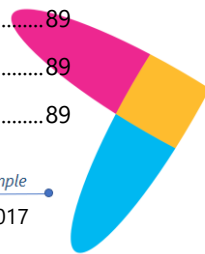
- 5.3.5 Agregar un referenceIdentification al AdditionalInformation de Detallista.....39
- 5.3.6 Establecer un DeliveryNote a Detallista.....39
- 5.3.7 Definir el Buyer (comprador) de Detallista .....39
- 5.3.8 Definir el Seller (vendedor) a Detallista .....39
- 5.3.9 Establer el shipTo (lugar de envío) de Detallista .....40
- 5.3.10 Establecer el InvoiceCreator a Detallista.....40
- 5.3.11 Agregar un Customs a Detallista.....40
- 5.3.12 Agregar un currency a Detallista .....40
- 5.3.13 Establecer el paymentTerms a Detallista.....40
- 5.3.14 Establecer un allowanceCharge a Detallista para definir cargos o descuentos.....40
- 5.3.15 Definir el totalAmount a Detallista .....41
- 5.3.16 Definir el TotalAllowanceCharge a Detallista para definir el monto total de cargos o descuentos .....41
- 5.3.17 Generar el complemento Detallista.....41
- 5.4 Complemento Recepción de Pagos .....41
  - 5.4.1 Definir Complemento de Recepción de Pagos .....41
  - 5.4.2 Agregar un DoctoRelacionado a Pago.....42
  - 5.4.3 Generar el Complemento Pago .....42
- 5.5 Definir y generar el complemento IEDU (a un concepto).....42
- 5.6 Complemento Impuestos Locales.....42
  - 5.6.1 Definir el complemento Impuestos Locales .....42
  - 5.6.2 Agregar Impuesto Local Trasladado al complemento.....43
  - 5.6.3 Generar el Complemento Impuestos Locales .....43
- 6. Proceso de Cancelación.....43
  - 6.1 Cancelar un CFDI 2018 (actualización) .....43
  - 6.2 Cancelar CFDIs Nuevo Proceso.....44
  - 6.3 Consulta CFDI relacionados .....45
  - 6.4 Consulta CFDI Estatus.....46
  - 6.5 Consulta peticiones de Cancelación pendientes (que requieren autorización).....47
  - 6.6 Procesar Respuesta de Peticion de Cancelacion (Aceptacion o Rechazo de cancelación) .....48
- 7. Comprobante de Retención de Información de Pagos .....49
  - 7.1 Instanciar un objeto de Comprobante de Retención e Información de Pagos.....49
  - 7.2 Definir la retención.....49
  - 7.3 Definir el Emisor.....50
  - 7.4 Definir Receptor .....50
  - 7.5 Definir el Periodo.....50
  - 7.6 Definir Totales.....51



7.7	Agregar Impuesto Retenido.....	51
7.8	Autorización para el uso de Sifei Signature.....	51
7.9	Generar el XML.....	51
7.10	Sellar el Comprobante de Retención e Información de Pagos.....	52
7.11	Timbrar el Comprobante de Retenciones e Información de Pagos.....	53
8.	Complementos de Comprobante de Retenciones e Información de Pagos.....	54
8.1	Complemento Servicios Plataformas Tecnológicas.....	54
8.1.1	Definir Complemento Servicios Plataformas Tecnológicas.....	54
8.1.2	Agregar DetallesDelServicio.....	55
8.1.3	Generar Complemento Servicios Plataformas Tecnológicas.....	55
9.	Referencia.....	56
9.1	Namespaces.....	56
9.2	Clases principales.....	56
9.3	Namespace SifeiSignature3_3.Version3_3.....	56
9.3.1	Clase CFDI33.....	57
9.3.2	Métodos.....	57
9.3.3	Enumeraciones:.....	61
9.4	Namespace SifeiSignature3_3.Timbre11.....	61
9.4.1	Clases.....	61
9.4.2	Clase TimbreBE.....	61
9.5	Namespace SifeiSignature3_3.Complementos.....	62
9.5.1	Clase Complemento.....	62
9.6	Namespace SifeiSignature3_3>Returns.....	74
9.6.1	Clase ReturnXML.....	74
9.6.2	Clase ReturnTimbre.....	75
9.6.3	Clase ReturnXmlComplemento.....	75
9.6.4	Clase ReturnCancelacion2018.....	76
9.6.5	Clase ReturnDocumentosRelacionados.....	76
9.6.6	Clase ReturnQrAcuse.....	76
9.6.7	Clase ReturnPeticonesPendientes.....	77
9.6.8	Clase ReturnAceptacionRechazoResult.....	77
9.6.9	Clase DetalleFolios.....	78
9.7	Namespace SifeiSignature3_3.Complementos.Detallista131.....	79
9.7.1	Enumeraciones de complemento Detallista versión 1.3.1.....	79
9.8	Namespace SifeiSignature3_3.Complementos.Iedu10.....	79
9.8.1	Enumeraciones de complemento Instituciones Educativas versión 1.0.....	79



9.8.2	Enumeraciones de complemento Nómina versión 1.2.....	79
9.9	Referencia del <i>Namespace</i> SifeiSignature3_3.Cancelacion.....	80
9.9.1	Clase Acuse.....	80
9.9.2	Clase AcuseDetalle.....	80
9.9.3	Clase Cancelacion (deprecada, obsoleta).....	80
9.9.4	Clase Cancelacion 2018.....	81
9.10	Referencia del <i>Namespace</i> SifeiSignature3_3.Complementos.Cce11.....	82
9.10.1	Clases.....	82
9.10.2	Clase ComercioExterior.....	82
9.10.3	Clase ComercioExteriorDestinatario.....	83
9.10.4	Clase ComercioExteriorDestinatarioDomicilio.....	83
9.10.5	Clase ComercioExteriorEmisor.....	84
9.10.6	Clase ComercioExteriorEmisor.....	84
9.10.7	Clase ComercioExteriorMercancia.....	84
9.10.8	Clase ComercioExteriorMercanciaDescripcionesEspecificas.....	85
9.10.9	Clase ComercioExteriorPropietario.....	85
9.10.10	Clase ComercioExteriorReceptor.....	85
9.10.11	Clase ComercioExteriorReceptorDomicilio.....	86
9.11	Referencia del <i>Namespace</i> SifeiSignature3_3.Complementos.ImpLocal10.....	86
9.11.1	Clase ImpuestosLocales.....	86
9.11.2	Clase ImpuestosLocalesRetencionesLocales.....	86
9.11.3	Clase ImpuestosLocalesTrasladosLocales.....	87
9.12	Referencia del <i>Namespace</i> SifeiSignature3_3.Complementos.Nomina12.....	87
9.12.1	Clases.....	87
9.12.2	Enumeración NominaReceptorSindicalizado.....	87
9.12.3	Clase Nomina.....	87
9.12.4	Clase NominaDeducciones.....	88
9.12.5	Clase NominaDeduccionesDeduccion.....	88
9.12.6	Clase NominaEmisor.....	88
9.12.7	Clase NominaEmisorEntidadSNCF.....	88
9.12.8	Propiedades.....	88
9.12.9	Clase NominaIncapacidad.....	89
9.12.10	Clase NominaOtroPago.....	89
9.12.11	Clase NominaOtroPagoCompensacionSalDOSAFavor.....	89
9.12.12	Clase NominaOtroPagoSubsidioAlEmpleo.....	89
9.12.13	Clase NominaPercepciones.....	89



9.12.14	Clase NominaPercepcionesJubilacionPensionRetiro.....	90
9.12.15	Clase NominaPercepcionesPercepcion.....	90
9.12.16	Clase NominaPercepcionesPercepcionAccionesOTitulos.....	90
9.12.17	Clase NominaPercepcionesPercepcionHorasExtra.....	90
9.12.18	Clase NominaPercepcionesSeparacionIndemnizacion.....	90
9.12.19	Clase NominaReceptor.....	91
9.12.20	Clase NominaReceptorSubContratacion.....	91
9.13	Referencia del <i>Namespace</i> SifeiSignature3_3.Complementos.Pagos10.....	91
9.13.1	Clases.....	91
9.13.2	Clase Pagos.....	91
9.13.3	Clase PagosPago.....	92
9.13.4	Clase PagosPagoDoctoRelacionado.....	92
9.13.5	Clase PagosPagoImpuestos.....	92
9.13.6	Clase PagosPagoImpuestosRetencion.....	93
9.13.7	Clase PagosPagoImpuestosTraslado.....	93
9.14	Referencia del <i>Namespace</i> SifeiSignature3_3.srvSifeiCancelacion2018.....	93
9.14.1	Clases.....	93
9.14.2	Enumeraciones.....	93
9.14.3	Clase folios.....	93
9.15	Códigos de Respuesta de Cancelación.....	93
9.16	Códigos de Error de la Matriz de validaciones para CFDI versión 3.3.....	94
9.17	Codigos de Error del SAT para el metodo Consulta CFDI Relacionado.....	98
9.18	Codigos de error de SAT para el metodo obtener Peticiones de Cancelacion Pendientes.....	98
9.19	Codigos de error del SAT para el metodo Aceptacion/Rechazo de cancelacion (procesa Respuesta).....	98
9.20	Codigos de respuesta del servicio de consulta CFDI.....	98
10.	Información de Contacto con SIFEI.....	100





## Listado de Imágenes

Imagen 1 Creación de proyecto .Net Framework 3.5 .....	5
Imagen 2 Proyecto recién creado.....	5
Imagen 3 Copiando los recursos al proyecto.....	6
Imagen 4 Inclusión de carpeta SAT en el proyecto.....	7
Imagen 5 Inclusión del resto de bibliotecas .....	7
Imagen 6 Modificando la propiedad "Copiar al directorio de Salida".....	8
Imagen 7 Modificando la propiedad "Copiar al directorio de salida" a "Copiar siempre".....	8
Imagen 8 Agregar referencia al proyecto.....	9
Imagen 9 Carpeta donde está SifeiSignature.....	9
Imagen 10 Referencia de biblioteca agregada .....	10
Imagen 11 Configuración de aplicación requerida para <i>SIFEI Signature</i> .....	11
Imagen 12 Seleccionado Archivo de Configuración de Aplicación .....	12
Imagen 13 Archivo de configuración limpio.....	12
Imagen 14 Archivo de configuración con la configuración necesaria para <i>SIFEI Signature</i> .....	13
Imagen 15 Ejecuta el <i>script</i> para la creación de la base de datos.....	13
Imagen 16 Consulta para mostrar el valor de <code>max_allowed_packet</code> .....	13
Imagen 17 Error 2006 MySQL server has gone away.....	14
Imagen 18 Variable <code>max_allowed_packet = 1MB</code> : causará "Error 2006: MySQL server has gone away".....	14
Imagen 19 Comando para cambiar de forma global el límite que define <code>max_allowed_packet</code> a 32 MB.....	14
Imagen 20 Cambiando variable <code>Max_allowed_packet</code> temporalmente (este efecto no cambia la sesión actual y dura solo mientras el servicio esté activo -hasta que el servicio sea reiniciado- ).....	15
Imagen 21 Archivo de configuración del servidor de MariaDB, cambio de 1mb a 16 MB (se debe reiniciar el servicio). 15	15
Imagen 22 Procedimiento para la ejecución del <i>script</i> , comenzando por el listado de la base de datos, la ejecución del <i>script</i> y finalizando por la comprobación de su creación.....	16
Imagen 23 Error de sintaxis en Mysql al importar el script de la base de datos catalogossat.....	17
Imagen 24 Vista del catalogossat que muestra <code>PAGE_CHECKSUM</code> y el numero de ocurrencias .....	17
Imagen 25 Catalogossat sin <code>PAGE_CHECKSUM</code> .....	18
Imagen 26 Captura que muestra de forma parcial las tablas creadas para la base de datos catalogossat en Mysql 5.5 y superior.....	19
Imagen 27 Captura de remplazar todas las ocurrencias de "ENGINE=Aria" por "ENGINE=InnoDB".....	20
Imagen 28 Captura de la importacion del script catalogossat con motor InnoDB.....	21
Imagen 29 Consulta de la variable <code>lower_case_table_names</code> donde su valor es 0, y por <b>tanto debe ser cambiado a 1</b> . 22	22
Imagen 30 Error al emitir un Cfdi cuando Mysql reside un servidor Unix y la variable de sistema <b><code>lower_case_table_names</code></b> tiene valor 0. ....	23
Imagen 31 Configurando el valor de <code>lower_case_table_names= 1</code> .....	24

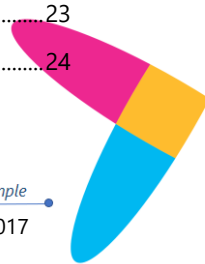


Imagen 32 Consulta de variable del sistema lower\_case\_table\_names , donde se muestra el cambio del valor de 0 a 1 (en un Centos) tras reiniciar Mysql Server, notar que mysql client pierde la conexión y se conecta de nuevo debido al reinicio de Mysql Server.....25

Imagen 33 Atributos de CFDI agregados.....26

Imagen 34 Nodo CfdiRelacionados agregado .....27

Imagen 35 CFDIRelacionado agregado .....27

Imagen 36 Nodo Emisor agregado.....27

Imagen 37 Nodo Receptor agregado .....28

Imagen 38 Nodo Concepto agregado .....28

Imagen 39 InformacionAduanera agregada a conceptos.....29

Imagen 40 Resultado de agregar un impuesto traslado al concepto.....29

Imagen 41 Nodo Impuestos agregado a comprobante .....30

Imagen 42 Impuesto traslado agregado.....30

Imagen 43 Autenticandose para el uso de Sifei Signature .....31

Imagen 44 Error de autenticación .....31

Imagen 45 Comprobante Sellado.....33

Imagen 46 Comprobante timbrado.....34

Imagen 47 Atributos de Retenciones agregados.....50

Imagen 48 Nodo Emisor agregado.....50

Imagen 49 Nodo Receptor agregado .....50

Imagen 50 Nodo Periodo agregado.....51

Imagen 51 Nodo Totales agregado.....51

Imagen 52 Nodo ImpRetenidos agregado.....51

Imagen 53 Comprobante Sellado.....53

Imagen 54 Comprobante timbrado.....54



## 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

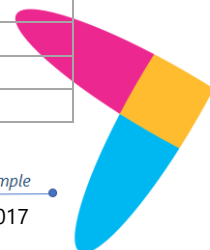
- ✚ Emitir CFDIs en versión 3.3.
- ✚ La certificación se realiza con base al Anexo 20 y a la *Matriz de Validaciones* de los CFDI y complementos.
- ✚ Entrega respuestas sobre los códigos de error de la *Matriz de Validaciones* vigente.

### 1.4 Requerimientos Mínimos

- ✚ Visual Studio 2012 o superior
- ✚ Tener instalado como mínimo .Net Framework 3.5
- ✚ Tener instalado una versión MariaDB (versión estable) 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.
  - Si se usara Mysql esta debe ser una version estable (5.1,5.5,5.6,5.7) y seguir los pasos de la sección [crear base de datos en Mysql](#)

### 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)	<a href="http://omawww.sat.gob.mx/tramitesyservicios/Paginas/anexo_20_version3-3.htm">http://omawww.sat.gob.mx/tramitesyservicios/Paginas/anexo_20_version3-3.htm</a>
Manual técnico de Cancelación Sifei.	<a href="https://www.sifei.com.mx/slides/slide/sifei-web-service-manual-tecnico-del-nuevo-servicio-de-cancelacion-99">https://www.sifei.com.mx/slides/slide/sifei-web-service-manual-tecnico-del-nuevo-servicio-de-cancelacion-99</a>

## 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 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
- Comprobante de Retención e Información de Pagos

### 1.9.1 Complementos para CFDI 3.3

- Comercio Exterior 1.1
- Nómina 1.2
- Detallista



- Pagos
- Impuesto Local
- IEDU

### 1.9.2 Complementos para Comprobante de Retención e Información de Pagos

- Servicios de Plataformas Tecnológicas

### 1.9.3 Métodos de Timbrado CFDI 3.3

- Recupera Timbre
- Deposita Factura Global
- Recupera Timbre Factura Global

### 1.9.4 Métodos de Timbrado Comprobante de Retención e Información de Pagos

- Recupera Timbre

### 1.9.5 Metodos de proceso de cancelación CFDI 3.3

- Cancela CFDI 2018 (Se recomienda utilizar el siguiente método Cancela 2018 Nuevo)
- Cancela 2018 Nuevo Proceso
- Consultar CFDI (Estado, Estado Cancelacion, etc)
- Consultar CFDI Relacionados
- Obtener Peticiones Pendientes
- Aceptacion/ Rechazo de Peticiones de cancelación.

## 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.

<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 parametros @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><b>Comentado [CM3]:</b> @pTotImpRetenidos y @pTotImpTrasladados son opcionales ya que llevan el operador de nulificación ? en su tipo primitivo</p>
---	---	--



### 3. Primeros pasos (*getting started*)

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

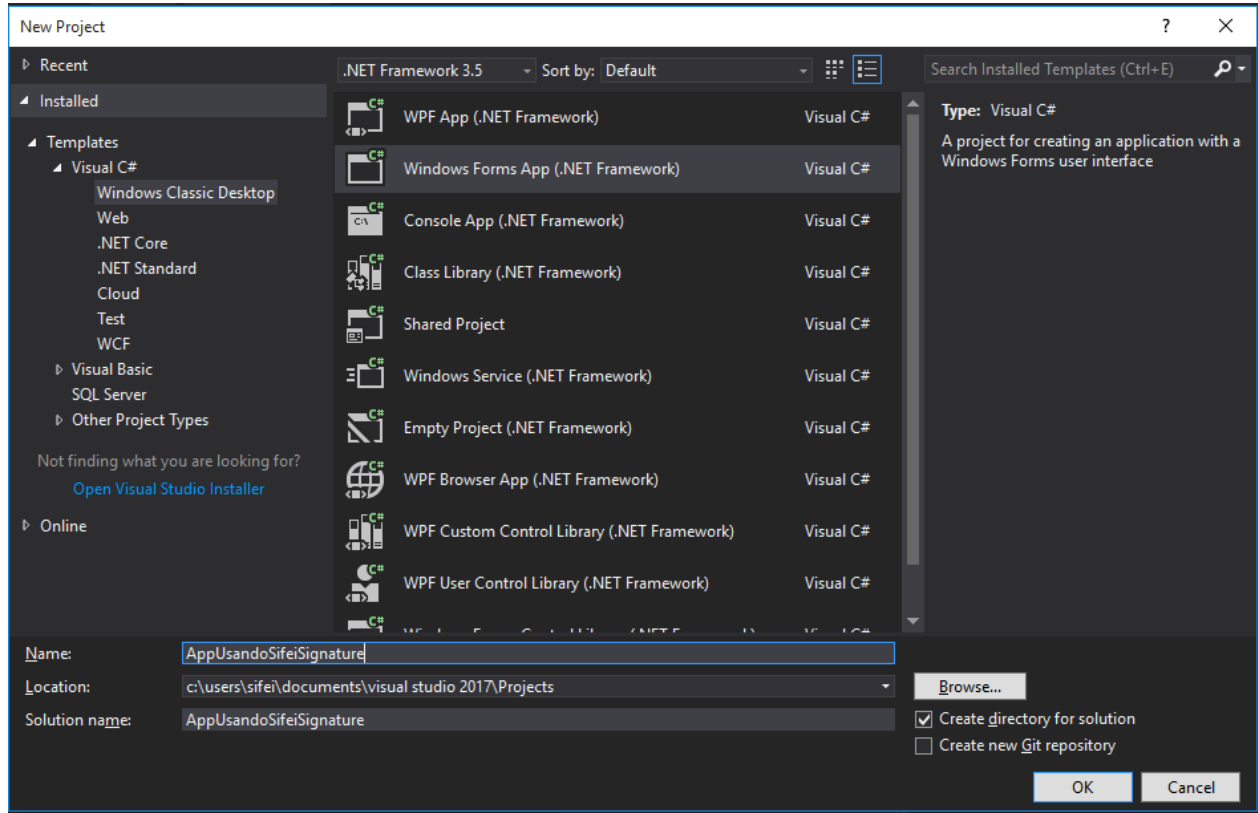


Imagen 1 Creación de proyecto .Net Framework 3.5

Lo cual debe generar algo similar a la siguiente imagen:

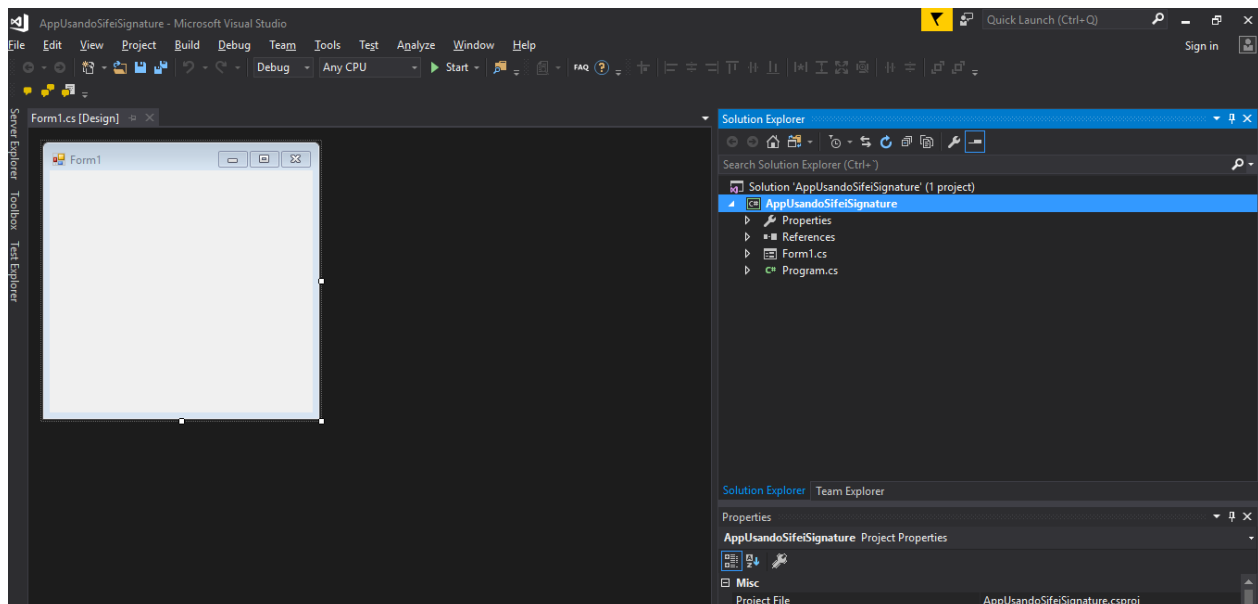


Imagen 2 Proyecto recién creado

### 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

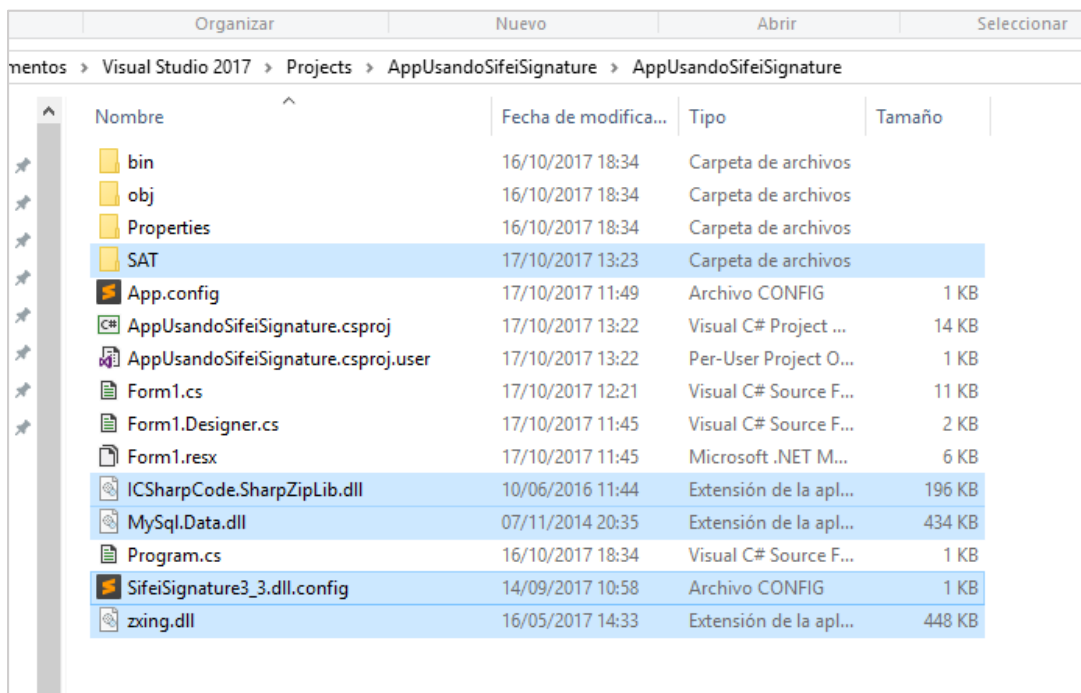


Imagen 3 Copiando los recursos al proyecto

#### 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).





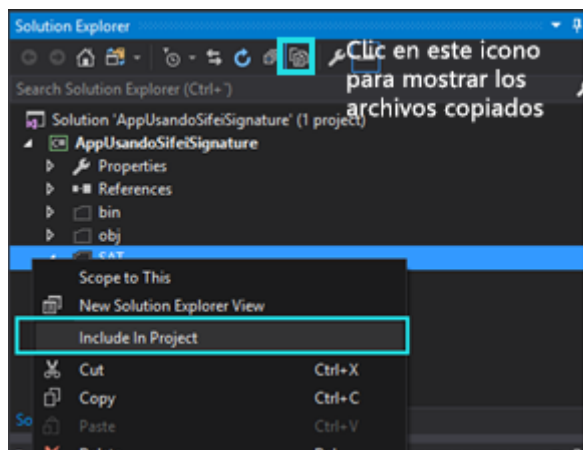


Imagen 4 Inclusión de carpeta SAT en el proyecto

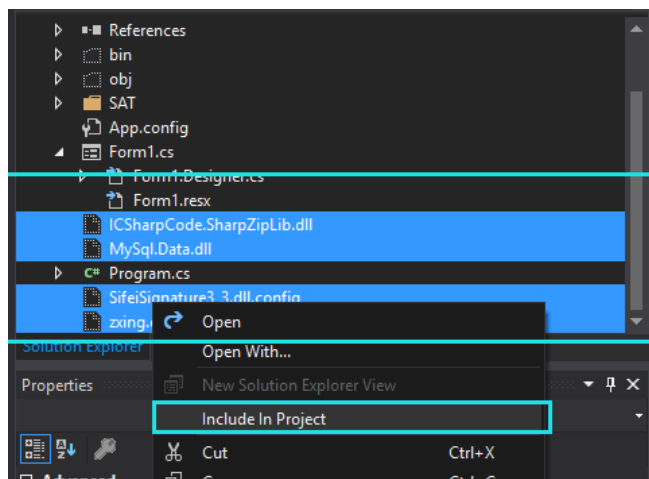


Imagen 5 Inclusión del resto de bibliotecas

### 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.



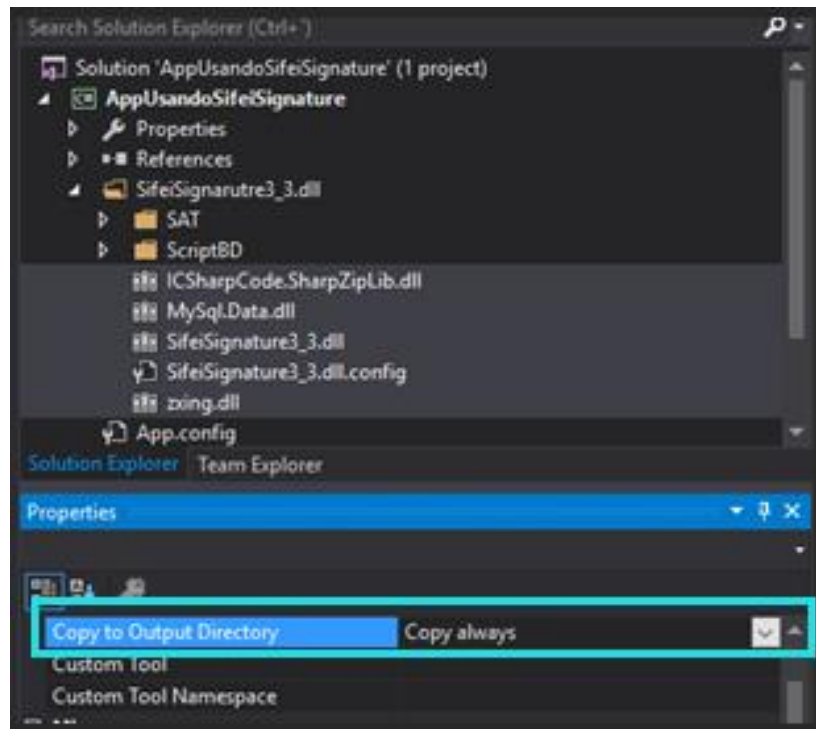


Imagen 6 Modificando la propiedad "Copiar al directorio de Salida"

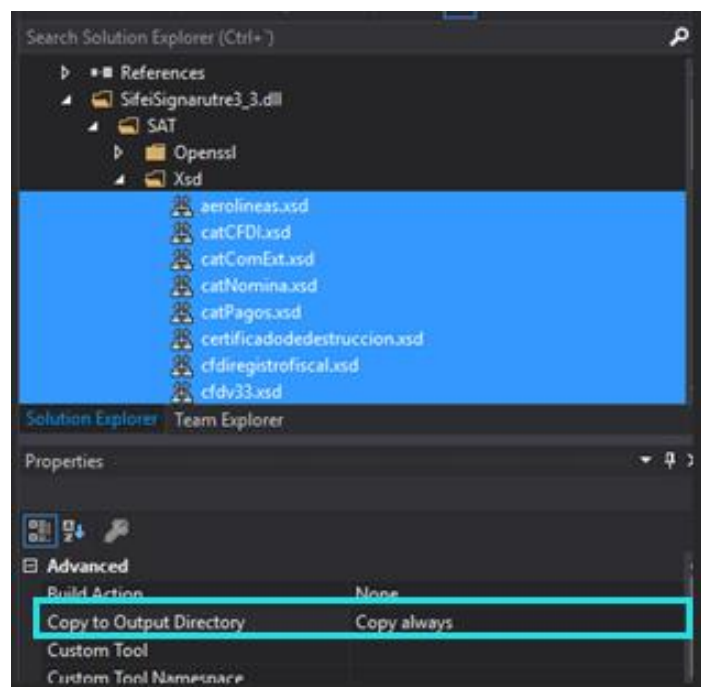


Imagen 7 Modificando la propiedad "Copiar al directorio de salida" a "Copiar siempre"

### 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).



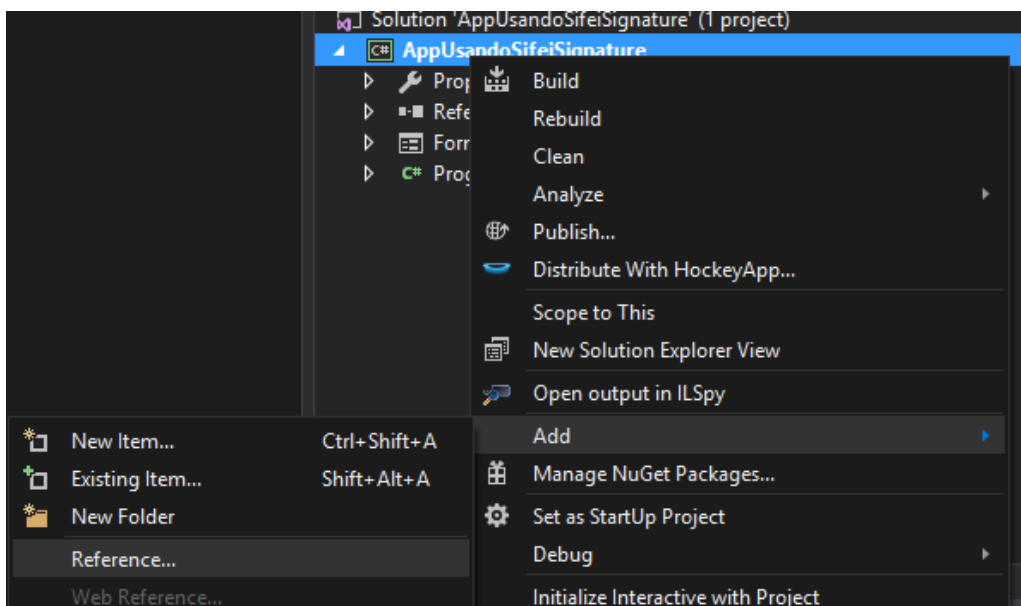


Imagen 8 Agregar referencia al proyecto

Se debe seleccionar la biblioteca "SifeiSignature3\_3.dll" y agregar.

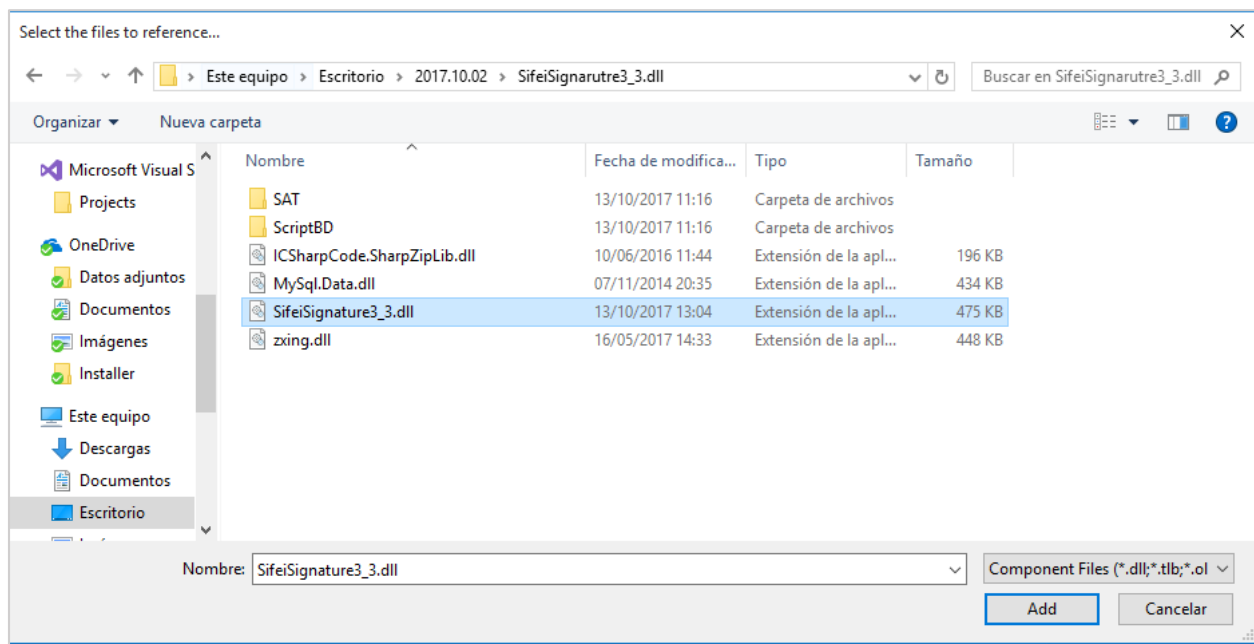


Imagen 9 Carpeta donde está SifeiSignature



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–.

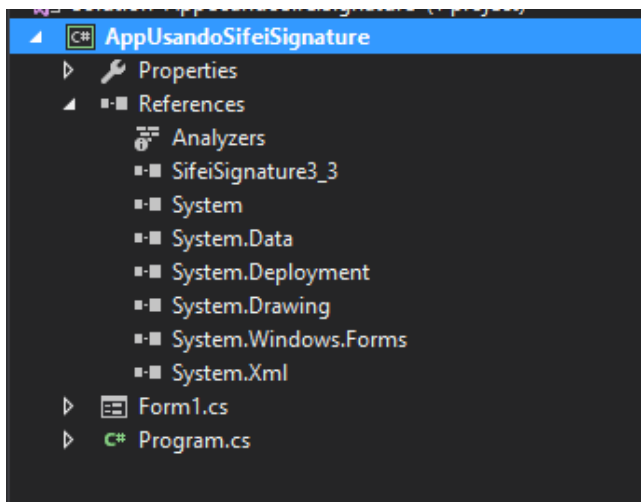


Imagen 10 Referencia de biblioteca agregada

### 3.3 Agregar información de los servicios web (configurar WS)

Para poder consumir los servicios a través de la DLL, es **necesario agregar los endpoint de los servicios**, para ello se debe agregar (si no existe) el siguiente nodo `system.serviceModel` al archivo de configuración App.config (como hijo de `configuration`). Donde deberá tener 5 nodos `endpoint` dentro del nodo `client` con sus respectivos `binding` en el nodo `basicHttpBinding`.



```
<system.serviceModel>
  <bindings>
    <basicHttpBinding>
      <binding name="SIFEIPortBinding" />
      <!-- Binding para nueva cancelacion -->
      <binding name="CancelacionPortBinding" />
      <binding name="BasicHttpBinding_IConsultaCFDIService">
        <security mode="Transport" />
      </binding>
      <binding name="SIFEIPortBinding1" />
      <binding name="WSseries-certificadosBinding" />
    </basicHttpBinding>
  </bindings>
  <client>
    <!--End point de timbrado, nota se controla con TipoTimbrado(en codigo) la URL. -->
    <endpoint address="http://192.168.16.251:8080/DESKTOP33/SIFEI" binding="basicHttpBinding"
bindingConfiguration="SIFEIPortBinding" contract="srvSifeiTimbrador.SIFEI" name="SIFEIPort" />
    <!--End point de cancelacion, nota se controla con TipoTimbrado(en codigo) la URL. -->
    <endpoint address="http://devcdfi.sifei.com.mx:8888/CancelacionSIFEI/Cancelacion"
binding="basicHttpBinding" bindingConfiguration="CancelacionPortBinding"
contract="srvSifeiCancelacion2018.Cancelacion" name="CancelacionPort" />
    <!--End point de consultaQr, nota se controla con TipoTimbrado(en codigo) la URL. -->
    <endpoint
address="https://consultaqr.facturaelectronica.sat.gob.mx/ConsultaCFDIService.svc"
binding="basicHttpBinding" bindingConfiguration="BasicHttpBinding_IConsultaCFDIService"
contract="srvSatConsultaQR.IConsultaCFDIService" name="BasicHttpBinding_IConsultaCFDIService" />
    <!--End point de Timbrado de Retenciones, nota se controla con TipoTimbrado(en codigo) la
URL. -->
    <endpoint address="http://192.168.16.251:8080/RETENCIONES/SIFEI"
binding="basicHttpBinding" bindingConfiguration="SIFEIPortBinding1"
contract="srvSifeiTimbradorRetenciones.SIFEI" name="SIFEIPort1" />
    <!--End point de Autenticación de Sifei Signature, nota se controla con TipoTimbrado(en
codigo) la URL. -->
    <endpoint address="http://192.168.16.101/cfdipac/server_WS/WSSerieCertif.php"
binding="basicHttpBinding" bindingConfiguration="WSseries-certificadosBinding"
contract="srvSifeiAuth.WSseriescertificadosPortType" name="WSseries-certificadosPort" />
  </client>
</system.serviceModel>
```

Imagen 11 Configuración de aplicación requerida para SIFEI Signature

Nota. Si se omite la configuración de algún servicio, una excepción de tipo `InvalidOperationException` será arrojada, ya que no encontrara el elemento endpoint correspondiente en el nodo `ServiceModel`, es necesario contar con estos 3 endpoint configurados.

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".



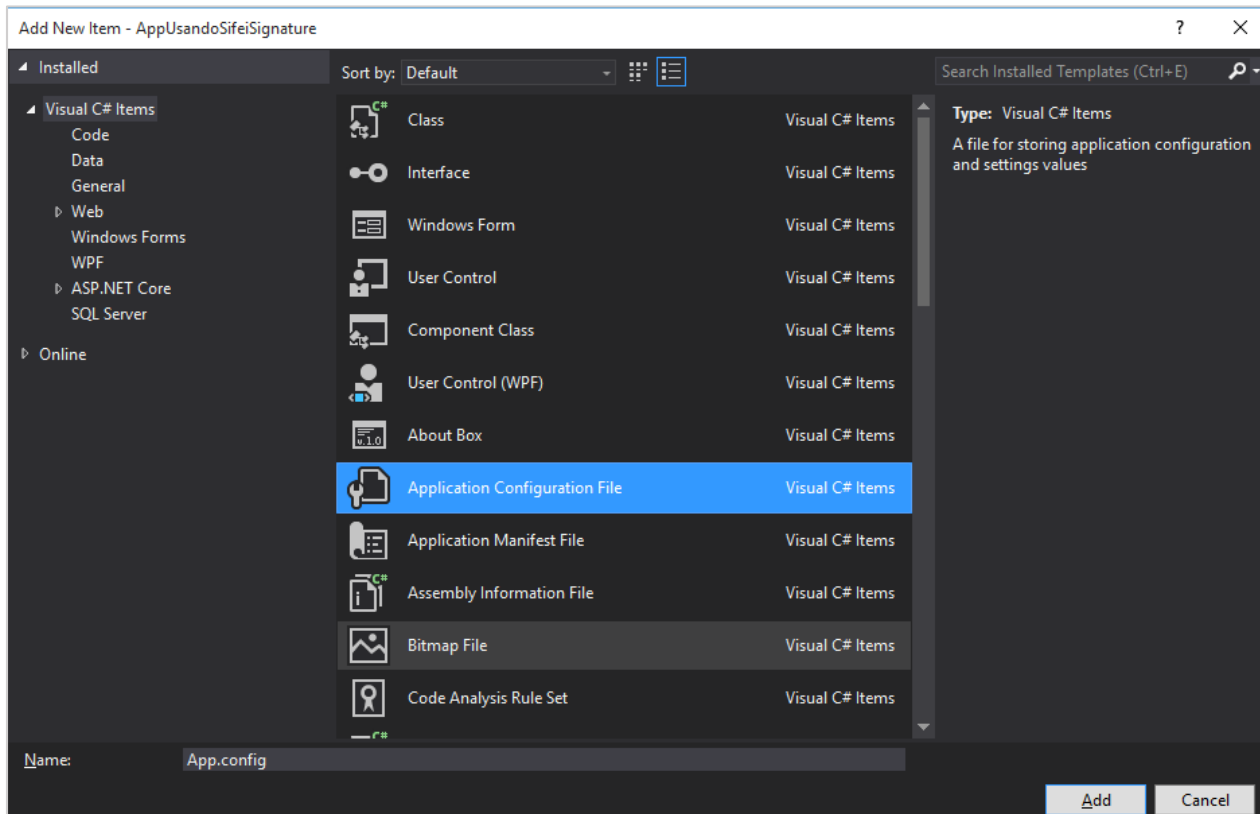


Imagen 12 Seleccionado Archivo de Configuración de Aplicación

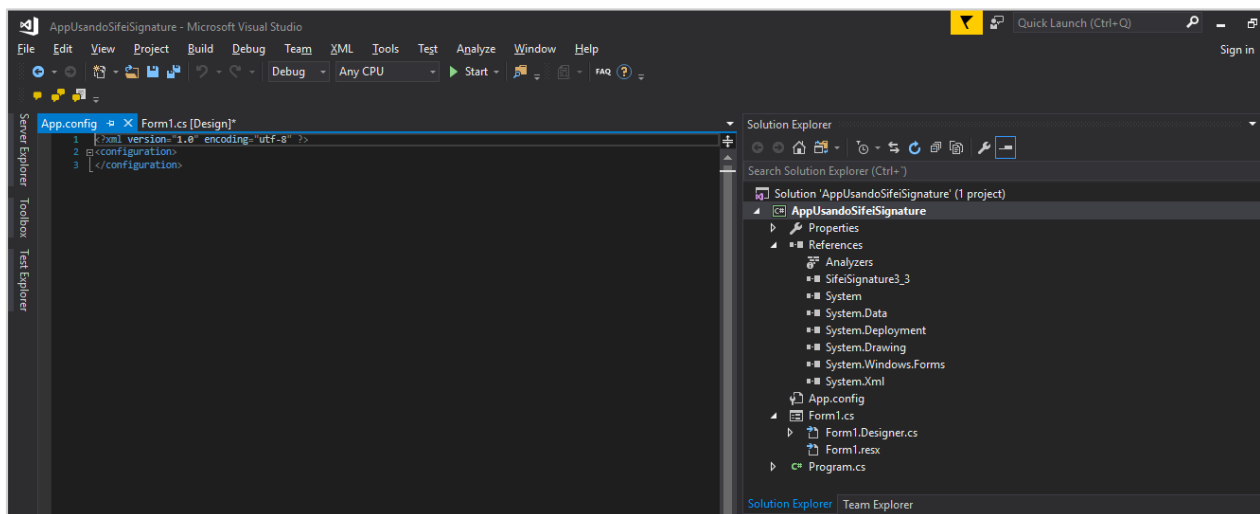


Imagen 13 Archivo de configuración limpio

Copiar el `system.servicemodel`.

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



```

ValidaPlatafor...cnologicas10.cs  App.config  Form1.cs [Diseño]  frmRetenciones.cs  frmRetenciones.cs [Diseño]
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <system.serviceModel>
    <bindings>
      <basicHttpBinding>
        <binding name="SIFEIPortBinding" />
        <binding name="CancelacionPortBinding" />
        <binding name="BasicHttpBinding_IConsultaCFDIService">
          <security mode="Transport" />
        </binding>
        <binding name="SIFEIPortBinding1" />
        <binding name="WSseries-certificadosBinding" />
      </basicHttpBinding>
    </bindings>
    <client>
      <endpoint address="http://192.168.16.251:8080/DESKTOP33/SIFEI"
        binding="basicHttpBinding" bindingConfiguration="SIFEIPortBinding"
        contract="srvSifeiTimbrador.SIFEI" name="SIFEIPort" />
      <endpoint address="https://pruebacfdiconsultaqr.cloudapp.net/ConsultaCFDIService.svc"
        binding="basicHttpBinding" bindingConfiguration="BasicHttpBinding_IConsultaCFDIService"
        contract="srvSatConsultaQR.IConsultaCFDIService" name="BasicHttpBinding_IConsultaCFDIService" />
      <endpoint address="http://devcfdi.sifei.com.mx:8888/CancelacionSIFEI/Cancelacion"
        binding="basicHttpBinding" bindingConfiguration="CancelacionPortBinding"
        contract="srvSifeiCancelacion2018.Cancelacion" name="CancelacionPort" />
      <endpoint address="http://192.168.16.251:8080/RETENCIONES/SIFEI"
        binding="basicHttpBinding" bindingConfiguration="SIFEIPortBinding1"
        contract="srvSifeiTimbradorRetenciones.SIFEI" name="SIFEIPort1" />
      <endpoint address="http://192.168.16.101/cfdipac/server_WS/WSerieCertif.php"
        binding="basicHttpBinding" bindingConfiguration="WSseries-certificadosBinding"
        contract="srvSifeiAuth.WSseriescertificadosPortType" name="WSseries-certificadosPort" />
    </client>
  </system.serviceModel>
</configuration>
  
```

Imagen 14 Archivo de configuración con la configuración necesaria para SIFEI Signature

### 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



### 3.4.4 Crear la base de datos en Mysql

**Nota:** Antes de seguir este apartado se debe leer el apartado [Crear base de datos](#), ya que en este se explica como importar la base de datos, los comandos y las configuraciones que pudiera requerir (validas para MariaDb y Mysql) , seguir los siguientes pasos tras tener configurado correctamente la variable [max\\_allowed\\_packet](#)

#### 3.4.4.1 Remover PAGE\_CHECKSUM=1

En el caso particular de Mysql, **se deben de hacer ligeras modificaciones al script** del catalogossat que se incluye, de lo contrario, un error como el que muestra en la imagen inferior ocurrirá:

```
F:\mysql-5.5.58-winx64\mysql-5.5.58-winx64\bin>mysql.exe -uroot --port=3308 <C:\Workspace\bd_catalogossat.sql
ERROR 1064 (42000) at line 23: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'PAGE_CHECKSUM=1' at line 7
F:\mysql-5.5.58-winx64\mysql-5.5.58-winx64\bin>
```

Imagen 23 Error de sintaxis en Mysql al importar el script de la base de datos catalogossat

Esto ocurre porque Mysql no reconoce PAGE\_CHECKSUM dentro de su sintaxis (es propia de MariaDB), así que para poder continuar debemos **remover todas** las ocurrencias de **'PAGE\_CHECKSUM=1'** que se encuentran dentro del **script**, para hacerlo rápido podemos apoyarnos de un editor través de la herramienta **"buscar y reemplazar"**, donde buscaremos **PAGE\_CHECKSUM=1** y reemplazaremos por vacío, es decir, **que no exista un carácter** en el cuadro de texto "Reemplazar con" (Replace with).

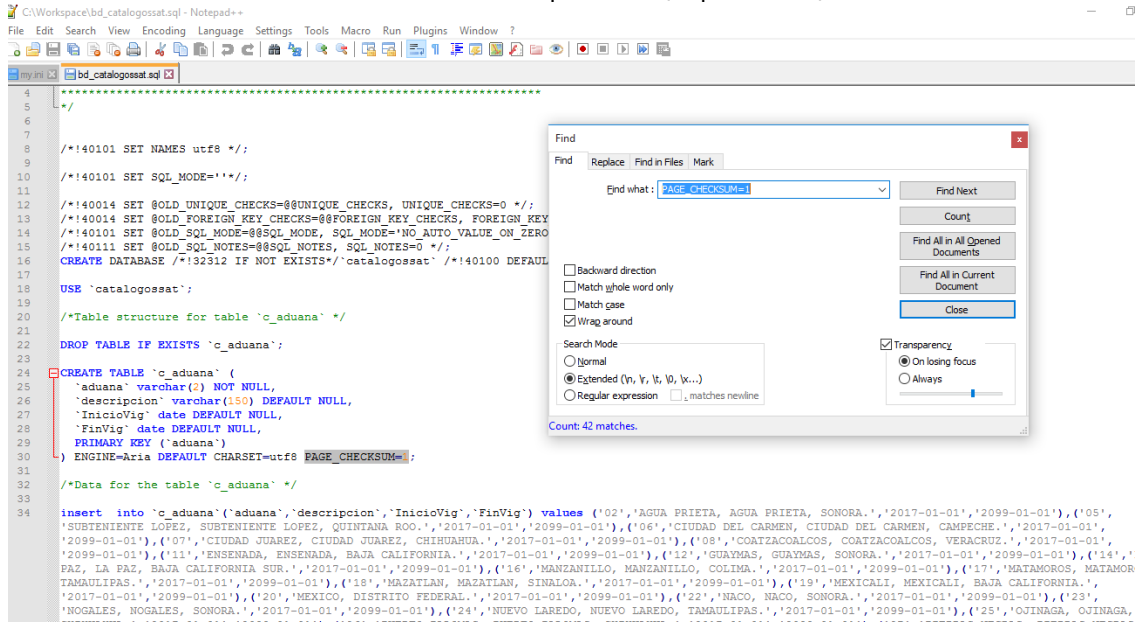


Imagen 24 Vista del catalogossat que muestra PAGE\_CHECKSUM y el numero de ocurrencias

Ahora basta con seleccionar **reemplazar a todo "Replace all"** y guardaremos el catalogo.



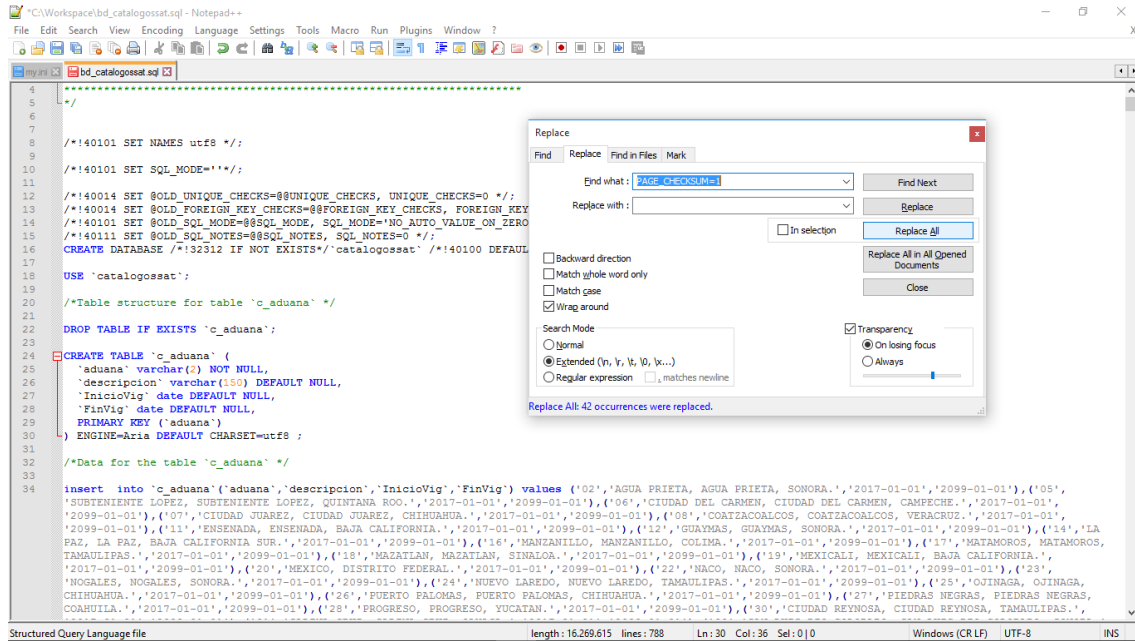
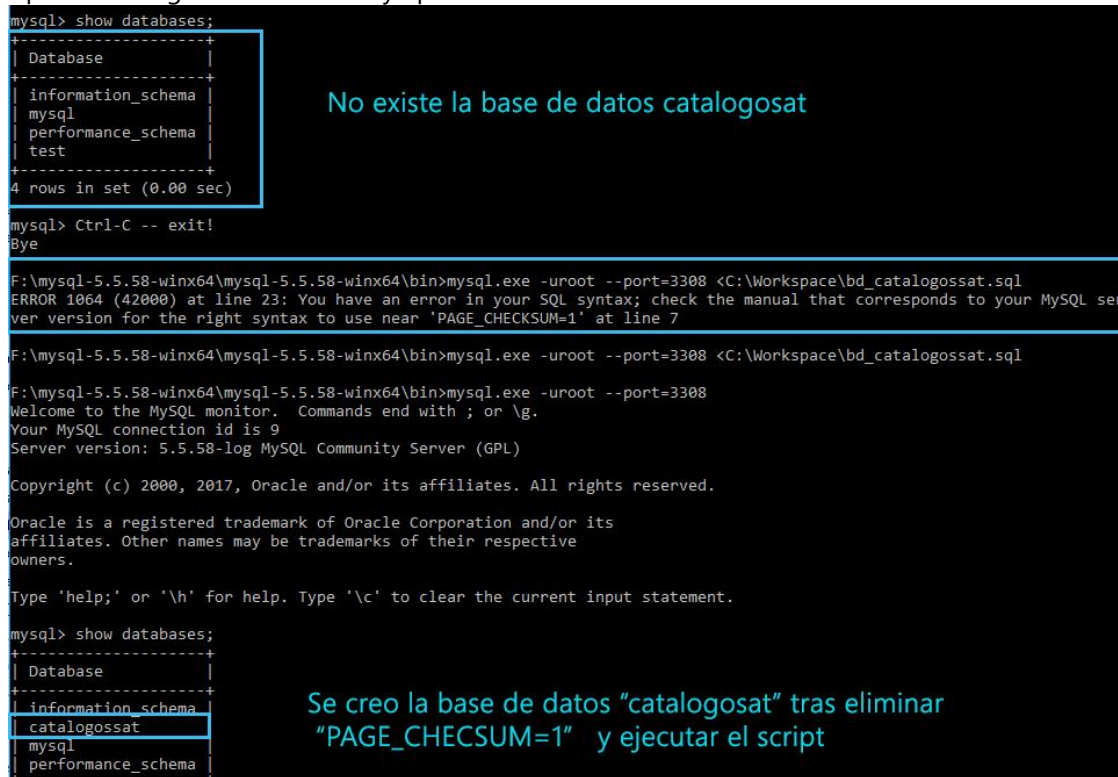


Imagen 25 Catalogosat sin PAGE\_CHECSUM

Ahora podemos regresar al cliente Mysql e intentarlo de nuevo.



Podemos consultar la información sobre las tablas de catalogosat si ejecutamos la siguiente instrucción:

```
Mysql> select table_schema,table_name,engine from information_schema.tables where table_schema='catalogosat';
```



```
mysql> select table_schema,table_name,engine from information_schema.tables where table_schema='catalogossat';
```

table_schema	table_name	engine
catalogossat	c_aduana	InnoDB
catalogossat	c_banco	InnoDB
catalogossat	c_clavepedimento	InnoDB
catalogossat	c_claveprodserv	InnoDB
catalogossat	c_claveunidad	InnoDB
catalogossat	c_codigopostal2017	InnoDB
catalogossat	c_colonia2017	InnoDB
catalogossat	c_estado	InnoDB
catalogossat	c_formapago	InnoDB
catalogossat	c_fraccionarancelaria2017	InnoDB
catalogossat	c_impuesto	InnoDB
catalogossat	c_incoterm	InnoDB
catalogossat	c_localidad	InnoDB
catalogossat	c_metodopago	InnoDB
catalogossat	c_moneda	InnoDB
catalogossat	c_motivotraslado2017	InnoDB
catalogossat	c_municipio	InnoDB
catalogossat	c_numpedimentoaduana	InnoDB
catalogossat	c_origenrecurso	InnoDB
catalogossat	c_pais	InnoDB
catalogossat	c_patenteaduana1	InnoDB
catalogossat	c_periodicidadpago	InnoDB
catalogossat	c_regimenfiscal2017	InnoDB
catalogossat	c_riesgopuesto	InnoDB
catalogossat	c_tasaocuota	InnoDB
catalogossat	c_tipocadenapago	InnoDB
catalogossat	c_tipocontrato	InnoDB
catalogossat	c_tipodecomprobante	InnoDB
catalogossat	c_tipodeduccion	InnoDB
catalogossat	c_tipofactor	InnoDB
catalogossat	c_tipohoras	InnoDB
catalogossat	c_tipoincapacidad	InnoDB
catalogossat	c_tipojornada	InnoDB
catalogossat	c_tiponmina	InnoDB
catalogossat	c_tipooperacion2017	InnoDB
catalogossat	c_tipootropago	InnoDB
catalogossat	c_tipopercepcion	InnoDB
catalogossat	c_tiporegimen	InnoDB
catalogossat	c_tiporelacion	InnoDB
catalogossat	c_unidadaduana2017	InnoDB

Imagen 26 Captura que muestra de forma parcial las tablas creadas para la base de datos catalogossat en Mysql 5.5 y superior

En este caso podemos notar que además de crear tablas, les asigno como motor InnoDB, esto ocurre porque en Mysql cuando un motor no esta compilado o desactivado (no esta disponible) Mysql tratara de usar el motor por defecto (InnoDB en Mysql 5.5 y superior, MYISAM como motor en 5.1) esto se explica en el siguiente párrafo de la documentación de Mysql:

*"If you try to use a storage engine that is not compiled in or that is compiled in but deactivated, MySQL instead creates a table using the default storage engine. This behavior is convenient when you want to copy tables between MySQL servers that support different storage engines."*

Ya que a partir de Mysql 5.5 el motor de base de datos por **defecto** es InnoDB en versiones iguales y superiores a esta el cambio a InnoDB sucede de forma transparente para el cliente(al menos que el motor por defecto sea diferente). Sin embargo en la misma documentación se indica que si en Mysql esta habilitado el modo sql [NO ENGINE SUBSTITUTION](#), un error ocurrirá y la tabla no será creada o alterada si el motor no está disponible:

*"This automatic substitution of the default storage engine for unavailable engines can be confusing for new MySQL users. A warning is generated whenever a storage engine is automatically changed. To prevent this from happening if the desired engine is unavailable, enable the [NO ENGINE SUBSTITUTION](#) SQL mode. **In this case, an error occurs instead of a warning and the table is not created or altered if the desired engine is unavailable"***



Por este motivo si ocurre un error al generar el catalogossat se deberá de deshabilitar el modo NO ENGINE SUBSTITUTION en el servidor Mysql, o bien como se propone a continuación **cambiar** en el script todas las ocurrencias de **Aria** por **InnoDB**.

### 3.4.4.2 Cambiar de motor de Aria por InnoDB en el script

Este método se propone como la solución más rápida y fácil, basta con reemplazar todas las ocurrencias de **ENGINE=Aria** por **ENGINE= InnoDB**, como se muestra a continuación:

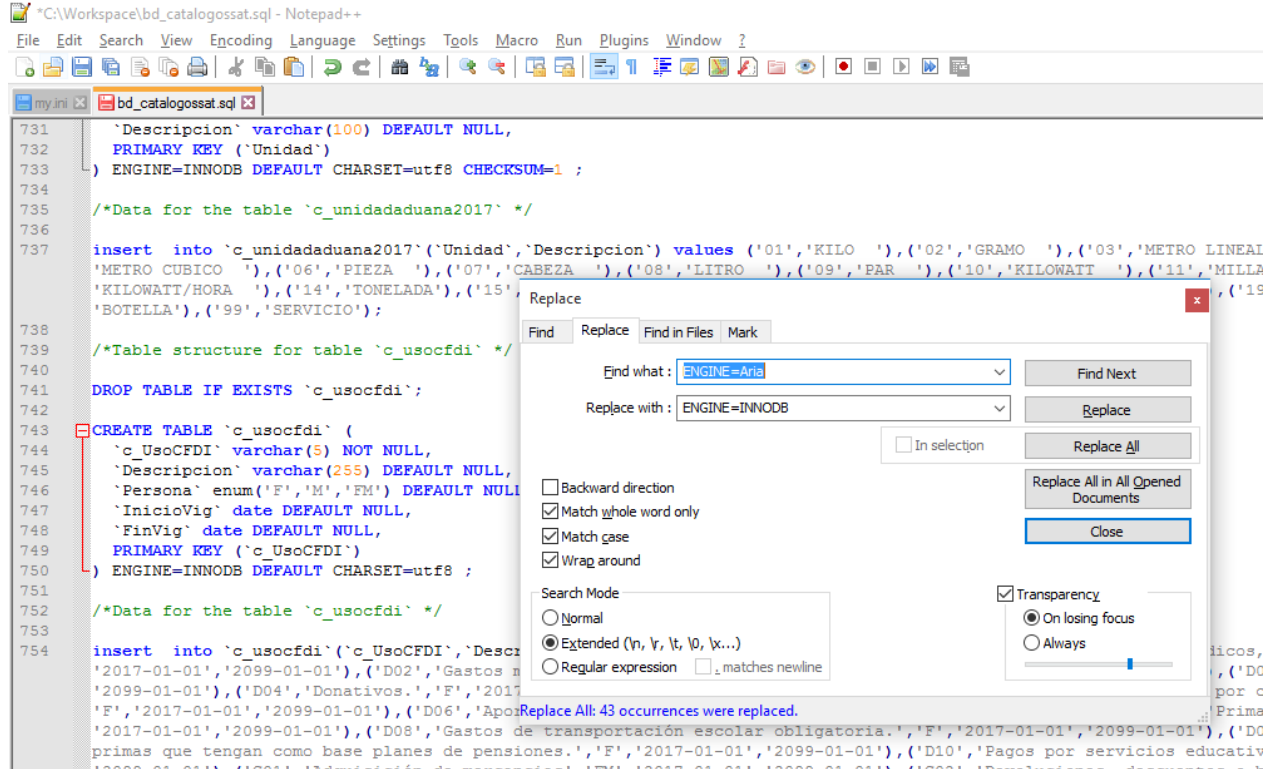


Imagen 27 Captura de reemplazar todas las ocurrencias de "ENGINE=Aria" por "ENGINE=InnoDB"

Se debe asegurar el cambio en el script antes de importarlo, se importa la base de datos y comprobamos que existe.



```
F:\mysql-5.1.73-win32\bin>mysql.exe -uroot --port=3309 <C:\Workspace\bd_catalogossat.sql

F:\mysql-5.1.73-win32\bin>mysql.exe -uroot --port=3309
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 5.1.73-community-log MySQL Community Server (GPL)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| catalogossat |
| mysql |
| test |
+-----+
4 rows in set (0.00 sec)

mysql> select table_schema,table_name,engine from information_schema.tables where table_schema='catalogossat';
+-----+-----+-----+
| table_schema | table_name | engine |
+-----+-----+-----+
| catalogossat | c_aduana | InnoDB |
| catalogossat | c_banco | InnoDB |
| catalogossat | c_clavepedimento | InnoDB |
| catalogossat | c_claveprodserv | InnoDB |
| catalogossat | c_claveunidad | InnoDB |
| catalogossat | c_codigopostal2017 | InnoDB |
| catalogossat | c_colonia2017 | InnoDB |
| catalogossat | c_estado | InnoDB |
| catalogossat | c_formapago | InnoDB |
| catalogossat | c_fraccionarancelaria2017 | InnoDB |
| catalogossat | c_impuesto | InnoDB |
| catalogossat | c_incoterm | InnoDB |
| catalogossat | c_localidad | InnoDB |
| catalogossat | c_metodopago | InnoDB |
+-----+-----+-----+
```

Imagen 28 Captura de la importación del script catalogossat con motor InnoDB

### 3.5 Mysql y MariaDb en Servidores Linux, Windows y Mac OS X

Ya habiendo ejecutado el script de la base de datos catalogossat, esto es, importar la base de datos, es importante tomar en cuenta el sistema operativo (SO) donde reside el Servidor Mysql o MariaDB, ya que afecta las configuración del servidor a través de las **variables de sistema** que pueden ser modificadas 1) en el archivos de configuración, ó 2) ser pasadas como parámetros a mysqld al iniciar el servidor.

Como nota los valores por defecto son aquellos que no se han definido en el archivo de configuración de Mysql —nombrado comunmente como **my.cnf**.

#### 3.5.1 Case sensitive (sensibilidad a minúsculas y mayúsculas)

La <<sensibilidad a minúsculas y mayúsculas>> (**case sensitive**) es una de las características que varia dependiendo del SO donde se instale Mysql/MariaDB, regida por la variable de sistema [lower case table names](#). En sistemas basados en unix el valor por defecto es 0(sensible), en Windows el valor por defecto es 1(insensible) y en OS X, el valor por defecto es 2. Es posible hacer que sistemas basados en unix se comporten como Windows y sean insensibles (como se ve en este apartado).

La siguiente tabla —extraída de la documentación de Mysql— muestra los valores posibles.

Valor	Descripcion
-------	-------------



0	Los nombre de tablas y base de datos son almacenados respetando el lettercase (tal cual estan escritas) de las sentencias the <a href="#">CREATE TABLE</a> or <a href="#">CREATE DATABASE</a> . La compacion de nombres es <b>sensible</b> a minisculas y mayusculas.
1	Los nombre de tablas son almacenadas en minisculas en el disco y las comparaciones de nombres son <b>insensibles</b> . Este comportamiento se aplica también a nombres de base de datos y alias de tablas. Mysql convierte todos los nombres de tabla a miniscula cuando se almacenan y realizan búsquedas,
2	Los nombres de tabla y base de datos son almacenados en disco usando el lettercase especificado en las sentencias <a href="#">CREATE TABLE</a> o <a href="#">CREATE DATABASE</a> pero Mysql convierte a minisculas cuando se hace una búsqueda. Los nombres de comparación son insensibles. Esto solo funciona en sistemas de archivos que son insensibles.

Nota: nunca cambiar el valor de **lower\_case\_table\_names** a 0 en sistemas insensibles como Windows(cuyo valor es 1).

Podemos consultar la variable con el siguiente comando:

```
Mysql> show variables where variable_name='lower_case_table_names';
```

Ejemplo de Servidor Mysql en un Centos 7



```
mysql> show variables where variable_name='lower_case_table_names';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| lower_case_table_names | 0 |
+-----+-----+
1 row in set (0.00 sec)

mysql>
```

el valor es 0 asi que las comparaciones son case sentive

Imagen 29 Consulta de la variable lowe\_case\_table\_names donde su valor es 0, y por **tanto debe ser cambiado a 1**

Por lo cual si no cambiamos el valor a 1 (comparaciones insensibles a Mayusculas, minúsculas) al usar la dll mostrara la siguiente excepción, ya que no encuentra la tabla **por que la comparación es sensible a minúsculas y mayusculas.**





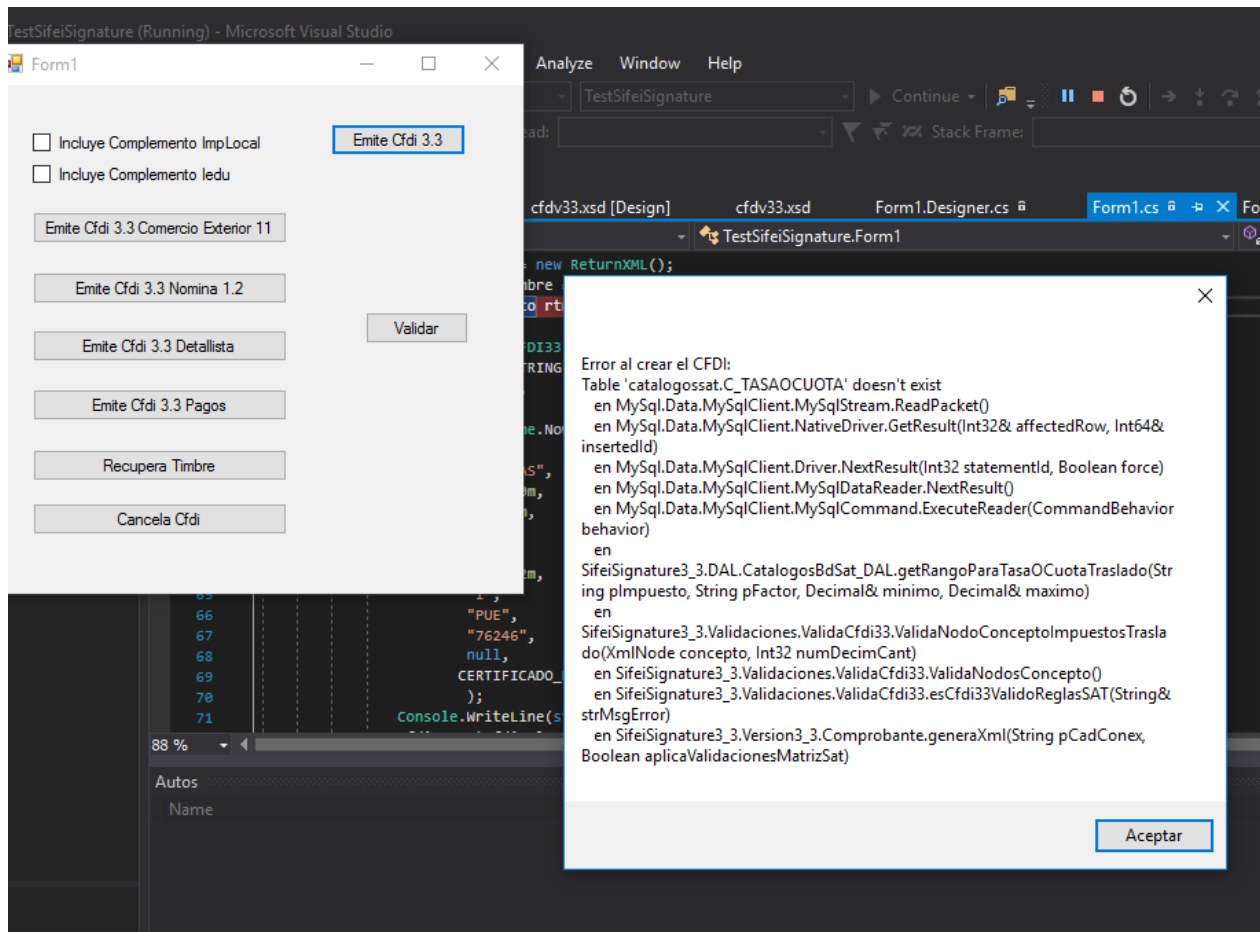
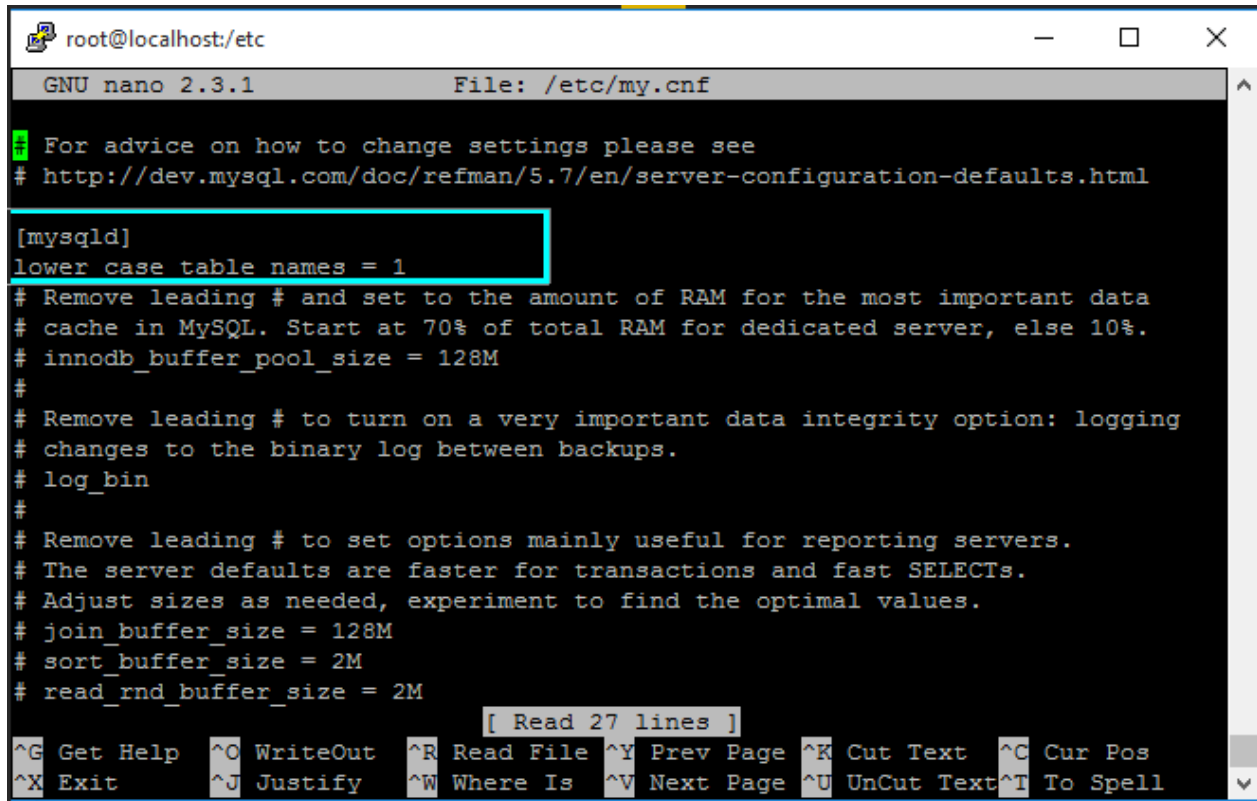


Imagen 30 Error al emitir un Cfdi cuando Mysql reside un servidor Unix y la variable de sistema **lower\_case\_table\_names** tiene valor 0.

### 3.5.2 Hacer insensible Mysql y MariaDb en sistemas basados en Unix.

Para hacer insensible la comparación de nombre de tablas se debe modificar el valor de `lower_case_table_names` a **1** en el archivo de configuración de Mysql/MariaDb tal y como se muestra a continuación (nota: es posible que la variable no este definida en el archivo, por lo cual debe agregarse).





```
root@localhost:/etc
GNU nano 2.3.1 File: /etc/my.cnf
# For advice on how to change settings please see
# http://dev.mysql.com/doc/refman/5.7/en/server-configuration-defaults.html

[mysql]
lower_case_table_names = 1
# Remove leading # and set to the amount of RAM for the most important data
# cache in MySQL. Start at 70% of total RAM for dedicated server, else 10%.
# innodb_buffer_pool_size = 128M
#
# Remove leading # to turn on a very important data integrity option: logging
# changes to the binary log between backups.
# log_bin
#
# Remove leading # to set options mainly useful for reporting servers.
# The server defaults are faster for transactions and fast SELECTs.
# Adjust sizes as needed, experiment to find the optimal values.
# join_buffer_size = 128M
# sort_buffer_size = 2M
# read_rnd_buffer_size = 2M

[ Read 27 lines ]
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Imagen 31 Configurando el valor de lowe\_case\_table\_names= 1

Ahora solo basta reiniciar Mysql (el comando varia entre distribuciones) y consultar que el valor de la variable haya cambiado a 1.



```

root@localhost:~
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> show variables where variable_name='lower_case_table_names';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| lower_case_table_names | 0 |
+-----+-----+
1 row in set (0.01 sec)

mysql> show variables where variable_name='lower_case_table_names';
ERROR 2006 (HY000): MySQL server has gone away
No connection. Trying to reconnect...
Connection id: 3
Current database: *** NONE ***

+-----+-----+
| Variable_name | Value |
+-----+-----+
| lower_case_table_names | 1 |
+-----+-----+
1 row in set (0.00 sec)

mysql>
    
```

Imagen 32 Consulta de variable del sistema lower\_case\_table\_names , donde se muestra el cambio del valor de 0 a 1 (en un Centos) tras reiniciar Mysql Server, notar que mysql client pierde la conexión y se conecta de nuevo debido al reinicio de Mysql Server.

Con esto ya es posible usar la base de datos en Servidores basados en Unix por la DII.



## 4. Ejemplos CFDI

A continuación, se presentan los ejemplos para emitir, sellar, timbrar y cancelar CFDIs y Comprobantes de Retención e Información de Pagos, además de recuperar timbres. Para mayor detalle sobre las clases, métodos y enumeraciones ver la sección 9 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 Catálogososat, el resto de los argumentos son los atributos que poseerá el CFDI.

```

01 cfdi33 = new CFDI33(
02     "Server=localhost;Port=3306;Database=catálogososat;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 33 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 34 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 35 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 36 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 37 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 38 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 39 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 trasladados 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 40 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 41 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>
    
```

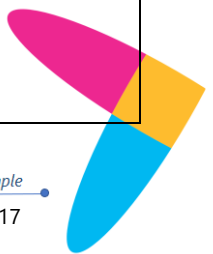
Imagen 42 Impuesto traslado agregado

#### 4.11 Autorización para el uso de Sifei Signature

Antes de utilizar los métodos para generar los complementos, generar el XML del comprobante CFDI 3.3 o de Retenciones e Información de Pagos, o sellar cualquiera de los tipos de comprobantes que emite Sifei Signature debe identificarse con los datos de acceso que le fueron asignados, ya sea en ambiente de pruebas o ambiente de producción y que son los mismos que se utilizan para timbrar. Esto se hace mediante el método `setIdenticacionSifei()`:

```

01 //identificándose para el uso de Sifei Signature
02         ret.setIdenticacionSifei(this.usuarioSifei, //Usuario Sifei
03                                 this.pwdSifei,      //Contraseña
04                                 this.tipoTimb);     //Tipo de timbrado
05
    
```





En caso de no haber invocado el método especificado, se generará la siguiente excepción:

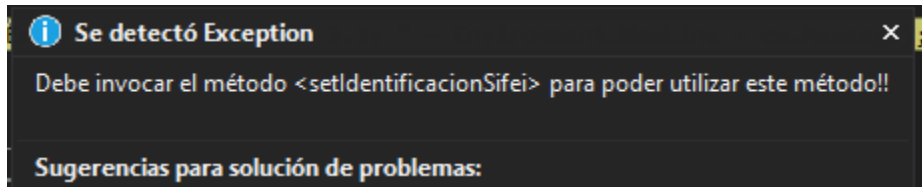


Imagen 43 Autenticandose para el uso de Sifei Signature

En caso de que los datos de usuario no estén correctos, se generará la siguiente excepción:

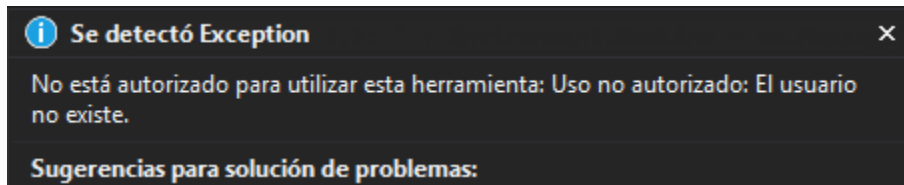


Imagen 44 Error de autenticación

## 4.12 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](#)).

Para utilizar este método antes debió invocar al método setIdentificacionSifei().

```

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/pdMhs4gN24v5rC3ad9AlGDVoMNs2gDeL1m+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.](#)



### 4.13 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\DEMO20170725123153.cer",
04 @"C:\Workspace\DatosBuzonMariaDB\Cert\DEMO20170725123153.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="f8rA2gcjnykVocsFL7BIQPjOXIbFDd0UFA9B8EcAX1U+yu36hm1R1pDPElFt8AINZ7uaJvedUu1HF0kqLib9mx/mCwVTTDF1/1YZPk95YoCuEr
6   FormaPago="02" NoCertificado="20001000000300022756"
7   Certificado="MIIF/zCCA+egAwIBAgIUMjAwMDEwMDAwMDAzMDAwMjI3NTYwDQYJKoZIhvcNAQELBQAwggFmMSAwHgYDVQQDBDBkBlkMuIDIgZGUGchJ1Z
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="G01" />
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" TasaCuota="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" TasaCuota="0.16" Importe="321.33" />
31 </cfdi:Traslados>
32 </cfdi:Impuestos>
33 </cfdi:Comprobante>
  
```

Imagen 45 Comprobante Sellado

Para utilizar este método antes debió invocar al método `setIdentificacionSifei()`.



#### 4.14 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("A",
03 "VPjftCMG1beV+/wkNwc8lkaFETxTrC/rt20oZgbnl30S6SD6kPAmLozt7CMkqzZK3L7+m1xZtXYDvu9oauJ1Wh
04 EWqSieNh96KCdNjdIXezM=",
05         rtnCfdi.XML);
06         //verificamos que haya sido timbrado:
07 if (rtnTimbre.IsTimbrado) {
08     //escribimos el cfdi timbrado
09     rtnTimbre.Xml.Save(@"C:\Workspace\SifeiSignature\cfdi_timbrado.xml");
10
11
  
```

Para utilizar este método antes debió invocar al método `setIdenticacionSifei()`. Es importante mencionar que este método ya no solicita el usuario y contraseña de timbrado, ni el ambiente en el que efectuará el timbrado (Pruebas o Producción), ya que estos son tomados de los valores que se le asignaron al invocar el método `setIdenticacionSifei()`.

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

```

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" xsi:schemaLocation="http://www.sat.gob.mx/cfd/3 http://www.sat.gob.mx/cfd/3" >
3   <cfdi:CfdiRelacionados TipoRelacion="04">
4     <cfdi:CfdiRelacionado UUID="2214C542-2FE6-45C5-BDA0-4C8EFF94CCFE" />
5     <cfdi:CfdiRelacionado UUID="806EA594-AB50-48C3-B56E-1EB0C4D570C8" />
6   </cfdi:CfdiRelacionados>
7   <cfdi:Emisor Rfc="MAR980114GQA" Nombre="UNION NACIONAL DE RESINEROS SA DE CV" RegimenFiscal="601" />
8   <cfdi:Receptor Rfc="CTR801211M15" Nombre="MARGARITA RODRIGUEZ TELLEZ" UsoCFDI="G01" />
9   <cfdi:Conceptos>
10    <cfdi:Concepto ClaveProdServ="01010101" Cantidad="8" ClaveUnidad="H87" Unidad="PZA" Descripcion="CALCIFLAVON (Tablets)" ValorUnitario="321.33">
11      <cfdi:Impuestos>
12        <cfdi:Traslados>
13          <cfdi:Traslado Base="2008.29" Impuesto="002" TipoFactor="Tasa" TasaOCuota="0.16" Importe="321.33" />
14        </cfdi:Traslados>
15      </cfdi:Impuestos>
16      <cfdi:InformacionAduanera NumeroPedimento="15 43 3740 6000564" />
17    </cfdi:Concepto>
18  </cfdi:Conceptos>
19  <cfdi:Impuestos TotalImpuestosRetenidos="0.00" TotalImpuestosTrasladados="321.33">
20    <cfdi:Traslados>
21      <cfdi:Traslado Impuesto="002" TipoFactor="Tasa" TasaOCuota="0.16" Importe="321.33" />
22    </cfdi:Traslados>
23  </cfdi:Impuestos>
24  <cfdi:Complemento>
25    <tfd:TimbreFiscalDigital xmlns:tfd="http://www.sat.gob.mx/TimbreFiscalDigital" xsi:schemaLocation="http://www.sat.gob.mx/TimbreFiscalDigital http://www.sat.gob.mx/TimbreFiscalDigital" >
26      Version="1.1" UUID="BBD670AA-9200-4AA2-944A-60B7D1DE0F3F" FechaTimbrado="2017-10-18T16:52:02"
27      RfcProvCertif="DAL050601L35" SelloCFD="f8rA2gcjnykVocsF17BIQPjOXIbFDd0UFA988EcAX1U+yu36hmLRlpDPElft8AINZ7uaJvedUu1HF0kqLi89mX/mCwVTTDFJ"
28      NoCertificadoSAT="20001000000300022323"
29      SelloSAT="o9X8srw45dIF0dLgKsk+2C9cvorCFkFvq4M+kuW/kOrJnC9JseP+Hx7heV4urDyTj+toVnbYFuilwbWg6fHxLLi0a4G/vx1n+mer5+sw22iIGM6G2qQtkYd6qocE"
30    </tfd:TimbreFiscalDigital>
31  </cfdi:Complemento>
  
```

Imagen 46 Comprobante timbrado





### 5.1.2 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
    
```

### 5.1.3 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
    
```

### 5.1.4 Agregar mercancía de Comercio Exterior

```

01 indexMerc = complCfdi.Cce11_addMercancia(
02     "A-RTGT-56", //pNoIdentificacion
03     "64041913", //pFraccionAranc
04     null, //pCantidadAduana
05     null, //pUnidadAduana
06     null, //pValorUnitAduana
07     795.96m); //pValorDolares
    
```

### 5.1.5 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](#)



## 5.2 Complemento Nómina

### 5.2.1 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
    
```

### 5.2.2 Agregar un emisor a Nómina

```

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

### 5.2.3 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
    
```

### 5.2.4 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
    
```





### 5.3.2 Establecer el requestForPaymentIdentification a Detallista

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

### 5.3.3 Agregar un specialInstruction a Detallista

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

### 5.3.4 Establecer el orderIdentification a Detallista

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

### 5.3.5 Agregar un referenceIdentification al AdditionalInformation de Detallista

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

### 5.3.6 Establecer un DeliveryNote a Detallista

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

### 5.3.7 Definir el Buyer (comprador) de Detallista

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

### 5.3.8 Definir el Seller (vendedor) a Detallista

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



### 5.3.9 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 );
    
```

### 5.3.10 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
    
```

### 5.3.11 Agregar un Customs a Detallista

```

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

### 5.3.12 Agregar un currency a Detallista

```

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

### 5.3.13 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
    
```

### 5.3.14 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 );
    
```



### 5.4.2 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
  
```

### 5.4.3 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
  
```

### 5.5 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 }
  
```

### 5.6 Complemento Impuestos Locales

#### 5.6.1 Definir el complemento Impuestos Locales

```

01 complCfdi.ImpLocal10_Define_ImpuestoLocal(
02                                     0.17m, //total traslado
03                                     0.00m); //total retencion
  
```



### 5.6.2 Agregar Impuesto Local Traslado al complemento

```

01 complCfdi.ImpLocal10_Agrega_TrasladoImpuestoLocal(
02     "ISH", //impuesto
03     0.17m, //importe
04     2.00m); //tasa
  
```

### 5.6.3 Generar el Complemento Impuestos Locales

```

01 ReturnXmlComplemento rtnXmlCompl = complCfdi. ImpLocal10_generaComplemento ();
02 //comprobamos que se creó el complemento
03 if (rtnXmlCompl.IsXmlComplementoCreated) {
04     //agregamos el complemento al concepto indicado por @indexConc
05     cfdi33.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
06     //agregamos el namespace de Iedu al comprobante
07     cfdi33.addNamespace(complCfdi. NsImpLocal10);
08 }
  
```

## 6. Proceso de Cancelación

### 6.1 Cancelar un CFDI 2018 (actualización)

Este método será eliminado en futura versión, por lo que se recomienda utilizar en su lugar el método `cancelaCfdisNuevoProceso()`

Este método permite iniciar la cancelación de uno o más CFDIs mediante el nuevo esquema de cancelación, para usarlo se debe configurar su respectivo "Endpoint", por lo que si ya se utilizaba la dll con anterioridad y no se ha implementado la llamada a este método se deberá de consumir este servicio mediante esta clase [Cancelacion2018](#) (Nota ver [sección configuración](#) para agregar su respectivo endpoint address). Para mayor detalle ver [Referencias](#).

```

01 Cancelacion2018 cancelacion2018 = new Cancelacion2018(
02     "JGE080430LN8", //Usuario Sifei
03     "0e84ec92", //Password Sifei
04     Cfdi33.tipoTimbrado.Pruebas); //Ambiente
05
06 string msg = "";
07 //Lista de uuids del comprobante a cancelar
08 List<string> uuids = new List<string>();
09 uuids.Add("XXXXXX-447C-4F10-BCD7-D1949CDF636CXXX");
10 try {
11     ReturnCancelacion2018 returnCancelacion2018 =
12     cancelacion2018.cancelaCfdis(ref msg,
13     uuids.ToArray(), //arreglo de Uuid
14     "MAR980114GQA",
15     CERTIFICADO_PATH,
16     CERTIFICADOKEY_PATH,
17     "12345678a"); //pass del certificado
18     if (!returnCancelacion2018.Procesado) {
19         Console.Write(returnCancelacion2018.CodigoError);
20         MessageBox.Show("Error al cancelar: " + returnCancelacion2018.CodigoError +
21         "detalle" + returnCancelacion2018.ErrorDetalle);
22     }
  
```



```

23     else {
24         //acuse detalle es una lista de objetos AcuseDetalle, uno por cada uuid mandado
25         a cancelar
26         foreach (var accuse in returnCancelacion2018.AcuseBE.AcuseDetalle) {
27             //codigo 201 indica que se realizó exitosamente ya sea para aprobación por
28             parte del receptor o si el comprobante se canceló sin autorización
29             if (accuse.Status == "201") {
30                 //se recomienda verificar con el metodo consulta CFDI
31                 }
32             }
33             XmlDocument xml = new XmlDocument();
34             xml.LoadXml(returnCancelacion2018.AcuseBE.AcuseXml);
35
36             xml.Save(WORK_DIR + "\\AcuseCancelacion" + DateTime.Now.ToString("yyyy-MM-
37             ddTHH.mm.ss") + ".xml");
38             MessageBox.Show("ok: " + returnCancelacion2018.AcuseBE.AcuseXml);
39             }
40         }
41     }
42     catch (Exception ex) {
43         MessageBox.Show("Error al cancelar: " + ex.Message);
44     }
45
    
```

Para mayor detalle ir a la sección 9 Referencia y a la sección ["Códigos de respuesta"](#).

## 6.2 Cancelar CFDIs Nuevo Proceso

Este método sustituye al método `cancelaCfdis()`

Este método realiza la cancelación de los CFDIs especificados, validando inicialmente el estatus del comprobante, sólo aquellos comprobantes que se encuentren en *Estado Vigente*, el valor de *EsCancelable* sea Cancelable (sin aceptación o con aceptación) y que su *EstatusCancelación* no sea En proceso, podrán ser procesados. Los UUIDs que no caigan en dichos supuestos se listarán en la respuesta del método, en el atributo *UuidsNoProcesados*.

```

01         Cancelacion2018 canc2018 = new Cancelacion2018(this.usuarioSifei, this.pwdS
02         ifei, this.tipoTimb);
03
04         List<estructuraConsultaQr> structUuids = new List<estructuraConsultaQr>();
05         estructuraConsultaQr uuid = new estructuraConsultaQr();
06         uuid.uuid = "67CC490A-1AE5-4D07-BCBB-542E17F8CF9D";
07         uuid.rfcReceptor = "TCM970625MB1";
08         uuid.total = 1334.09m;
09         uuid.selloCfd8 = "tBcd7w==";
10         structUuids.Add(uuid);
11         uuid = new estructuraConsultaQr();
12         uuid.uuid = "EA9BCCE8-3617-46E2-9FE0-C8EB0BBC25B9";
13         uuid.total = 1334.09m;
14         uuid.rfcReceptor = "TCM970625MB1";
15         uuid.selloCfd8 = "N4tOTQ==";
16         structUuids.Add(uuid);
17
18         string msg = "";
19
    
```

```

20         ReturnCancelacion2018 result = canc2018.cancelaCfdisNuevoProceso(ref msg, s
21         tructUuids.ToArray(), "S&S051221SE2",
22         @" CSD_S_&_SOFTWARE_S.A_DE_C.V_S&S051221SE2_20190627_135134s.cer",
23         @" CSD_S_&_SOFTWARE_S.A_DE_C.V_S&S051221SE2_20190627_135134.key",
24         "12345678a");
25
26         if (result.Procesado)
27         {
28             XmlDocument xmlAc = new XmlDocument();
29             xmlAc.LoadXml(result.AcuseBE.AcuseXml);
30             xmlAc.Save(@"acuseCanc.xml");
31         }
32         else
33         {
34             MessageBox.Show("Error al procesar cancelación: " + result.CodigoError
35             + " - " + result.ErrorDetalle);
36             return;
37         }
    
```

Es importante hacer notar que aunque la respuesta en su atributo **Procesado** sea **true**, se debe revisar el valor del atributo **UuidsNoProcesados**, ya que el proceso de cancelación realiza el proceso de cancelación del al menos 1 UUID, por ejemplo si envía a cancelar 5 UUIDs pero 3 de ellos son No Cancelables, el sistema procesa la cancelación de los 2 restantes.

### 6.3 Consulta CFDI relacionados

```

01 //instanciando clase cancelacion 2018 que permite consumir el metodo Consulta CFDI
02 relacionados
03     Cancelacion2018 cancelacion2018 = new Cancelacion2018(
04         "JGE080430LN8",           //Usuario Sifei
05         "0e84ec92",             //Password Sifei
06         CFDI33.tipoTimbrado.Pruebas); //Ambiente
07
08     string msg = "";
09     //UUID del CFDI a consultar, para saber si tiene CFDI relacionados y cuales
10     son.
11     string uuidCfdi = INEXISTENTE_UUID;
12
13     try {
14         ReturnDocumentosRelacionados returnCfdiRelacionados =
15         cancelacion2018.consultaDocumentosRelacionados(
16         Cancelacion2018.tipoConsultaRelacionados.Emisor, //Consultar por (Emisor,
17         Receptor)
18         "LAN7008173R5", //Emisor
19         "TCM970625MB1", //Receptor
20         CERTIFICADO_PATH,
21         CERTIFICADO_KEY,
22         "12345678a", //pass del certificado
23         uuidCfdi, //uuid del CFDI a consultar
24         ref msg);
25         if (!returnCfdiRelacionados.Procesado) {
26             Console.Write(returnCfdiRelacionados.CodigoError);
    
```

```

27     }
28     else {
29         //en caso de exito (procesado = true)
30         //se puede saber si tiene CFDI hijos o padres, el cual contiene UUID, RFC Emisor
31         y RFC Receptor
32         StringBuilder sb = new StringBuilder();
33         sb.AppendLine(string.Format("Numero de CFDI hijos:{0}",
34         returnCfdiRelacionados.Result.UuidsRelacionadosHijos.Length));
35         foreach (var padre in returnCfdiRelacionados.Result.UuidsRelacionadosHijos) {
36             sb.AppendLine( string.Format("UUID:{0}, RFC Emisor:{1}, RFC Receptor:{2}",
37             padre.Uuid, padre.RfcEmisor, padre.RfcReceptor));
38         }
39         sb.AppendLine(string.Format("Numero de CFDI Padre:{0}",
40         returnCfdiRelacionados.Result.UuidsRelacionadosHijos.Length));
41
42         foreach (var padre in returnCfdiRelacionados.Result.UuidsRelacionadosPadres) {
43             sb.AppendLine( string.Format("UUID:{0}, RFC Emisor:{1}, RFC
44         Receptor:{2}",padre.Uuid,padre.RfcEmisor,padre.RfcReceptor));
45         }
46         //guardando el acuse
47         XmlDocument xml = new XmlDocument();
48         xml.LoadXml(returnCfdiRelacionados.AcuseXml);
49         xml.Save(WORK_DIR+"\\AcuseCFDIRelacionados" + DateTime.Now.ToString("yyyy-MM-
50         ddTHH.mm.ss") + ".xml");
51         MessageBox.Show(sb.ToString());
52     }
53 }
54 }
55 catch (Exception ex) {
56     MessageBox.Show("Error al cancelar: " + ex.Message);
57 }
    
```

Para mayor detalle referirse a la sección 9.17 [códigos de error de CFDI Relacionados](#).

## 6.4 Consulta CFDI Estatus

```

01 Cancelacion2018 cancelacion2018 = new Cancelacion2018(
02     "JGE080430LN8",           //Usuario Sifei
03     "0e84ec92",             //Password Sifei
04     CFDI33.tipoTimbrado.Pruebas); //Ambiente
05 try {
06     // se consulta el metodo
07     ReturnQrAcuse returnQr = cancelacion2018.consultaEstatusCfdi(
08     uuid, //UUID
09     rfcEmisor, // RFC Emisor
10     rfcReceptor, //Rfc Receptor
11     total, // total
12     sello // sello ultimos 8 digitos
13 );
14
15     //Cancelable con aceptación|Cancelable sin aceptación
16     //S - Comprobante obtenido satisfactoriamente|
17     if (returnQr.Procesado) {
18         //Cancelable sin aceptación|
19         if (returnQr.EsCancelable == "Cancelable sin aceptación") {
20
21         }
22         if (returnQr.Estado == "Cancelado") { //"Vigente", "Cancelado","No Encontrado"
23             //Cancelado sin aceptación
    
```



```

24     Console.WriteLine("", returnQr.EstatusCancelacion);
25     }
26
27     XmlDocument xml = new XmlDocument();
28     xml.LoadXml(returnQr.AcuseXml);
29     xml.Save(WORK_DIR + "\\AcuseConsultaEstatus" + DateTime.Now.ToString("yyyy-MM-
30     ddTHH.mm.ss") + ".xml");
31     MessageBox.Show(returnQr.AcuseXml);
32     }
33
34 }
35 catch (Exception ex) {
36     MessageBox.Show(string.Format("Error al consultar CFDIS {0}: ",
37     ex.Message));
38 }
    
```

## 6.5 Consulta peticiones de Cancelación pendientes (que requieren autorización).

```

01 /*En el rol de rfc receptor, con este metodo se obtienen las peticiones pendientes
02 de Cancelacion que requieren la autorizacion
03 * del RFC como Receptor (aceptacion) para aprobar o rechazar la cancelacion de
04 un CFDI.
05 * */
06 Cancelacion2018 cancelacion2018 = new Cancelacion2018(
07     "JGE080430LN8", //Usuario Sifei
08     "0e84ec92", //Password Sifei
09     CFDI33.tipoTimbrado.Pruebas); //Ambiente
10
11 string msg = "";
12 try {
13     //se obtiene las peticiones pendientes de cancelacion al RFC Receptor:
14     ReturnPeticionesPendientes returnPeticionesPendientes =
15     cancelacion2018.getPeticionesCancelacionPendientes(
16     "MAR980114GQA" //RFC receptor quien aprueba o rechaza la cancelacion del CFDI
17 );
18     //se verifica si fue procesado es false lo que indicaria que hubo un error a
19 nivel de conxion, o excepcion a nivel de WebService)
20     if (!returnPeticionesPendientes.Procesado) {
21         MessageBox.Show("Codigo de error:" + returnPeticionesPendientes.CodigoError + "Error
22 detalle: " + returnPeticionesPendientes.ErrorDetalle);
23     }
24     //en caso de exito
25     else {
26         PrepareListViewForPeticionesPendientes();
27         uuidPeticionesPendientes.Clear();
28         // 1100 Se obtuvo petición del RFC receptor de forma exitosa
29         if (returnPeticionesPendientes.CodEstatus == "1100") {
30             //si el codigo es 1100, entonces recorrer el UUID de cada peticion.
31             foreach (var uuid in returnPeticionesPendientes.UUID) {
32                 //en este ejemplo se usa un ListView para mostrar el UUID, pero es posible que
33 se desee guardar a BD como registro
34                 listView2.Items.Add(new ListViewItem(uuid));
35                 uuidPeticionesPendientes.Add(uuid);
36             }
37             //guardadndo el acuse a archivo(ejemplo)
38             XmlDocument xml = new XmlDocument();
    
```

```

39         xml.LoadXml(returnPeticionesPendientes.AcuseXml);
40
41     xml.Save(WORK_DIR+"\\AcusePeticionesPendientes_"+DateTime.Now.ToString("yyyy-MM-
42     ddTHH.mm.ss")+ ".xml");
43     }
44     //1101 no existen peticiones pendientes
45     else if (returnPeticionesPendientes.CodEstatus == "1101") {
46         Console.WriteLine("No hay peticiones pendientes");
47     }
48     }
49     }
50     }
51     catch (Exception ex) {
52         MessageBox.Show("Error al obtener peticiones Pendinetes de Cancelacion: " +
53         ex.Message);
    }

```

Para mayor detalle referirse a la sección códigos de error consulta [Peticiones de Cancelacion pendientes](#).

## 6.6 Procesar Respuesta de Peticion de Cancelacion (Aceptacion o Rechazo de cancelación)

```

01 Cancelacion2018 cancelacion2018 = new Cancelacion2018(
02         "JGE080430LN8",           //Usuario Sifei
03         "0e84ec92",             //Password Sifei
04         CFDI33.tipoTimbrado.Pruebas); //Ambiente
05     //lista de objeto folios requerido por el servicio, incluye el UUID y la
06     respuesta:aceptacion o rechazo
07     List<folios> lstFolios = new List<folios>();
08     //Se establecen los UUID de las peticiones
09     List<string> uuidSeleccionados= getUUIDSeleccionados();
10
11     foreach (string uuid in uuidSeleccionados) {
12         //se crea objeto clase folio , que requiere el UUID y la respuesta
13         folios f = new folios();
14         //se asigna el UUID de la peticion a la que se le dara respuesta
15         f.uuid = uuid;
16         //se asigna la respuesta, que es donde se indica si se acepta o rechaza
17         f.respuesta = respuestaParaUUIDMarcados;
18         //se indica que el valor respuesta no es uno por defecto,sino que de lado del
19         cliente, realmente se esta estableciendo este valor
20         f.respuestaSpecified = true;
21         lstFolios.Add(f);//agregamos la respuesta
22     }
23     string msg="";
24     try {
25         // con este metodo, como receptor se acepta o rechaza
26         ReturnAceptacionRechazoResult returnAceptacionRechazoResult =
27         cancelacion2018.procesaRespuestaPeticionesCancelacion(
28         "MAR980114GQA",
29         CERTIFICADO_PATH,
30         CERTIFICADO_KEY_PATH,
31         "12345678a", //pass del certificado
32         ref msg,
33         lstFolios.ToArray());
34
35         if (!returnAceptacionRechazoResult.Procesado) {
36             MessageBox.Show("Codigo de error:" + returnAceptacionRechazoResult.CodigoError +
37             "Error detalle: " + returnAceptacionRechazoResult.ErrorDetalle);
38

```

```

39     }
40     //en caso de ok
41     if (returnAceptacionRechazoResult.Procesado) {
42     if (returnAceptacionRechazoResult.CodEstatus == "1000") {
43         Console.WriteLine(returnAceptacionRechazoResult.AcuseXML);
44     }
45     XmlDocument xml = new XmlDocument();
46     xml.LoadXml(returnAceptacionRechazoResult.AcuseXML);
47     xml.Save(WORK_DIR + "\\AcuseProcesaRespuestaAceptacionRechazo_" +
48     DateTime.Now.ToString("yyyy-MM-ddTHH.mm.ss") + ".xml");
49     }
50     }
51     }
52     catch (Exception ex) {
53         MessageBox.Show("Error al procesar respuesta: " + ex.Message);
54     }
55     }
    
```

Para mayor detalle referirse a la sección [códigos de error del metodo Aceptacion/Rechazo de Cancelacion](#).

## 7. Comprobante de Retención de Información de Pagos

### 7.1 Instanciar un objeto de Comprobante de Retención e Información de Pagos

La clase retención es la clase principal para generar Comprobantes de Retención e Información de Pagos, para poder utilizar esta clase, se debe pasar como parámetro la cadena de conexión a la base de datos Catálogosat.

```

01 retencion ret = new retencion(
02     "Server=localhost;Port=3306;Database=catálogosat;Uid=root;pwd=123456;",
03     DateTime.Now);
    
```

### 7.2 Definir la retención

Haciendo uso de la clase `retencion` que se inicializó, se invoca a los métodos necesarios para generar el comprobante. El primer paso es definir los parámetros generarles del comprobante utilizando el método `Reten_Define_Retenciones()`:

```

01 ret.Reten_Define_Retenciones(
02     this.rutaCert,           //pRutaCert
03     "DEMO276",             //pFolioInt
04     DateTime.Now,          //pFechaExp
05     "26",                  //pCveRetenc
06     ""                      //pDescRetenc
07     );
08
    
```

Lo cual produce un XML con lo siguiente:



```
<?xml version="1.0" encoding="utf-8"?>
<retenciones:Retenciones xmlns:plataformasTecnologicas="http://www.sat.gob.mx/esquemas/retencionpago/1/P
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.sat.gob.mx/esquemas/retencionpago/1 http://www.sa
Version="1.0"
FolioInt="DEMO276"
Sello="@@"
NumCert="20001000000300022815"
Cert="MIIFxTCCA62gAwIBAgIUMjAwMDEwMDAwMDAwMDAwMjI4MTUwDQYJKoZIhvcNAQELBQAwggFmMS
FechaExp="2019-08-07T14:19:25-05:00"
CveRetenc="26"
xmlns:retenciones="http://www.sat.gob.mx/esquemas/retencionpago/1">
```

Imagen 47 Atributos de Retenciones agregados

### 7.3 Definir el Emisor

Para establecer los datos del Emisor del comprobante se debe utilizar el método Reten\_Set\_Emisor().

```
01 ret.Reten_Set_Emisor(this.rfcEmisor, //pRFCEmisor
02 "FRIOS Y CONGELADOS DEL SURESTE SA DE CV", //pNombDenRazSocE
03 "", //pCURPE
04 );
```

El nodo generado es Emisor:

```
xmlns:retenciones="http://www.sat.gob.mx/esquemas/retencionpago/1">
<retenciones:Emisor RFCEmisor="LAN7008173R5" NomDenRazSocE="FRIOS Y CONGELADOS DEL SURESTE SA DE CV" />
<retenciones:Receptor Nacionalidad="Nacional">
```

Imagen 48 Nodo Emisor agregado

### 7.4 Definir Receptor

Para establecer los datos de Receptor, se puede utilizar el método Reten\_Set\_ReceptorNacional() o Reten\_Set\_ReceptorExtranjero() según sea el caso, en el siguiente ejemplo se utiliza el primer método:

```
01 ret.Reten_Set_ReceptorNacional(this.rfcReceptor, //pRFCRecep
02 "CONSTANTINO GARCIA PONTON", //pNombDenRazSocR
03 "GAPC530722HDFRNN04" //pCURPR
04 );
```

El nodo generado es Receptor:

```
<retenciones:Emisor RFCEmisor="LAN7008173R5" NomDenRazSocE="FRIOS Y CONGELADOS DEL SURESTE SA DE CV" />
<retenciones:Receptor Nacionalidad="Nacional">
  <retenciones:Nacional RFCRecep="MAR980114GQA" NomDenRazSocR="CONSTANTINO GARCIA PONTON" CURPR="GAPC530722HDFRNN04" />
</retenciones:Receptor>
<retenciones:Periodo MesIni="7" MesFin="7" Ejerc="2019" />
```

Imagen 49 Nodo Receptor agregado

### 7.5 Definir el Periodo

Para establecer el periodo del comprobante se debe utilizar el método Reten\_Set\_Periodo():

```
01 ret.Reten_Set_Periodo(7, //mesIni
02 7, //mesFin
03 2019); //ejerc
04
```

El nodo generado es Periodo:

```
<retenciones:Nacional RFCRecep="MAR980114GQA" NomDenRazSocR="CONSTANTINO GARCIA"  

</retenciones:Receptor>  

<retenciones:Periodo MesIni="7" MesFin="7" Ejerc="2019" />  

<retenciones:Totales montoTotOperacion="1001.00" montoTotGrav="1001.00" montoTotExe  

<retenciones:ImpRetenidos BaseRet="1001.00" Impuesto="02" montoRet="80.08" TipoP
```

Imagen 50 Nodo Periodo agregado

## 7.6 Definir Totales

Para establecer los totales del comprobante de Retenciones e Información de Pagos, se debe utilizar el método `Reten_Set_Totales()`:

```
01 ret.Reten_Set_Totales(1001.00m, //pmontoTotOperacion  

02 1001.00m, //pmontoTotGrav  

03 0.00m, //pmontoTotExent  

04 80.08m); //pmontoTotRet
```

Lo cual genera el siguiente nodo Totales:

```
<retenciones:Periodo MesIni="7" MesFin="7" Ejerc="2019" />  

<retenciones:Totales montoTotOperacion="1001.00" montoTotGrav="1001.00" montoTotExent="0.00" montoTotRet="80.08">  

<retenciones:ImpRetenidos BaseRet="1001.00" Impuesto="02" montoRet="80.08" TipoPagoRet="Pago definitivo" />
```

Imagen 51 Nodo Totales agregado

## 7.7 Agregar Impuesto Retenido

Para agregar un impuesto retenido se debe utilizar el método `Reten_Add_ImpRetenidos()`:

```
01 ret.Reten_Add_ImpRetenidos(1001.00m, //pBaseRet  

02 "02", //pImpuesto  

03 80.08m, //pmontoRet  

04 retencion.tipoPago.Definitivo); //pTipoPagoRet
```

Lo cual genera el nodo `ImpRetenidos`:

```
<retenciones:Totales montoTotOperacion="1001.00" montoTotGrav="1001.00" montoTotExent="0.00" montoTotRet="80.08">  

<retenciones:ImpRetenidos BaseRet="1001.00" Impuesto="02" montoRet="80.08" TipoPagoRet="Pago definitivo" />  

</retenciones:Totales>
```

Imagen 52 Nodo ImpRetenidos agregado

## 7.8 Autorización para el uso de Sifei Signature

La clase `retencion` incluye el método `setIdenticacionSifei()` que debe invocarse antes de invocar a los métodos `generaXmlRetencion()` y `sellaRetencion()`, referirse a la misma sección para CDFI 3.3.

## 7.9 Generar el XML

Para generar el `XmlDocument` debemos invocar al método `Reten_GeneraXML()`, 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](#)).



Para utilizar este método antes debió invocar al método `setIdenticacionSifei()`.

```

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 rtnRet = ret.Reten_GeneraXML(true, MidpointRounding.AwayFromZero);
04 //comprobamos que se ha creado con la propiedad IsXMLCreated
05 if (rtnRet.IsXMLCreated) {
06     // si se creó, obtenemos de la propiedad XML para guardarlo
07     rtnRet.XML.Save(@"C:\Workspace\SifeiSignature\RetGeneradosTest\ret_creado.xml");
08     if (!File.Exists(@"C:\Workspace\SifeiSignature\RetGeneradosTest\ret_creado.xml")) {
09         MessageBox.Show("No se escribió el archivo");
10     }
11 }
12
    
```

El XML generado sería parecido al siguiente:

```

<?xml version="1.0" encoding="utf-8"?>
<retenciones:Retenciones
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.sat.gob.mx/esquemas/retencio
nPago/1 http://www.sat.gob.mx/esquemas/retencionpago/1/retencionpagov1.xsd"
    Version="1.0"
    FolioInt="DEMO276"
    Sello="@ "
    NumCert="20001000000300022815"
    Cert="MIIFxTCCA62gAwIBA..."
    FechaExp="2019-08-07T14:19:25-05:00"
    CveRetenc="26"
    xmlns:retenciones="http://www.sat.gob.mx/esquemas/retencion
pago/1">
    <retenciones:Emisor RFCEmisor="LAN7008173R5" NomDenRazSocE="FRIOS Y CONGELADOS DE
L SURESTE SA DE CV" />
    <retenciones:Receptor Nacionalidad="Nacional">
        <retenciones:Nacional RFCRecep="MAR980114GQA" NomDenRazSocR="CONSTANTINO GARCIA
PONTON" CURPR="GAPC530722HDFRNN04" />
    </retenciones:Receptor>
    <retenciones:Periodo MesIni="7" MesFin="7" Ejerc="2019" />
    <retenciones:Totales montoTotOperacion="1001.00" montoTotGrav="1001.00" montoTotE
xent="0.00" montoTotRet="80.08">
        <retenciones:ImpRetenidos BaseRet="1001.00" Impuesto="02" montoRet="80.08" Tipo
PagoRet="Pago definitivo" />
    </retenciones:Totales>
</retenciones:Retenciones>
    
```

## 7.10 Sellar el Comprobante de Retención e Información de Pagos

Es posible sellar un Comprobante de Retención e Información de Pagos a través del método `sellRetencion()` 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 de Retención.

```

01 //sellamos el comprobante pasándole el path del certificado y de la llave
02 ReturnXML result= ret.sellaRetencion(result.XML,
03                                     this.rutaCert,
04                                     this.rutaKey,
05                                     this.pwdKey,
06                                     true,
07                                     true);
08 //comprobamos que se halla sellado
09 if (result.EstaSellado) {
10     //escribimos el Comprobante sellado:
11
12     result.XML.Save(@"C:\Workspace\SifeiSignature\RetGeneradosTest\ret_sellado.xml");
13 }
14
15
    
```

Comprobante de retenciones sellado atributo sello (línea 6):

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <retenciones:Retenciones xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3     xsi:schemaLocation="http://www.sat.gob.mx/esquemas/retencionpago/1 http://www.sat.gob.mx/esquemas/retencionpago
4     Version="1.0"
5     FolioInt="DEMO276"
6     Sello="QnZSpDc2wke6i2hwExxcS/PER2HDAruqNqG/HxU98BuMssXxo/120elgkk2uy8WwAl5IJs6AIgRfOXE9bdHj5vvpLCrN11a2i2EZ3g59;
7     NumCert="20001000000300022815"
8     Cert="MIIFxTCCA62gAwIBAgIUMjAwMDEwMDAwMDAwMDAwMjI4MTUwDQYJKoZIhvcNAQELBQAwggFmMSAwHgYDVQDDb8LkMuIDIgZGUGcHJlZj
9     FechaExp="2019-08-07T14:19:32-05:00"
10    CveRetenc="26"
11    xmlns:retenciones="http://www.sat.gob.mx/esquemas/retencionpago/1">
12    <retenciones:Emisor RFCEmisor="LAN7008173R5" NomDenRazSocE="FRIOS Y CONGELADOS DEL SURESTE SA DE CV" />
13    <retenciones:Receptor Nacionalidad="Nacional">
14        <retenciones:Nacional RFCRecep="MAR980114GQA" NomDenRazSocR="CONSTANTINO GARCIA PONTON" CURPR="GAPC530722HDFRNN04" />
15    </retenciones:Receptor>
16    <retenciones:Periodo MesIni="7" MesFin="7" Ejerc="2019" />
17    <retenciones:Totales montoTotOperacion="1001.00" montoTotGrav="1001.00" montoTotExent="0.00" montoTotRet="80.08">
18        <retenciones:ImpRetenidos BaseRet="1001.00" Impuesto="02" montoRet="80.08" TipoPagoRet="Pago definitivo" />
19    </retenciones:Totales>
20
    
```

Imagen 53 Comprobante Sellado

Para utilizar este método antes debió invocar al método `setIdentificacionSifei()`.

## 7.11 Timbrar el Comprobante de Retenciones e Información de Pagos

Es posible timbrar un Comprobante de Retenciones e Información de Pagos a través del método `timbraRetencion()` 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 = ret.timbraRetencion(this.serieSifei,
03                                             this.idEquipo,
04                                             result.XML);
05
06     //verificamos que haya sido timbrado:
07     if (rtnTimbre.IsTimbrado) {
08         //escribimos el cfdi timbrado
09         rtnTimbre.Xml.Save(@"C:\Workspace\SifeiSignature\ret_timbrado.xml");
10
11
    
```

Para utilizar este método antes debió invocar al método `setIdentificacionSifei()`. Es importante mencionar que este método ya no solicita el usuario y contraseña de timbrado, ni el ambiente en el que efectuará el timbrado (Pruebas o Producción), ya que estos son tomados de los valores que se le asignaron al invocar el método `setIdentificacionSifei()`.

Al guardar el Comprobante de Retenciones contendrá el nodo `TimbreFiscalDigital` (líneas 12 a 19):

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <retenciones:Retenciones xmlns:retenciones="http://www.sat.gob.mx/esquemas/retencionpago/1" xmlns:plataformasTecnologicas="http://www.
3 <retenciones:Emisor RFCEmisor="LAN7008173R5" NomDenRazSocE="FRIOS Y CONGELADOS DEL SURESTE SA DE CV" />
4 <retenciones:Receptor Nacionalidad="Nacional">
5 <retenciones:Nacional RFCRecep="MAR980114GQA" NomDenRazSocR="CONSTANTINO GARCIA PONTON" CURPR="GAPC530722HDFRNN04" />
6 </retenciones:Receptor>
7 <retenciones:Periodo MesIni="7" MesFin="7" Ejerc="2019" />
8 <retenciones:Totales montoTotOperacion="1001.00" montoTotGrav="1001.00" montoTotExent="0.00" montoTotRet="80.08">
9 <retenciones:ImpRetenidos BaseRet="1001.00" Impuesto="02" montoRet="80.08" TipoPagoRet="Pago definitivo" />
10 </retenciones:Totales>
11 <retenciones:Complemento>
12 <tfd:TimbreFiscalDigital xmlns:tfd="http://www.sat.gob.mx/TimbreFiscalDigital"
13 xsi:schemaLocation="http://www.sat.gob.mx/TimbreFiscalDigital http://www.sat.gob.mx/sitio internet/cfd/Tim
14 version="1.0"
15 UUID="64C36989-1A2E-4178-8BC0-890136412CEC"
16 selloCFD="QnZSpDc2wke6i2hwExxcS/PER2HDAruqNqG/HxU9BBuMssXxo/120elgkk2uyBhwA15IJs6AIgRfOXE9bdHj5vvpLCrN11a2
17 noCertificadoSAT="30001000000300036825"
18 FechaTimbrado="2019-08-07T14:19:36"
19 selloSAT="VIDg+Nkxukz0zVxTLZw2WETHUgcalyzpzTaxqHjIO+ApQ59iwUvHqcDb4ATQQRUHbxPkhz7Pz8MY7cudzuDZ7PwfenART9i3u
20 </retenciones:Complemento>
21 </retenciones:Retenciones>
    
```

Imagen 54 Comprobante timbrado

## 8. Complementos de Comprobante de Retenciones e Información de Pagos

Con la misma clase `retencion` es posible crear complementos del Comprobante de retención e información de pagos. Actualmente el único complemento disponible es `Servicios Plataformas Tecnológicas`.

### 8.1 Complemento Servicios Plataformas Tecnológicas

#### 8.1.1 Definir Complemento Servicios Plataformas Tecnológicas

```

01 //Complemento servicios plataformas tecnologicas
02 ret.Plat_Define_ServiciosPlataformasTecnologicas(
03     "02", //pPeriodicidad
04     2, //pNumServ
05     1001.00m, //pMonTotServSIVA
06     160.16m, //pTotalIVATrasladado
07     80.08m, //pTotalIVARetenido
08     30.03m, //pTotalISRRetenido
09     80.08m, //pDifIVAEntregadoPrestServ
10     10.71m, //pMonTotalporUsoPlataforma
11     5.26m //pMonTotalContribucionGubernamental
12 );
    
```





### 8.1.2 Agregar DetallesDelServicio

```

01         ret.Plat_Add_DetallesDelServicio(
02             "01",           //pFormaPagoServ
03             "01",           //pTipoDeServ
04             "04",           //pSubTipServ
05             "BBBB010101000", //pRFCTerceroAutorizado
06             new DateTime(2019, 7, 15), //pFechaServ
07             600.555555m,     //pPrecioServSinIVA
08             600.555555m,     //pImpTras_Base
09             "02",           //pImpTras_Impuesto
10             retencion.tipoTasaOCuota.IVA_16, //pImpTras_TasaCuota
11             96.088889m,     //pImpTras_Importe
12             16.00m,         //pComServ_Base
13             0.05m,         //pComServ_Porcentaje
14             5.257914m,     //pComServ_Importe
15             true,           //integraContribucionGubernamental
16             5.257914m,     //pContrGub_ImpContrib
17             "01"           //pContrGub_EntidadDondePaga
18         );
19     
```

### 8.1.3 Generar Complemento Servicios Plataformas Tecnológicas

```

01         ReturnXmlComplemento rtnCompl = ret.Plat_generaComplemento();
02         if (!rtnCompl.IsXmlComplementoCreated)
03         {
04             MessageBox.Show("Error al crear el Xml del Complemento Servicios de Plata
05 formas Tecnológicas 1.0: " + Environment.NewLine + rtnCompl.ErrorDetalle);
06             return;
07         }
08         ret.Retén_AddComplementoComprobante(rtnCompl.XMLComplemento);
09         ret.Retén_AddNamespace(ret.getNsPlataformasTecnologicas10());
    
```



## 9. Referencia

### 9.1 Namespaces

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

- Namespace [Cancelacion](#)
- Namespace [Complementos](#)
- Namespace [Returns](#)
- Namespace [Version3\\_3](#)
- Namespace [Timbre11](#)
- Namespace *ClasesRetenciones*

### 9.2 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.
retencion	Clase que representa al Comprobante de Retenciones e Información de Pagos, así como los métodos para la definición de valores de nodos y atributos. También se encarga de encapsular los métodos correspondientes a los Complementos aplicables a Comprobantes de Retención e Información de Pagos.
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 <code>CFDI33.generaXml()</code> , 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 <code>CFDI33.recuperaTimbre()</code> , por tanto, no contiene métodos.
ReturnXmlComplemento	Clase que representa el resultado de generar un complemento a través del método <code>Pagos10.generaComplemento</code> de clase <code>Complemento</code> , por tanto, esta clase no tiene métodos.

### 9.3 Namespace SifeiSignature3\_3.Version3\_3

Clases:

- `CFDI33`



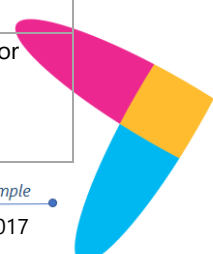
### 9.3.1 Clase CFDI33

#### 9.3.1.1 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, decimal pTotal, string pTipoCompr, string pMetodoPago, string pLugarEped, string pConfirmacion, string rutaCert );</pre>	<p>Inicializa el objeto <b>CFDI33</b> con los atributos principales del comprobante</p>

### 9.3.2 Métodos

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



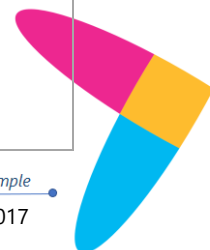
Método de CFDI33	Descripción
);	<p>Cfdi:Conceptos/<b>cfdi:concepto</b>[@indexConcepto]/<b>ComplementoConcepto</b></p> <p><b>Excepciones:</b>  <b>Exception</b>            Si el concepto apuntado por @indexConcepto no existe.</p>
<pre>public int addConcepto( string pCveProdServ, string pNoIdentif, decimal pCantidad, string pCveUnidad, string pUnidad, string pDescripcion, decimal pValorUnit, decimal pImporte, decimal? pDescuento, string pNumCuentaPredial );</pre>	<p>Crea y agrega un nodo hijo <b>concepto</b> al nodo <b>conceptos</b> (Cfdi:Conceptos).</p> <p>xPath:            Cfdi:Conceptos/<b>cfdi:concepto</b></p> <p><b>Returns</b> Índice del nodo <b>concepto</b> que se agregó</p> <p><b>Excepciones:</b>  <b>Exception</b>            Si los parámetros no cumplen el formato y restricciones definidas por el SAT</p> <p><b>Returns</b> Índice del nodo <b>Concepto</b></p>
<pre>public void addImpRetencion(string pImpuesto, decimal pImporte);</pre>	<p>Crea y agrega un nodo <b>Retencion</b> con el impuesto e importe indicado al nodo <b>Impuestos</b> 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:<b>Impuestos</b>/cfdi:<b>Retenciones</b>/cfdi:<b>Retencion</b></p> <p><b>Excepciones:</b>  <b>Exception</b>            Si el nodo <b>Impuestos</b> no existe (es null), no ha sido definida. Lanza excepciones los parámetros no cumplen el formato y restricciones definidas por el SAT.  <b>ArgumentException</b>            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 <b>Retencion</b> al Concepto indicado por el valor @pIndexConc, el nodo <b>retencion</b> tendrá los valores: pBase, pImpuesto, pTipoFactor, pTasaOCuota, pImporte.</p> <p>XPath:            Cfdi:Conceptos/cfdi:concepto[@pIndexConc]/<b>cfdi:Impuestos</b>/cfdi:<b>Retenciones</b>/cfdi:<b>Retencion</b></p> <p><b>Excepciones:</b>  <b>Exception</b>            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.  <b>ArgumentException</b>            Si alguno de los parámetros: @pImpuesto, @pTipoFactor no existe en su catálogo.</p>



Método de CFDI33	Descripción
<pre>public void addImpTraslado( string pImpuesto, string pTipoFactor, decimal pTasaCuota, decimal pImporte );</pre>	<p>Crea y agrega un nodo <b>Traslado</b> con el @pImpuesto, @ptipofactor, @pTasaCuota e @pimporte indicado al nodo <b>Impuestos</b> 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:  <code>cfdi:Impuestos/cfdi:Traslados/cfdi:Traslado</code></p> <p><b>Excepciones:</b>  <b>Exception</b>          Lanza excepciones los parámetros no cumplen el formato y restricciones definidas por el SAT.  <b>ArgumentException</b>          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 <b>Traslado</b> al Concepto indicado por el valor @pIndexConc, el nodo <b>traslado</b> tendrá los valores: @pBase, @pImpuesto, @pTipoFactor, @pTasaOCuota, @pImporte.</p> <p>XPath:  <code>Cfdi:Conceptos/cfdi:concepto[@pIndexConc]/cfdi:Impuestos/cfdi:Traslados/cfdi:Traslado</code></p> <p><b>Excepciones:</b>  <b>Exception</b>          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 <b>InformacionAduanera</b> al Concepto indicado por el valor @pIndexConc, el nodo <b>InformacionAduanera</b> 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:  <code>Cfdi:Conceptos/cfdi:concepto[@pIndexConc]/cfdi:InformacionAduanera</code></p> <p><b>Excepciones:</b>  <b>Exception</b>          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, MidpointRounding</pre>	<p>Genera el comprobante con los datos de la instancia, y retorna un objeto tipo <b>ReturnXML</b>, el cual contiene información acerca del resultado de generar el XML, contiene una referencia al objeto <code>xmlDocument</code>.</p>



Método de CFDI33	Descripción
<pre>pTipoRedondeo = MidpointRounding.AwayFromZero); public List&lt;namespaces&gt; getNamespaces(); public ReturnXML sellaComprobante(XmlDocument xml, string rutaCert, string rutaKey, string pwdCert, bool asignaFechaEmisionActual, bool validaXsds);</pre>	<p>Obtiene la lista de <i>namespaces</i> del comprobante.</p> <p>Recibe una referencia del de tipo <code>XmlDocument</code> (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 <code>CfdiRelacionados</code> 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 <code>CfdiRelacionado</code> con el UUID especificado.</p> <p>XPath:  <b>Cfdi:CfdiRelacionados/cfdi:Relacionado</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Si los parámetros no cumplen con el formato y restricciones establecidas por el SAT.  <b>ArgumentException</b>  Si alguno de los parámetros: <code>@pTipoRelacion</code> no existe en su catálogo.</p>
<pre>public void setEmisor( string pRfc, string pNombre, string pRegFiscal );</pre>	<p>Crea y establece un nodo <code>Emisor</code> como hijo directo al comprobante, verifica que existe el régimen fiscal en el catálogo y lanza una excepción sino existe.</p> <p>XPath:  <b>Cfdi:Emisor</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Si los parámetros no cumplen con el formato y restricciones establecidas por el SAT.  <b>ArgumentException</b>  Si alguno de los parámetros: <code>@pRegFiscal</code> 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 <code>Receptor</code> como hijo directo al comprobante.</p> <p>XPath:  <b>Cfdi:Receptor</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Si los parámetros no cumplen con el formato y restricciones establecidas por el SAT.  <b>ArgumentException</b>  Si alguno de los parámetros: <code>@pUsoCfdi</code>, <code>@pResidenciaFiscal</code> no existe en su catálogo.</p>
<pre>public void setTotalImpuestos( decimal? pTotImpRetenidos, decimal? pTotImpTrasladados );</pre>	<p>Crea y establece un nodo <code>Impuestos</code> (ver XPath abajo) como hijo directo al comprobante, los parámetros <code>@pTotImpRetenidos</code> y <code>@pTotImpTrasladados</code> son opcionales, pero si se pasan deben cumplir con el tipo <code>t_Importe</code>, en caso contrario se lanza una excepción.</p> <p>XPath:  <b>Cfdi:Impuestos</b></p>



Método de CFDI33	Descripción
	<p><b>Excepciones:</b>  <b>Exception</b>            Si los parámetros son establecidos (no <code>null</code>) y no cumplen los formatos y restricciones del SAT.</p>
<pre>public ReturnTimbre timbraComprobante( string serie, string idEquipo, XmlDocument cfdi);</pre>	<p>Genera el timbre del comprobante a partir de XML. Retorna un objeto <code>ReturnTimbre</code> con el Comprobante timbrado y el cbb.</p> <p>@cfdi XmlDocument con el comprobante a timbrar.</p>
<pre>public void setIdentificacionSifei(string pUsuarioSifei, string pPasswordSifei, tipoTimbrado pTipo)</pre>	<p>Establece los valores para autenticarse en los servidores de Sifei. Estos valores se utilizarán para las operaciones de generación de XML, sellado y timbrado.</p> <p>@usuarioSifei asignado para usar <i>SifeiSignature</i>            @passwordSifei asignado para usar <i>SifeiSignature</i>            @Tipo es un valor de enumerado <code>tipoTimbrado</code> para indicar si es de tipo pruebas o productivo.</p>
<pre>public ReturnTimbre depositaFacturaGlobal(string serie, string idEquipo, XmlDocument cfdi, ref string msg)</pre>	<p>Deposita una factura global (cantidad de conceptos mayor o igual a 3000) para su procesamiento para timbrado, retornando un Id de petición, con la que se debe recuperar el timbre.</p> <p>@cfdi XmlDocument con el comprobante a timbrar.</p>
<pre>public static ReturnTimbre recuperaTimbreFacturaGlobal(stri ng usuario, string password, XmlDocument xml, long idPetición, tipoTimbrado tipoTimbrado, ref string msg)</pre>	<p>Recupera el timbre de una factura global depositada previamente, especificando el Id de Petición con el cual se hizo el depósito de la factura global.</p> <p>@idPetición El Id de la Petición por la que se consulta el timbre de la Factura Global.</p>

### 9.3.3 Enumeraciones:

#### 9.3.3.1 Enumeración `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> .

## 9.4 Namespace `SifeiSignature3_3.Timbre11`

### 9.4.1 Clases

- `TimbreBE`

### 9.4.2 Clase `TimbreBE`

Clase que contiene los datos del Timbre Fiscal versión 1.1.

#### 9.4.2.1 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>

## 9.5 Namespace SifeiSignature3\_3.Complementos

### 9.5.1 Clase Complemento

#### 9.5.1.1 Enumeraciones

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

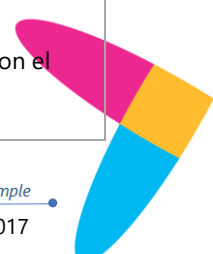
#### 9.5.1.2 Propiedades de clase

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

#### 9.5.1.3 Métodos

Métodos complemento	Descripción
<code>public void Cce11_addDescripcionEspecifica( int pIndexMercancia, string pMarca, string pModelo, string pSubmodelo, string pNumSerie );</code>	Agrega la descripción al complemento de Comercio Exterior.  XPath: Cfdi:Complemento/cce11:ComercioExterior/Mercancias/Mercancia/DescripcionesEspecificas  <b>Excepciones:</b> <b>Exception</b> 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.



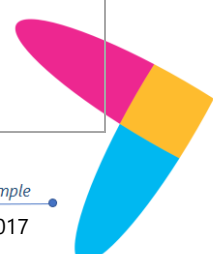


Métodos complemento	Descripción
<pre>public int Cce11_addDestinatario( string pNumRegIdTrib, // número de identificación o registro fiscal del país string pNombre, //nombre del destinatario string pDomCalle, string pDomNumExt, string pDomNumInt, string pDomColonia, string pDomLocalidad, string pDomReferencia, string pDomMunicipio, string pDomEstado, string pDomPais, string pDomCP)</pre>	<p>Crea el nodo <b>Destinatario</b> y su hijo <b>Domicilio</b>. Se hacen las validaciones correspondientes.</p> <p><b>Xpath:</b>  Cfdi:Complemento/<b>cce11:ComercioExterior/Destinatario</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no cumple con el formato válido establecido por el SAT.  <b>ArgumentException</b> Ocurre cuando pDomPais no existe en el catálogo.</p> <p><b>Returns:</b> Índice del nodo <b>Destinatario</b> que se agregó.</p>
<pre>public void Cce11_addDestinatarioDomicilio(in t pIndexDestinatario, 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 hijo <b>Domicio</b> al nodo <b>Destinatario</b>.</p> <p><b>Xpath:</b>  Cfdi:Complemento/<b>cce11:ComercioExterior/Destinatario/Domicio</b></p>
<pre>public int Cce11_addMercancia( string pNoIdentificacion, string pFraccionAranc, decimal? pCantidadAduana, string pUnidadAduana, decimal? pValorUnitAduana, decimal pValorDolares)</pre>	<p>Crea y establece un nodo hijo <b>Mercancia</b> a <b>Mercancias</b>.</p> <p><b>Xpath:</b>  Cfdi:Complemento/<b>cce11:ComercioExterior/Mercancias/Mercancia</b></p> <p><b>Returns:</b> Índice del nodo <b>Mercancia</b> que se agregó.</p>
<pre>public void Cce11_addPropietario( string pNumRegIdTrib, string pResidenciaFiscal )</pre>	<p>Crea y establece un nodo <b>Propietario</b> como hijo a <b>ComercioExterior</b>.</p> <p><b>Xpath:</b>  Cfdi:Complemento/<b>cce11:ComercioExterior/Propietario</b></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 <b>ComercioExterior</b>.</p> <p><b>Xpath:</b>  Cfdi:Complemento/<b>cce11:ComercioExterior</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no cumple con el formato válido establecido por el SAT.  <b>ArgumentException</b> 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 <b>Cce11_Define_ComercioExterior()</b>)</p>

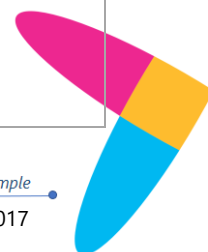


Métodos complemento	Descripción
<pre>public void Cce11_setEmisor(string pCarp, string pDomCalle, string pDomNumExt, string pDomNumInt, string pDomColonia, string pDomLocalidad, string pDomReferencia, string pDomMunicipio, string pDomEstado, string pDomPais, string pDomCP)</pre>	<p>Crea y establece un nodo <b>Emisor</b> (como hijo de <b>ComercioExterior</b>).</p> <p>Xpath:                      Cfdi:Complemento/<b>cce11:ComercioExterior/Emisor</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Si algún atributo no cumple con el formato definido por el SAT, o si no se ha definido el complemento <b>ComercioExterior11</b>.</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 <b>Receptor</b> (como hijo de <b>ComercioExterior</b>).</p> <p>Xpath:                      Cfdi:Complemento/<b>cce11:ComercioExterior/Emisor</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Si algún atributo no cumple con el formato definido por el SAT, o si no se ha definido el complemento <b>ComercioExterior11</b></p>
<pre>public void Det131_AddAllowanceCharge(detalli staAllowanceChargeAllowanceCharge Type allChargeType, detallistaAllowanceChargeSettleme ntType settType, string numConsec, detallistaAllowanceChargeSpecialS ervicesType? specialServType, decimal? porcent)</pre>	<p>Crea y establece un nodo <b>allowanceCharge</b> (como hijo de <b>Detallista</b>).</p> <p>Xpath:                      Cfdi:Complemento/<b>Detallista/ allowanceCharge</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Si algún atributo no cumple con el formato definido por el SAT, o si no se ha definido el complemento <b>Detallista</b> (ver <b>Det131_Define_Detallista()</b>), o si se ha sobrepasado el límite de 99.</p>
<pre>public void Det131_addCurrency( detallistaCurrencyCurrencyISOCode isoCode, /*enumeración */ detallistaCurrencyCurrencyFuncio n funcion1, /*enumeración */ detallistaCurrencyCurrencyFuncio n? funcion2, /*enumeración */ detallistaCurrencyCurrencyFuncio n? funcion3, /*enumeración */ decimal? rateOfChange)</pre>	<p>Crea y agrega un nodo <b>currency</b> como hijo al nodo <b>detallista</b>.</p> <p>Xpath:                      Cfdi:Complemento/<b>Detallista/ currency[]</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha establecido en complemento <b>Detallista</b> o si se superó el número máximo de nodos <b>Currency</b>.</p>
<pre>public void Det131_AddCustomsGln( string gln/* número global de localización (GLN) de la aduana */ )</pre>	<p>Crea y agrega un nodo <b>Customs</b> como hijo al nodo <b>detallista</b>.</p> <p>Xpath:                      Cfdi:Complemento/<b>Detallista/ Customs[]</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha establecido en complemento <b>Detallista</b> o si se superó el número máximo de nodos <b>Customs</b>.</p>
<pre>public void Det131_AddReferenceIdentification ToAdditionalInformation(</pre>	<p>Crea y agrega un nodo <b>referenceIdentification</b> como hijo al nodo <b>AdditionalInformation</b>.</p> <p>Xpath:</p>

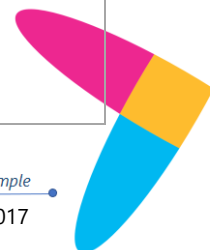
Métodos complemento	Descripción
<pre> detallistaReferenceIdentification Type tipo, /* el número de referencia adicional*/ string valor ) </pre>	<p>Cfdi:Complemento/<b>Detallista</b>/ AdditionalInformation/<b>referenceIdentification</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos (99) <b>referenceIdentification</b>.</p>
<pre> public void Det131_AddReferenceIdentification ToDeliveryNote( string referenceIdentification /* folio de folio de recepción de mercancia*/ ) </pre>	<p>Crea y agrega un nodo <b>referenceIdentification</b> como hijo al nodo <b>DeliveryNote</b>.</p> <p><b>Xpath:</b>  Cfdi:Complemento/<b>Detallista</b>/ DeliveryNote/<b>referenceIdentification</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos (30) <b>referenceIdentification</b>. Por último, si no se ha establecido <b>DeliveryNote</b> (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 <b>referenceIdentification</b> como hijo al nodo <b>orderIdentification</b>.</p> <p><b>Xpath:</b>  Cfdi:Complemento/<b>Detallista</b>/ orderIdentification /<b>referenceIdentification</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos (30) <b>referenceIdentification</b>. Por último, si no se ha establecido <b>orderIdentification</b> (que es requerido).</p>
<pre> public void Det131_AddTotalAllowanceCharge( detallistaTotalAllowanceChargeAll owanceOrChargeType allOrChrgType, /*enum para indicar si es cargo o descuento*/ detallistaTotalAllowanceChargeSpe cialServicesType? specServType, /* enum para indicar el tipo de servicio*/ decimal? Amnt /* total de los cargos o descuentos*/ ) </pre>	<p>Crea y agrega un nodo <b>TotalAllowanceCharge</b> como hermano al nodo <b>TotalAllowanceCharge</b>.  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><b>Xpath:</b>  Cfdi:Complemento/<b>Detallista</b>/ <b>TotalAllowanceCharge[]</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos (30) <b>referenceIdentification</b>. Por último, si no se ha establecido <b>orderIdentification</b> (que es requerido).</p>
<pre> public int Det131_AgregaSpecialInstruction( detallistaSpecialInstructionCode Code,/* tipo de instrucciones comerciales*/ string text ) </pre>	<p>Crea y agrega un nodo <b>specialInstruction</b> como hermano al nodo <b>specialInstruction</b>.</p> <p><b>Xpath:</b>  Cfdi:Complemento/<b>Detallista</b>/ <b>specialInstruction[]</b></p> <p><b>Returns:</b> Índice del nodo <b>specialInstruction</b> que se agregó</p> <p><b>Excepciones:</b></p>



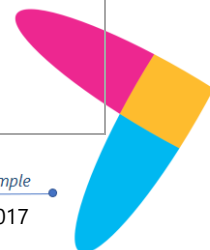
Métodos complemento	Descripción
	<b>Exception</b> Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos (4) <code>specialInstruction</code> .
<pre>public void Det131_AgregaTextASpecialInstruccion( int indexSpecialInstr, /*index del nodo specialInstruction */ string text )</pre>	<p>Crea y agrega un nodo <code>text</code> a <code>specialInstruction</code> (como hijo) dado un índice (el cual indica a qué nodo <code>specialInstruction</code> será el padre).</p> <p><b>Xpath:</b>  <code>Cfdi:Complemento/Detallista/specialInstruction[indexSpecialInstr]/text</code></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha establecido en complemento Detallista, si no existe el nodo <code>specialInstruction</code> o si se superó el número máximo de nodos (15) <code>text</code>.</p>
<pre>public void Det131_Agrega_RequestForPaymentId entification_Detallista( detallistaRequestForPaymentIdenti ficationEntityType EntityType /* tipo de transacción*/ )</pre>	<p>Crea y agrega (o reemplaza si existe) un <code>requestForPaymentIdentification</code> a <code>detallista</code> (como hijo) para indicar la transacción.</p> <p><b>Xpath:</b>  <code>Cfdi:Complemento/Detallista/requestForPaymentIdentification/ entityType</code></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha establecido en complemento Detallista.</p>
<pre>public void Det131_Define_Detallista(detallis taDocumentStatus DocumentStatus)</pre>	<p>Define el complemento <code>detallista</code> del comprobante.</p> <p><b>Xpath:</b>  <code>Cfdi:Complemento/Detallista</code></p>
<pre>public ReturnXmlComplemento Det131_generaComplemento()</pre>	<p>Retorna el complemento <code>detallista</code> en el objeto <code>ReturnXmlComplemento</code> si todo fue bien.</p> <p><b>Excepciones:</b>  <b>Exception</b> 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 <code>buyer</code> a <code>detallista</code> (como hijo) para indicar la información del comprador.</p> <p><b>Xpath:</b>  <code>Cfdi:Complemento/Detallista/ buyer/gln, buyer/contactInformation/ personOrDepartmentName/text</code></p> <p><b>Excepciones:</b>  <b>Exception</b> 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 <code>DeliveryNote</code> a <code>detallista</code> (como hijo) para indicar la información recepción de mercancía.</p> <p><b>Xpath:</b>  <code>Cfdi:Complemento/Detallista/ DeliveryNote</code></p> <p><b>Excepciones:</b></p>



Métodos complemento	Descripción
	<b>Exception</b> Esta excepción ocurre si no se ha establecido en complemento Detallista.
<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>Creación y establece el <b>InvoiceCreator</b> a <b>detallista</b> (como hijo).</p> <p><b>Xpath:</b>  Cfdi:Complemento/<b>Detallista</b>/ <b>InvoiceCreator</b>/nameAndAddress,  <b>InvoiceCreator</b>/ alternatePartyIdentification</p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha establecido el complemento Detallista.</p>
<pre>public void Det131_SetOrderIdentification( DateTime? referenceDate, string ReferenceIdentification )</pre>	<p>Creación y establece el <b>orderIdentification</b> a <b>detallista</b> (como hijo).</p> <p><b>Xpath:</b>  Cfdi:Complemento/<b>Detallista</b>/ <b>orderIdentification</b>/  <b>referenceIdentification</b></p> <p><b>Excepciones:</b>  <b>Exception</b> 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? 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>Creación y establece el <b>paymentTerms</b> a <b>detallista</b> (como hijo) para especificar los términos de pago de la factura</p> <p><b>Xpath:</b>  Cfdi:Complemento/<b>Detallista</b>/ <b>paymentTerms</b>/ y/o  Cfdi:Complemento/<b>Detallista</b>/ <b>paymentTerms</b>/paymentTimePeriod  y/o  Cfdi:Complemento/<b>Detallista</b>/ <b>paymentTerms</b>/discountPayment</p> <p><b>Excepciones:</b>  <b>Exception</b> 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 <b>seller</b> a <b>detallista</b> (como hijo) para especificar la información del vendedor.</p> <p><b>Xpath:</b>  Cfdi:Complemento/<b>Detallista</b>/ <b>seller</b></p> <p><b>Excepciones:</b></p>



Métodos complemento	Descripción
	<b>Exception</b> Esta excepción ocurre si no se ha establecido el complemento Detallista.
<pre>public void Det131_SetShipTo( string gln, string name, string streetAddressOne, string city, string cp )</pre>	<p>Crea y establece el <b>shipTo</b> a <b>detallista</b> (como hijo) para especificar la ubicación donde debe realizarse la entrega de la mercancía.</p> <p><b>Xpath:</b> Cfdi:Complemento/<b>Detallista</b>/<b>shipTo</b></p> <p><b>Excepciones:</b> <b>Exception</b> Esta excepción ocurre si no se ha establecido el complemento Detallista.</p>
<pre>public void Det131_setTotalAmount( decimal amnt )</pre>	<p>Crea y establece el <b>totalAmount</b> a <b>detallista</b> (como hijo).</p> <p><b>Xpath:</b> Cfdi:Complemento/<b>Detallista</b>/<b>totalAmount</b></p> <p><b>Excepciones:</b> <b>Exception</b> 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 <b>ReturnXmlComplemento</b>.</p> <p><b>Excepciones:</b> <b>Exception</b> Esta excepción ocurre si los parámetros no cumplen la especificación del SAT en formato o restricción: @curp, @rfcPago, @autRVOE.</p>
<pre>public void ImpLocal10_Agrega_RetencionImpues toLocal( string nombre, /*Nombre del impuesto local retenido*/ decimal importe, decimal tasa)</pre>	<p>Crea y establece el <b>RetencionesLocales</b> a <b>ImpuestosLocales</b> (como hijo).</p> <p><b>Xpath:</b> Cfdi:Complemento/<b>ImpuestosLocales</b>/<b>RetencionesLocales</b></p> <p><b>Excepciones:</b> <b>Exception</b> 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 <b>RetencionesLocales</b> a <b>ImpuestosLocales</b> (como hijo).</p> <p><b>Xpath:</b> Cfdi:Complemento/<b>ImpuestosLocales</b>/<b>RetencionesLocales</b></p> <p><b>Excepciones:</b> <b>Exception</b> 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 <b>ImpuestosLocales</b>.</p> <p><b>Xpath:</b> Cfdi:Complemento/<b>ImpuestosLocales</b></p>

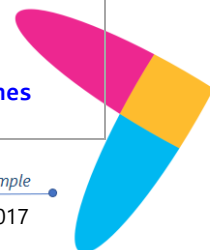


Métodos complemento	Descripción
	<p><b>Excepciones:</b>  <b>Exception</b> Ninguna.</p>
<pre>public ReturnXmlComplemento Implocal10_generaComplemento()</pre>	<p>Genera el complemento impuestos locales que es devuelto en el objeto <b>ReturnXmlComplemento</b>, si existe un error no fatal es incluido en el mismo objeto. Ver propiedades:</p> <ul style="list-style-type: none"> <li>• <b>ReturnXmlComplemento.IsXmlComplementoCreated</b></li> <li>• <b>ReturnXmlComplemento.ErrorDetalle</b></li> </ul> <p><b>Excepciones:</b>  <b>Exception</b>                  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 <b>Deduccion</b> a <b>Deducciones</b> (como hijo).</p> <p><b>Xpath:</b>                  Cfdi:Complemento/nomina12: Nomina/nomina12:  <b>Deducciones/Deduccion[]</b></p> <p><b>Excepciones:</b>  <b>Exception</b> 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><b>ArgumentException</b> 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, decimal pImportePagado )</pre>	<p>Crea y establece un nodo <b>HorasExtra</b> (como hijo) a la <b>Percepcion</b> apuntada por @indexPercepcion.</p> <p><b>Xpath:</b>                  Cfdi:Complemento/nomina12: Nomina/nomina12:Percepciones/  <b>Percepcion[@indexPercepcion]/HorasExtra[]</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha establecido el complemento Nómina 1.2.                  Si no se ha establecido el nodo <b>Percepcion</b>.                  Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p> <p><b>ArgumentException</b> 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 <b>Incapacidad</b> como hijo a <b>Incapacidades</b>.</p> <p><b>Xpath:</b>                  Cfdi:Complemento/nomina12: Nomina/nomina12:<b>Incapacidades/                  Incapacidad[]</b></p> <p><b>Excepciones:</b></p>

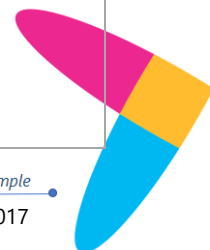
Métodos complemento	Descripción
	<p><b>Exception</b> Esta excepción ocurre si no se ha establecido el complemento Nómina 1.2. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p> <p><b>ArgumentException</b> si @pTipo no existe en el catálogo definido 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 <b>OtroPago</b> como hijo a <b>OtrosPagos</b> para expresar otros pagos aplicables.</p> <p><b>Xpath:</b> Cfdi:Complemento/nomina12: Nomina/nomina12:OtrosPagos/<b>OtroPago[]</b></p> <p><b>Returns:</b> Índice del nodo <b>OtroPago</b> que se agregó</p> <p><b>Excepciones:</b> <b>Exception</b> 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><b>ArgumentException</b> 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 <b>Percepcion</b> como hijo a <b>Percepciones</b>.</p> <p><b>Xpath:</b> Cfdi:Complemento/nomina12: Nomina/nomina12:Percepciones/<b>Percepcion[]</b></p> <p><b>Returns:</b> Índice del nodo <b>Percepcion</b> que se agregó</p> <p><b>Excepciones:</b> <b>Exception</b> 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><b>ArgumentException</b> 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 <b>SubContratacion</b> como hijo a <b>Receptor</b>.</p> <p><b>Xpath:</b> Cfdi:Complemento/nomina12: Nomina/nomina12:Receptor/<b>SubContratacion[]</b></p> <p><b>Excepciones:</b> <b>Exception</b> 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,</pre>	<p>Define (establece) el complemento Nómina versión 1.2.</p>



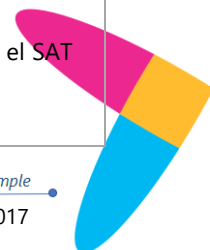
Métodos complemento	Descripción
<pre>DateTime pFechaPago, DateTime fechaInicPago, DateTime fechaFinPago, decimal numDiasPag, decimal? totPercepciones, decimal? totDeducciones, decimal? totOtrosPagos )</pre>	<p>Xpath:  Cfdi:Complemento/<b>nomina12: Nomina/</b></p> <p><b>Excepciones:</b>  <b>Exception</b>  Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.  <b>ArgumentException</b>  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 <b>ReturnXmlComplemento</b>, si existe un error no fatal es incluido en el mismo objeto. Ver propiedades:</p> <ul style="list-style-type: none"> <li>• <b>ReturnXmlComplemento.IsXmlComplementoCreated</b></li> <li>• <b>ReturnXmlComplemento.ErrorDetalle</b></li> </ul> <p><b>Excepciones:</b>  <b>Exception</b>  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 <b>AccionesOTitulos</b> como hijo a la <b>Percepcion</b> apuntada por @indexPercepcion.</p> <p>Xpath:  Cfdi:Complemento/<b>nomina12: Nomina/nomina12:Percepcion[@indexPercepcion]/AccionesOTitulos</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.  Si la <b>Percepcion</b> padre apuntada no existe.  Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void Nomina12_setCompensacionSalDOSAFavor( int indexOtroPago, decimal pSaldoAFavor, short pAño, decimal pRemanenteAFavor)</pre>	<p>Crea y define la <b>CompensacionSalDOSAFavor</b> al <b>OtroPago</b> apuntado por @indexOtroPago.</p> <p>Xpath:  Cfdi:Complemento/<b>nomina12: Nomina/nomina12:/OtrosPagos/OtroPago [@indexOtroPago]/CompensacionSalDOSAFavor</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.  Si <b>OtrosPagos</b> 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 <b>Deducciones</b> como hijo a <b>nomina</b>.</p> <p>Xpath:  Cfdi:Complemento/<b>nomina12: Nomina/nomina12:/Deducciones</b></p>



Métodos complemento	Descripción
	<p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.</p>
<pre>public void Nomina12_setEmisor( string pCarp, string pRegPatronal, string pRfcPatronOrigen )</pre>	<p>Crea y define el nodo <b>Emisor</b> como hijo a <b>nomima</b>.</p> <p>Xpath:  Cfdi:Complemento/<b>nomina12: Nomina</b>/nomina12:/<b>Emisor</b></p> <p><b>Excepciones:</b>  <b>Exception</b> 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 <b>EntidadSNCF</b> como hijo a <b>Emisor</b>.</p> <p>Xpath:  Cfdi:Complemento/<b>nomina12: Nomina</b>/nomina12:/<b>Emisor/EntidadSNCF</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.  Si no se ha definido la instancia Emisor.  <b>ArgumentException</b>  Si @pOrigenRecurso no existe en el catálogo definido por el SAT.</p>
<pre>public void Nomina12_setJubilacionPensionRetiro( decimal? pTotalUnaExhibicion, decimal? pTotalParcialidad, decimal? pMontoDiario, decimal pIngresoAcumulable, decimal pIngresoNoAcumulable )</pre>	<p>Crea y define el nodo <b>JubilacionPensionRetiro</b> como hijo a <b>Percepciones</b>.</p> <p>Xpath:  Cfdi:Complemento/<b>nomina12: Nomina</b>/nomina12:/<b>Percepciones/JubilacionPensionRetiro</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.  Si no se ha definido la instancia nodo <b>Percepciones</b>.  <b>ArgumentException</b>  Ninguna.</p>
<pre>public void Nomina12_setPercepciones( decimal? pTotalSueldos, decimal? pTotalSeparacionIndemn, decimal? pTotalJubilacionPensionRetiro, decimal pTotalGravado, decimal pTotalExento )</pre>	<p>Crea y define el nodo <b>Percepciones</b> como hijo a <b>nomina</b>.</p> <p>Xpath:  Cfdi:Complemento/<b>nomina12: Nomina</b>/nomina12:/<b>Percepciones</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.  <b>ArgumentException</b>  Ninguna.</p>
<pre>public void Nomina12_setReceptor( string pCarp, string pNumSegSocial, DateTime? pFechaInicRelLab, string pAntiguedad,</pre>	<p>Crea y define el nodo Emisor como hijo a <b>Receptor</b>.</p> <p>Xpath:  Cfdi:Complemento/<b>nomina12: Nomina</b>/nomina12:/<b>Receptor</b></p>



Métodos complemento	Descripción
<pre>string pTipoContrato, Nomina12_pSiNoNoEspecificado 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><b>Excepciones:</b>  <b>Exception</b> 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> <p><b>ArgumentException</b>                  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 <b>SeparacionIndemnizacion</b> como hijo a <b>Percepciones</b>.</p> <p>Xpath:                  Cfdi:Complemento/nomina12: <b>Nomina</b>/nomina12:/<b>Percepciones/</b>  <b>SeparacionIndemnizacion</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.                  Si no se ha definido la instancia nodo <b>Percepciones</b>.                  Si @pNumAñosServicio es mayor a 99.</p> <p><b>ArgumentException</b>                  Ninguna.</p>
<pre>public void Nomina12_setSubsidioAlEmpleo(int indexOtroPago, decimal pSubsidioCausado)</pre>	<p>Crea y define el nodo <b>SubsidioAlEmpleo</b> como hijo a <b>OtroPago</b>.</p> <p>Xpath:                  Cfdi:Complemento/nomina12:  <b>Nomina</b>/nomina12:/<b>OtrosPagos/OtroPago/SubsidioAlEmpleo</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.                  Si no se ha definido la instancia nodo <b>OtrosPagos</b>.                  Si @pNumAñosServicio es mayor a 99.</p> <p><b>ArgumentException</b>                  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 <b>DoctoRelacionado</b> como hijo de <b>Pago</b>.</p> <p>Xpath:                  Cfdi:Complemento/pagos/pago/ <b>DoctoRelacionado[]</b></p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha definido el complemento Pago 1.0.                  Si no se ha definido la instancia nodo <b>pago</b>.                  Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>



Métodos complemento	Descripción
)	<b>ArgumentException</b> Si @pMonedaDRes ó @pMetodoDePagoDR no existen en el catálogo del SAT.
<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 <b>Pago</b> como hijo de <b>Pagos</b>.</p> <p>Xpath: Cfdi:Complemento/<b>pagos/pago[]</b></p> <p><b>Returns:</b> Índice del nodo <b>pago</b> que se agregó</p> <p><b>Excepciones:</b>  <b>Exception</b> Esta excepción ocurre si no se ha definido el complemento Pago 1.0.                      Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p> <p><b>ArgumentException</b>                      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 <b>ReturnXmlComplemento</b>, si existe un error no fatal es incluido en el mismo objeto. Ver propiedades:</p> <ul style="list-style-type: none"> <li>• <b>ReturnXmlComplemento.IsXmlComplementoCreated</b></li> <li>• <b>ReturnXmlComplemento.ErrorDetalle</b></li> </ul> <p><b>Excepciones:</b>  <b>Exception</b>                      Si los parámetros no cumplen con las especificaciones 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>

## 9.6 Namespace SifeiSignature3\_3.Returns

### 9.6.1 Clase ReturnXML

#### 9.6.1.1 Constructores

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

#### 9.6.1.2 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 " <a href="#">Matriz de Errores</a> " 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.

## 9.6.2 Clase ReturnTimbre

### 9.6.2.1 Constructores

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

### 9.6.2.2 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 <i>XmlElement</i> 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.
<code>public bool EsFacturaGlobalDepositada { get; set; }</code>	Si se procesó factura global indica si se depositó en el servicio de timbrado.
<code>public long IdPeticionFacturaGlobal { get; set; }</code>	Id de Petición de timbrado de factura global, debe utilizarse este valor para recuperar el timbre de la factura global.
<code>public int FoliosConsumidosFacturaGlobal { get; set; }</code>	Indica la cantidad de folios que fueron utilizados para timbrar la factura global.

## 9.6.3 Clase ReturnXmlComplemento

### 9.6.3.1 Constructores

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

### 9.6.3.2 Propiedades

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



## 9.6.4 Clase ReturnCancelacion2018

### 9.6.4.1 Constructores

Constructor	Descripción
<code>public ReturnCancelacion2018();</code>	Constructor vacío.

### 9.6.4.2 Propiedades

Constructor	Descripción
<code>public bool Procesado{ get; set; }</code>	Indica si fue recibida la petición al servicio de cancelacion.
<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>	<i>String</i> con el detalle del error.
<code>public List&lt;string&gt; UuidsNoProcesados { get; set; }</code>	Lista de los UUIDs que no fueron procesados para su cancelación ya que los únicos que se procesan son los que tienen Estado Vigente y su estatus de Ccancelación es diferente a "En proceso".  Siempre se debe revisar este campo independientemente del resultado de la cancelación.

## 9.6.5 Clase ReturnDocumentosRelacionados

### 9.6.5.1 Constructores

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

### 9.6.5.2 Propiedades

Constructor	Descripción
<code>public bool Procesado{ get; set; }</code>	Indica si fue recibida la petición al servicio consulta CFDI relacionados.
<code>public string AcuseXml{ get; set; }</code>	Acuse de metodo
<code>public string CodigoError { get; set; }</code>	Indica el código de error (si es que hubo).
<code>public string ErrorDetalle { get; set; }</code>	<i>String</i> con el detalle del error.
<code>public ConsultaRelacionados Result { get; set; }</code>	Objeto <code>ConsultaRelacionados</code> que contiene los datos de los CFDI padre e hijo.

## 9.6.6 Clase ReturnQrAcuse

### 9.6.6.1 Constructores

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

### 9.6.6.2 Propiedades

Constructor	Descripción
<code>public bool Procesado{ get; set; }</code>	Indica si fue recibida la petición al servicio consulta CFDI relacionados.
<code>public string AcuseXml{ get; set; }</code>	Acuse de metodo
<code>public string CodigoError { get; set; }</code>	Indica el código de error (si es que hubo).
<code>public string ErrorDetalle { get; set; }</code>	<i>String</i> con el detalle del error.
<code>public string CodigoEstatus { get; set; }</code>	Valor relacionado a la solicitud de consulta(no al CFDI)
<code>public string EsCancelable { get; set; }</code>	Indica si el CFDI es cancelable: No cancelable Cancelable sin aceptación Cancelable con aceptación
<code>public string Estado { get; set; }</code>	Estado del CFDI:Vigente,Cancelado, No Encontrado
<code>public string EstatusCancelacion { get; set; }</code>	Estatus de la cancelación: Cancelado sin aceptación En proceso Plazo vencido Cancelado con aceptación Solicitud rechazada

### 9.6.7 Clase ReturnPeticonesPendientes

#### 9.6.7.1 Constructores

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

#### 9.6.7.2 Propiedades

Constructor	Descripción
<code>public bool Procesado{ get; set; }</code>	Indica si fue recibida la petición al servicio consulta CFDI relacionados.
<code>public string AcuseXml{ get; set; }</code>	Acuse de metodo
<code>public string CodigoError { get; set; }</code>	Indica el código de error (si es que hubo).
<code>public string ErrorDetalle { get; set; }</code>	<i>String</i> con el detalle del error.
<code>public string CodEstatus { get; set; }</code>	<i>Codigo del WS</i>
<code>public string CodEstatusDescripcion { get; set; }</code>	Descripción del código del WS
<code>public string[] UUID { get; set; }</code>	Lista de UUID de peticiones pendientes.

### 9.6.8 Clase ReturnAceptacionRechazoResult

#### 9.6.8.1 Constructores

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



### 9.6.8.2 Propiedades

Constructor	Descripción
<code>public bool Procesado { get; set; }</code>	Indica si fue recibida la petición al servicio consulta CFDI relacionados.
<code>public string AcuseXml { get; set; }</code>	Acuse de metodo
<code>public string CodigoError { get; set; }</code>	Indica el código de error (si es que hubo).
<code>public string ErrorDetalle { get; set; }</code>	<i>String</i> con el detalle del error.
<code>public string CodEstatus { get; set; }</code>	<i>Codigfo de estatus donde, 1000 = Se recibió la respuesta de la petición de forma exitosa</i>
<code>public string CodEstatusDescripcion { get; set; }</code>	Descripcion del código de estatus
<code>public string RfcReceptor { get; set; }</code>	RFC receptor que envio a cancelar los CFDI.
<code>public string RfcPac { get; set; }</code>	RFC del PAC por el cual son cancelados los CFDI.
<code>public DateTime Fecha { get; set; }</code>	Fecha
<code>public string Sello { get; set; }</code>	Sello
<code>public List&lt;DetalleFolios&gt; Folios { get; set; }</code>	Colección de objetos DatelleFolio

### 9.6.9 Clase DetalleFolios

#### 9.6.9.1 Constructores

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

#### 9.6.9.2 Propiedades

Constructor	Descripción
<code>public string UUID { get; set; }</code>	UUID la petición
<code>public string EstatusUUID { get; set; }</code>	Estatus especifico para la petición Cancelada
<code>public string EstatusUUIDDscripcion { get; }</code>	Descripcion del estatus UUID.
<code>public TipoRespuesta Respuesta { get; set; }</code>	Respuesta correspondiente al folio que se envio.

#### 9.6.9.3 Enumeracion TipoRespuesta

TipoRespuesta
Aceptacion = 0,
Rechazo = 1





## 9.7 Namespace SifeiSignature3\_3.Complementos.Detallista131

### 9.7.1 Enumeraciones de complemento Detallista versión 1.3.1

Enumeraciones de Detallista
detallistaAllowanceChargeAllowanceChargeType
detallistaAllowanceChargeMonetaryAmountOrPercentageRateBase
detallistaAllowanceChargeSettlementType
detallistaAllowanceChargeSpecialServicesType
detallistaCurrencyCurrencyFunction
detallistaCurrencyCurrencyISOCODE
detallistaDocumentStatus
detallistaInvoiceCreatorAlternatePartyIdentificationType
detallistaLineItemAdditionalInformationReferenceIdentificationType
detallistaLineItemAdditionalQuantityQuantityType
detallistaLineItemAllowanceChargeAllowanceChargeType
detallistaLineItemAllowanceChargeSettlementType
detallistaLineItemAllowanceChargeSpecialServicesType
detallistaLineItemAlternateTradeItemIdentificationType
detallistaLineItemCustomsAlternatePartyIdentificationType
detallistaLineItemLogisticUnitsSerialShippingContainerCodeType
detallistaLineItemPalletInformationDescriptionType
detallistaLineItemPalletInformationTransportMethodOfPayment
detallistaLineItemTradeItemDescriptionInformationLanguage
detallistaLineItemTradeItemTaxInformationTaxCategory
detallistaLineItemTradeItemTaxInformationTaxTypeDescription
detallistaOrderIdentificationReferenceIdentificationType
detallistaPaymentTermsDiscountPaymentDiscountType
detallistaPaymentTermsNetPaymentNetPaymentTermsType
detallistaPaymentTermsNetPaymentPaymentTimePeriodTimePeriodDueTimePeriod
detallistaPaymentTermsPaymentTermsEvent
detallistaPaymentTermsPaymentTermsRelationTime
detallistaReferenceIdentificationType
detallistaRequestForPaymentIdentificationEntityType
detallistaSellerAlternatePartyIdentificationType
detallistaSpecialInstructionCode
detallistaTotalAllowanceChargeAllowanceOrChargeType
detallistaTotalAllowanceChargeSpecialServicesType

## 9.8 Namespace SifeiSignature3\_3.Complementos.Iedu10

### 9.8.1 Enumeraciones de complemento Instituciones Educativas versión 1.0

Namespace: SifeiSignature3_3.Complementos.Iedu10
instEducativasNivelEducativo

### 9.8.2 Enumeraciones de complemento Nómina versión 1.2

Namespace: SifeiSignature3_3.Complementos.Nomina12
NominaReceptorSindicalizado



## 9.9 Referencia del *Namespace* SifeiSignature3\_3.Cancelacion

### 9.9.1 Clase Acuse

#### 9.9.1.1 Constructores

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

#### 9.9.1.2 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&lt;AcuseDetalle&gt; AcuseDetalle { get; set; }</code>	Lista de detalles de acuse (indicando en UUID y estatus) ver clase <a href="#">AcuseDetalle</a> .

### 9.9.2 Clase AcuseDetalle

#### 9.9.2.1 Constructores

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

#### 9.9.2.2 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 0 Códigos de Respuesta de Cancelación).
<code>public string Mensaje { get; set; }</code>	Mensaje relacionado al estatus.

### 9.9.3 Clase Cancelacion (deprecada, obsoleta)

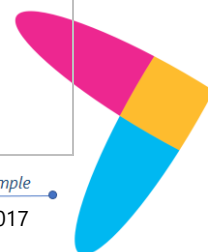
Esta clase no debe usarse, favor de referirse a [Cancelacion2018](#).

#### 9.9.3.1 Constructores

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

#### 9.9.3.2 Métodos

Metodos de Cancelacion	Descripción
<code>public ReturnCancelacion cancela(                      ref string msg,                      string[] listaUUIDS, /*arreglo de UUID a                      cancelar*/                      string rfcSifei,                      string password,</code>	Cancela todos los CFDI pasados en el parámetro @listaUUIDS  <b>Excepciones:</b> <b>Exception</b>



Metodos de Cancelacion	Descripción
<pre>string rfcEmisor, string rutaCert, /*ruta del certificado*/ string rutaKey, /*ruta del key*/ string pwdCert, /*Contraseña del certificado*/ Definiciones.TipoTimbrado pTipoTimbrado );</pre>	<p>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><b>ArgumentException</b>          Ninguna.</p> <p><b>Returns:</b>          Objeto de tipo <a href="#">ReturnCancelacion</a> que indica el resultado de la operación cancelar (ver clase <a href="#">ReturnCancelacion</a>).</p>

## 9.9.4 Clase Cancelacion 2018

### 9.9.4.1 Constructores

Constructor	Descripción (deprecada,obsoleta)
<pre>public Cancelacion2018 ();</pre>	Constructor vacío.

### 9.9.4.2 Métodos

Metodos de Cancelacion2018	Descripción
<pre>public ReturnCancelacion2018 cancelaCfdi( ref string msg, string[] listaUUIDS, string rfcSifei, string password, string rfcEmisor, string rutaCert, string rutaKey, string pwdCert, Definiciones.TipoTimbrado pTipoTimbrado);</pre>	<p>Cancela todos los CFDI pasados en el parámetro @listaUUIDS, siguiendo el nuevo esquema de cancelación, es decir, se sigue el nuevo mecanismo implementado para la cancelación.</p> <p><b>Excepciones:</b>  <b>Exception</b>          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><b>ArgumentException</b>          Ninguna.</p> <p><b>Returns:</b>          Objeto de tipo <a href="#">ReturnCancelacion</a> que indica el resultado de la operación cancelar (ver clase <a href="#">ReturnCancelacion</a>).</p>
<pre>public ReturnDocumentosRelacionados consultaDocumentosRelacionados( string rfcUsuarioSifei, string password, tipoConsultaRelacionados consultar, string rfcEmisor, string rfcReceptor, string rutaCert, string rutaKey, string pwdCert, string uuid, ref string msg, Definiciones.TipoTimbrado pTipoTimbrado);</pre>	<p>Retorna todos los documentos relacionados al CFDI. El parámetro @consultar especifica qué sujeto está realizando la consulta (Emisor o Receptor) y por lo tanto los certificados deben corresponder al rfc del sujeto que esta realizando la consulta. El parámetro @rfcEmisor se vuelve requerido si el parámetro @consultar corresponde a Emisor, por el contrario el parámetro @rfcReceptor se vuelve requerido si el parámetro @consultar corresponde a Receptor</p> <p><b>Returns:</b>          Objeto de tipo <a href="#">ReturnDocumentosRelacionados</a> que indica el resultado de la operación, contiene tanto los UUID, y RFC de los CFDI padre e hijos.</p>
<pre>public ReturnQrAcuse consultaEstatusCfdi(</pre>	<p>Consulta el estado de un CFDI.</p>

Metodos de Cancelacion2018	Descripción
<pre>string uuid, string rfcEmisor, string rfcReceptor, decimal total, string selloCfd8, Definiciones.TipoTimbrado tipoServicio );</pre>	<p>@selloCfd8 Ultimos 8 caracteres del sello</p> <p><b>Returns:</b>          Objeto de tipo <code>ReturnDocumentosRelacionados</code> que indica el resultado de la operación, contiene tanto los UUID, y RFC de los CFDI padre e hijos.</p>
<pre>public ReturnPeticionesPendientes getPeticionesCancelacionPendientes( string rfcUsuarioSifei, string password, string rfcReceptor, Definiciones.TipoTimbrado pTipoTimbrado );</pre>	<p>Consulta las peticiones de cancelación pendiente que requieren autorización para ser canceladas por el @ rfcReceptor</p> <p><b>Returns:</b>          Objeto de tipo <code>ReturnPeticionesPendientes</code>,</p>
<pre>public ReturnAceptacionRechazoResult procesaRespuestaPeticionesCancelacion( string rfcUsuarioSifei, string password, string rfcReceptor, string rutaCert, string rutaKey, string pwdCert, ref string msg, folios[] respFolios, //array de objetos folio donde se indica si se acepta o no la peticion Definiciones.TipoTimbrado pTipoTimbrado);</pre>	<p>Metodo que realiza la operación de aceptación y rechazo de peticiones de cancelación. Para hacerlo se deben pasar el parámetro @folios, que es el arreglo de objetos tipo <code>folios</code>, en el se indica el UUID a la vez que la respuesta (Aceptacion o Rechazo)</p> <p><b>Returns:</b>  <code>ReturnAceptacionRechazoResult</code></p>

## 9.10 Referencia del *Namespace* SifeiSignature3\_3.Complementos.Cce11

### 9.10.1 Clases

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

### 9.10.2 Clase ComercioExterior

#### 9.10.2.1 Constructores

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



### 9.10.2.2 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>

### 9.10.3 Clase ComercioExteriorDestinatario

#### 9.10.3.1 Constructores

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

#### 9.10.3.2 Propiedades

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

### 9.10.4 Clase ComercioExteriorDestinatarioDomicilio

#### 9.10.4.1 Constructores

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

#### 9.10.4.2 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 string CodigoPostal { get; set; }</code>

### 9.10.5 Clase ComercioExteriorEmisor

#### 9.10.5.1 Constructores

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

#### 9.10.5.2 Propiedades

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

### 9.10.6 Clase ComercioExteriorEmisor

#### 9.10.6.1 Constructores

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

#### 9.10.6.2 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>
<code>public string Estado { get; set; }</code>
<code>public string Pais { get; set; }</code>
<code>public string CodigoPostal { get; set; }</code>

### 9.10.7 Clase ComercioExteriorMercancia

#### 9.10.7.1 Constructores

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



### 9.10.7.2 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>

### 9.10.8 Clase ComercioExteriorMercanciaDescripcionesEspecificas

#### 9.10.8.1 Constructores

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

#### 9.10.8.2 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>

### 9.10.9 Clase ComercioExteriorPropietario

#### 9.10.9.1 Constructores

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

#### 9.10.9.2 Propiedades

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

### 9.10.10 Clase ComercioExteriorReceptor

#### 9.10.10.1 Constructores

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

#### 9.10.10.2 Propiedades

Propiedades
-------------



```
public ComercioExteriorReceptorDomicilio Domicilio { get; set; }
public string NumRegIdTrib { get; set; }
```

### 9.10.11 Clase ComercioExteriorReceptorDomicilio

#### 9.10.11.1 Constructores

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

#### 9.10.11.2 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>

## 9.11 Referencia del Namespace SifeiSignature3\_3.Complementos.ImpLocal10

### 9.11.1 Clase ImpuestosLocales

#### 9.11.1.1 Constructores

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

#### 9.11.1.2 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>
<code>public decimal TotaldeTraslados { get; set; }</code>

### 9.11.2 Clase ImpuestosLocalesRetencionesLocales

#### 9.11.2.1 Constructores

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

#### 9.11.2.2 Propiedades

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





### 9.11.3 Clase ImpuestosLocalesTrasladosLocales

#### 9.11.3.1 Constructores

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

#### 9.11.3.2 Propiedades

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

## 9.12 Referencia del *Namespace* SifeiSignature3\_3.Complementos.Nomina12

### 9.12.1 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](#)

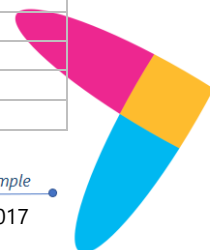
### 9.12.2 Enumeración [NominaReceptorSindicalizado](#)

Enumeraciones de NominaReceptorSindicalizado
<a href="#">NominaReceptorSindicalizado</a>

### 9.12.3 Clase [Nomina](#)

#### 9.12.3.1 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>

## 9.12.4 Clase NominaDeducciones

### 9.12.4.1 Propiedades

Propiedades
<code>public NominaDeduccionesDeduccion[] Deduccion { 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>

## 9.12.5 Clase NominaDeduccionesDeduccion

### 9.12.5.1 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>

## 9.12.6 Clase NominaEmisor

### 9.12.6.1 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>

## 9.12.7 Clase NominaEmisorEntidadSNCF

### 9.12.8 Propiedades

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



## 9.12.9 Clase NominaIncapacidad

### 9.12.9.1 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>

## 9.12.10 Clase NominaOtroPago

### 9.12.10.1 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>

## 9.12.11 Clase NominaOtroPagoCompensacionSaldosAFavor

### 9.12.11.1 Propiedades

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

## 9.12.12 Clase NominaOtroPagoSubsidioAlEmpleo

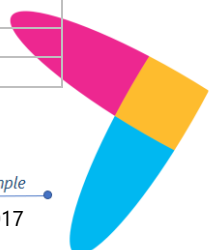
### 9.12.12.1 Propiedades

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

## 9.12.13 Clase NominaPercepciones

### 9.12.13.1 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>



### 9.12.14 Clase NominaPercepcionesJubilacionPensionRetiro

#### 9.12.14.1 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>

### 9.12.15 Clase NominaPercepcionesPercepcion

#### 9.12.15.1 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>

### 9.12.16 Clase NominaPercepcionesPercepcionAccionesOTitulos

#### 9.12.16.1 Propiedades

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

### 9.12.17 Clase NominaPercepcionesPercepcionHorasExtra

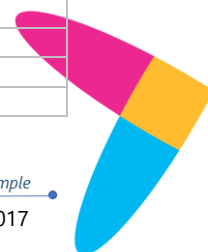
#### 9.12.17.1 Propiedades

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>

### 9.12.18 Clase NominaPercepcionesSeparacionIndemnizacion

#### 9.12.18.1 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>



### 9.12.19 Clase NominaReceptor

#### 9.12.19.1 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>
<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>

### 9.12.20 Clase NominaReceptorSubContratacion

#### 9.12.20.1 Propiedades

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

## 9.13 Referencia del Namespace SifeiSignature3\_3.Complementos.Pagos10

### 9.13.1 Clases

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

### 9.13.2 Clase Pagos

#### 9.13.2.1 Propiedades

Propiedades
-------------



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

### 9.13.3 Clase PagosPago

#### 9.13.3.1 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>
<code>public string CadPago { get; set; }</code>
<code>public byte[] SelloPago { get; set; }</code>

### 9.13.4 Clase PagosPagoDoctoRelacionado

#### 9.13.4.1 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>

### 9.13.5 Clase PagosPagoImpuestos

#### 9.13.5.1 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>

### 9.13.6 Clase PagosPagoImpuestosRetencion

#### 9.13.6.1 Propiedades

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

### 9.13.7 Clase PagosPagoImpuestosTraslado

#### 9.13.7.1 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>

## 9.14 Referencia del Namespace SifeiSignature3\_3.srvSifeiCancelacion2018

### 9.14.1 Clases

- `class folios`

### 9.14.2 Enumeraciones

- `Respuesta`

Respuesta
Aceptacion = 0,
Rechazo = 1

### 9.14.3 Clase folios

#### 9.14.3.1 Propiedades

Propiedades
<code>public respuesta respuesta { get; set; }</code>
<code>public bool respuestaSpecified { get; set; }</code>
<code>public string uuid { get; set; }</code>

## 9.15 Códigos de Respuesta de Cancelación

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

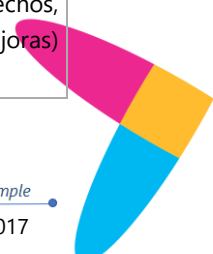
Código	Descripción
201	Cancelado
202	Previamente cancelado



Código	Descripción
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

### 9.16 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.
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.

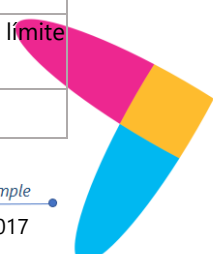




Código	Error
CFDI33119	Cuando el valor del campo Total se encuentre fuera de los límites establecidos, debe existir el campo Confirmacion.
CFDI33120	El campo TipoDeComprobante, no contiene un valor del catálogo c_TipoDeComprobante.
CFDI33121	El campo MetodoPago, no contiene un valor del catálogo c_MetodoPago.
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.
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.



Código	Error
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 Tasa0Cuota ni Importe.
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.



Código	Error
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.
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.



Código	Error
CFDI33196	Error no identificado.

### 9.17 Codigos de Error del SAT para el metodo Consulta CFDI Relacionado.

Código	Descripción
2000	Existen CFDI relacionados al folio fiscal
2001	No existen CFDI relacionados al folio fiscal
2002	El folio fiscal no pertenece al receptor
1101	No existen peticiones pendientes para el RFC Receptor

### 9.18 Codigos de error de SAT para el metodo obtener Peticiones de Cancelacion Pendientes.

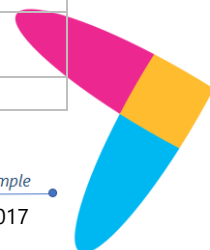
Código	Descripción
300	Usuario no valido
301	XML mal formado
1100	Se obtuvo petición del RFC receptor de forma exitosa
1101	No existen peticiones para el RFC del receptor

### 9.19 Codigos de error del SAT para el metodo Aceptacion/Rechazo de cancelacion (procesa Respuesta).

Código	Descripción
300	Usuario no válido
301	XML mal formado
302	Sello mal formado
304	Certificado revocado o caduco
305	Certificado inválido
309	Patrón de folio inválido
310	CSD inválido
1000	Se recibió la respuesta de la petición de forma exitosa
1001	No existen peticiones de cancelación en espera de respuesta para el UUID
1002	Ya se recibió una respuesta para la petición de cancelación del UUID
1003	Sello no corresponde al RFC receptor
1004	Existen más de una petición de cancelación para el mismo UUID
1005	El UUID es nulo no posee el formato correcto
1006	Se rebasó el número máximo de solicitudes permitidas

### 9.20 Codigos de respuesta del servicio de consulta CFDI.

Atributo	Descripción
Estado	"Vigente","Cancelado","No encontrado"
CodigoEstatus	"S - Comprobante obtenido satisfactoriamente"," N - 601: La expresión impresa proporcionada no es válida"," N - 602: Comprobante no encontrado"
EsCancelable	"No cancelable"," Cancelable sin aceptación"," Cancelable con aceptación"





## 10. 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](mailto: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	<a href="http://www.sifei.com.mx/">http://www.sifei.com.mx/</a>
Facebook	<a href="http://www.facebook.com/sifeimexico">http://www.facebook.com/sifeimexico</a>
Twitter	<a href="http://twitter.com/#!/SIFEIMexico">http://twitter.com/#!/SIFEIMexico</a>
YouTube	<a href="https://www.youtube.com/channel/UCDSLHrFOg2tGvG3AruHNmaw">https://www.youtube.com/channel/UCDSLHrFOg2tGvG3AruHNmaw</a>
LinkedIn	<a href="https://www.linkedin.com/company/18265037/">https://www.linkedin.com/company/18265037/</a>

### UBICACIÓN DE OFICINA MATRIZ

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

<Fin del Documento>

