



Manual Técnico de SIFEI Signature

CFDI v4.0

Mayo 2023

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

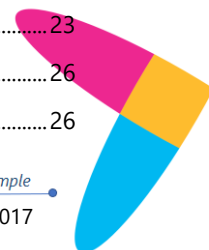
Control de Autorizaciones

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

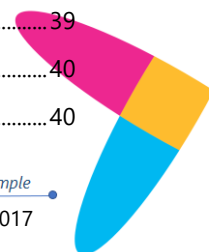


Contenido

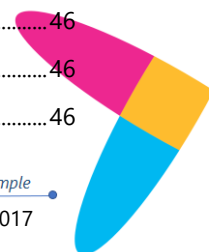
Listado de Imágenes.....	xi
1. Generalidades.....	1
1.1 Introducción.....	1
1.2 Objetivo.....	1
1.3 Alcance.....	1
1.4 Requerimientos Mínimos.....	1
1.5 Términos y Definiciones.....	1
1.6 Términos y definiciones técnicas.....	2
1.7 Referencias.....	2
1.8 Recursos.....	2
1.9 Versiones de CFDI que se timbran.....	2
1.9.1 Complementos para CFDI 4.0.....	2
1.9.2 Complementos para Comprobante de Retención e Información de Pagos.....	3
1.9.3 Métodos de Timbrado CFDI 4.0.....	3
1.9.4 Métodos de Timbrado Comprobante de Retención e Información de Pagos.....	3
1.9.5 Metodos de proceso de cancelación CFDI 4.0.....	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 dll 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 CFDI40.....	26



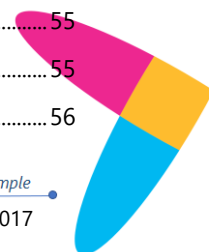
4.2	Establecer información del nodo InformacionGlobal.....	27
4.3	Agregar CfdiRelacionados	27
4.4	Establecer información del Emisor.....	27
4.5	Establecer información del Receptor	28
4.6	Agregar un Concepto	28
4.7	Agregar Información aduanera a un Concepto.....	29
4.8	Agregar un impuesto traslado a un concepto	29
4.9	Agregar un impuesto retenido a un concepto	29
4.10	Establecer información de A Cuenta de Terceros.....	30
4.11	Agregar Cuenta Predial al concepto	30
4.12	Establecer información del total impuestos de comprobante	30
4.13	Agregar un impuesto traslado a impuestos de comprobante.....	31
4.14	Autorización para el uso de Sifei Signature	31
4.15	Generar el XML.....	32
4.16	Sellar el comprobante	34
4.17	Timbrar el comprobante.....	35
4.18	Recuperar timbre.....	36
5.	Crear Complementos	36
5.1	Complemento Comercio Exterior	36
5.1.1	Definir complemento Comercio Exterior	36
5.1.2	Establecer Emisor de Comercio Exterior	37
5.1.3	Definir Receptor de Comercio Exterior	37
5.1.4	Agregar mercancía de Comercio Exterior	37
5.1.5	Generar el Complemento Comercio Exterior.....	37
5.2	Complemento Comercio Exterior 2.0.....	38
5.2.1	Definir complemento Comercio Exterior	38
5.2.2	Establecer Emisor de Comercio Exterior	38
5.2.3	Definir Receptor de Comercio Exterior	38
5.2.4	Agregar mercancía de Comercio Exterior	38
5.2.5	Generar el Complemento Comercio Exterior.....	39
5.3	Complemento Nómina	39
5.3.1	Definir complemento Nómina	39
5.3.2	Agregar un emisor a Nómina	39
5.3.3	Agregar un receptor a Nómina.....	39
5.3.4	Establecer que Nómina tiene percepciones	40
5.3.5	Agregar una percepción a percepciones de Nómina.....	40



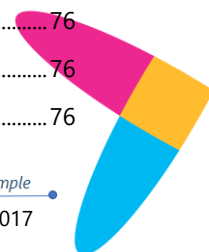
5.3.6	Agregar una Separación indemnización a percepciones de Nómina	40
5.3.7	Establar deducciones a Nómina	40
5.3.8	Agregar una deducción a deducciones de Nómina	40
5.3.9	Generar el Complemento Nómina	40
5.4	Complemento Detallista.....	41
5.4.1	Definir complemento Detallista.....	41
5.4.2	Establecer el requestForPaymentIdentification a Detallista.....	41
5.4.3	Agregar un specialInstruction a Detallista.....	41
5.4.4	Establecer el orderIdentification a Detallista.....	41
5.4.5	Agregar un referenceIdentification al AdditionalInformation de Detallista	41
5.4.6	Establecer un DeliveryNote a Detallista	41
5.4.7	Definir el Buyer (comprador) de Detallista	41
5.4.8	Definir el Seller (vendedor) a Detallista	41
5.4.9	Establar el shipTo (lugar de envío) de Detallista	42
5.4.10	Establecer el InvoiceCreator a Detallista.....	42
5.4.11	Agregar un Customs a Detallista.....	42
5.4.12	Agregar un currency a Detallista	42
5.4.13	Establecer el paymentTerms a Detallista	42
5.4.14	Establecer un allowanceCharge a Detallista para definir cargos o descuentos	43
5.4.15	Definir el totalAmount a Detallista	43
5.4.16	Definir el TotalAllowanceCharge a Detallista para definir el monto total de cargos o descuentos	43
5.4.17	Generar el complemento Detallista	43
5.5	Complemento Recepción de Pagos 2.0.....	43
5.5.1	Definir el nodo Totales	43
5.5.2	Agregar un Pago	44
5.5.3	Agregar un DoctoRelacionado al último Pago agregado.....	44
5.5.4	Agregar un Impuesto TrasladoDR / RetencionDR al último DoctoRelacionado agregado	44
5.5.5	Agregar un Impuesto TrasladoP al ultimo Pago agregado.	45
5.5.6	Agregar un Impuesto RetencionP al ultimo Pago agregado.	45
5.5.7	Generar el Complemento Pago	45
5.6	Definir y generar el complemento IEDU (a un concepto)	45
5.7	Definir y generar el complemento Hidrocarburos y Petrolíferos 1.0 (a un concepto).....	46
5.8	Definir y generar el complemento Donatarias 1.1	46
5.9	Complemento Carta Porte 2.0	46
5.9.1	Definir el complemento Carta Porte.....	46
5.9.2	Agregar una Ubicación.....	46



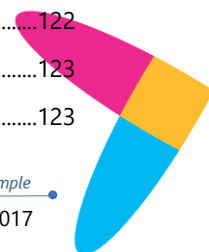
5.9.3	Establecer el Domicilio para una ubicación.....	47
5.9.4	Establecer información de Mercancías.....	47
5.9.5	Agregar una Mercancía.....	47
5.9.6	Agregar CantidadTransporta a la última Mercancía.....	48
5.9.7	Establecer elemento Autotransporte para Mercancías.....	48
5.9.8	Establecer información de Seguros para el Autotransporte.....	48
5.9.9	Agregar un Tipo de Figura para FiguraTransporte.....	48
5.9.10	Generar el Complemento Carta Porte 2.0.....	49
5.10	Complemento Carta Porte 3.0.....	49
5.10.1	Definir el complemento Carta Porte.....	49
5.10.2	Agregar una Ubicación.....	49
5.10.3	Establecer el Domicilio para una ubicación.....	49
5.10.4	Establecer información de Mercancías.....	50
5.10.5	Agregar una Mercancía.....	50
5.10.6	Agregar DocumentacionAduanera a la última Mercancía.....	51
5.10.7	Agregar CantidadTransporta a la última Mercancía.....	51
5.10.8	Establecer elemento Autotransporte para Mercancías.....	51
5.10.9	Establecer información de Seguros para el Autotransporte.....	51
5.10.10	Agregar un Tipo de Figura para FiguraTransporte.....	51
5.10.11	Generar el Complemento Carta Porte 3.0.....	52
5.11	Complemento Carta Porte 3.1.....	52
5.11.1	Definir el complemento Carta Porte.....	52
5.11.2	Agregar un Régimen Aduanero.....	52
5.11.3	Agregar una Ubicación.....	52
5.11.4	Establecer el Domicilio para una ubicación.....	53
5.11.5	Establecer información de Mercancías.....	53
5.11.6	Agregar una Mercancía.....	53
5.11.7	Agregar DocumentacionAduanera a la última Mercancía.....	54
5.11.8	Agregar CantidadTransporta a la última Mercancía.....	54
5.11.9	Establecer elemento Autotransporte para Mercancías.....	54
5.11.10	Establecer información de Seguros para el Autotransporte.....	55
5.11.11	Agregar un Tipo de Figura para FiguraTransporte.....	55
5.11.12	Generar el Complemento Carta Porte 3.1.....	55
5.12	Complemento Impuestos Locales.....	55
5.12.1	Definir el complemento Impuestos Locales.....	55
5.12.2	Agregar Impuesto Local Traslado al complemento.....	56



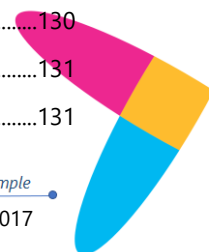
5.12.3	Generar el Complemento Impuestos Locales	56
6.	Proceso de Cancelación	56
6.1	Cancelar un CFDI 2018 (actualización)	56
6.2	Cancelar CFDIs Nuevo Proceso	57
6.3	Consulta CFDI relacionados	58
6.4	Consulta CFDI Estatus	59
6.5	Consulta peticiones de Cancelación pendientes (que requieren autorización)	60
6.6	Procesar Respuesta de Peticion de Cancelacion (Aceptacion o Rechazo de cancelación)	61
7.	Comprobante de Retención de Información de Pagos	62
7.1	Instanciar un objeto de Comprobante de Retención e Información de Pagos	62
7.2	Definir la retención	62
7.3	Definir CfdiRetenRelacionados	63
7.4	Definir el Emisor	63
7.5	Definir Receptor	63
7.6	Definir el Periodo	64
7.7	Definir Totales	64
7.8	Agregar Impuesto Retenido	64
7.9	Autorización para el uso de Sifei Signature	65
7.10	Generar el XML	65
7.11	Sellar el Comprobante de Retención e Información de Pagos	66
7.12	Timbrar el Comprobante de Retenciones e Información de Pagos	67
8.	Complementos de Comprobante de Retenciones e Información de Pagos	68
8.1	Complemento Servicios Plataformas Tecnológicas	68
8.1.1	Definir Complemento Servicios Plataformas Tecnológicas	68
8.1.2	Agregar DetallesDelServicio	68
8.1.3	Generar Complemento Servicios Plataformas Tecnológicas	69
9.	Referencia	70
9.1	<i>Namespaces</i>	70
9.2	Clases principales	70
9.3	<i>Namespace SifeiSignature3_3.Version3_3</i>	70
9.3.1	Clase CFDI33	71
9.3.2	Métodos	71
9.3.3	Enumeraciones:	76
9.4	<i>Namespace SifeiSignature3_3.Timbre11</i>	76
9.4.1	Clases	76
9.4.2	Clase TimbreBE	76



9.5	Namespace SifeiSignature3_3.Complementos	76
9.5.1	Clase Complemento	76
9.6	Namespace SifeiSignature3_3>Returns	111
9.6.1	Clase ReturnXML	111
9.6.2	Clase ReturnTimbre	111
9.6.3	Clase ReturnXmlComplemento	112
9.6.4	Clase ReturnCancelacion2018	112
9.6.5	Clase ReturnDocumentosRelacionados	113
9.6.6	Clase ReturnQrAcuse	113
9.6.7	Clase ReturnPeticonesPendientes	114
9.6.8	Clase ReturnAceptacionRechazoResult	114
9.6.9	Clase DetalleFolios	115
9.7	Namespace SifeiSignature3_3.Complementos.Detallista131	115
9.7.1	Enumeraciones de complemento Detallista versión 1.3.1	115
9.8	Namespace SifeiSignature3_3.Complementos.Iedu10	116
9.8.1	Enumeraciones de complemento Instituciones Educativas versión 1.0	116
9.8.2	Enumeraciones de complemento Nómina versión 1.2	116
9.9	Referencia del Namespace SifeiSignature3_3.Cancelacion	116
9.9.1	Clase Acuse	116
9.9.2	Clase AcuseDetalle	117
9.9.3	Clase Cancelacion (deprecada, obsoleta)	117
9.9.4	Clase Cancelacion 2018	118
9.10	Referencia del Namespace SifeiSignature3_3.Complementos.Cce11	119
9.10.1	Clases	119
9.10.2	Clase ComercioExterior	119
9.10.3	Clase ComercioExteriorDestinatario	120
9.10.4	Clase ComercioExteriorDestinatarioDomicilio	120
9.10.5	Clase ComercioExteriorEmisor	121
9.10.6	Clase ComercioExteriorEmisor	121
9.10.7	Clase ComercioExteriorMercancia	121
9.10.8	Clase ComercioExteriorMercanciaDescripcionesEspecificas	122
9.10.9	Clase ComercioExteriorPropietario	122
9.10.10	Clase ComercioExteriorReceptor	122
9.10.11	Clase ComercioExteriorReceptorDomicilio	122
9.11	Referencia del Namespace SifeiSignature3_3.Complementos.ImpLocal10	123
9.11.1	Clase ImpuestosLocales	123



9.11.2	Clase ImpuestosLocalesRetencionesLocales	123
9.11.3	Clase ImpuestosLocalesTrasladosLocales.....	123
9.12	Referencia del <i>Namespace</i> SifeiSignature3_3.Complementos.Nomina12.....	124
9.12.1	Clases	124
9.12.2	Enumeración NominaReceptorSindicalizado	124
9.12.3	Clase Nomina.....	124
9.12.4	Clase NominaDeducciones	125
9.12.5	Clase NominaDeduccionesDeduccion	125
9.12.6	Clase NominaEmisor	125
9.12.7	Clase NominaEmisorEntidadSNCF.....	125
9.12.8	Propiedades.....	125
9.12.9	Clase NominaIncapacidad	125
9.12.10	Clase NominaOtroPago.....	126
9.12.11	Clase NominaOtroPagoCompensacionSalDOSAFavor	126
9.12.12	Clase NominaOtroPagoSubsidioAlEmpleo.....	126
9.12.13	Clase NominaPercepciones.....	126
9.12.14	Clase NominaPercepcionesJubilacionPensionRetiro	126
9.12.15	Clase NominaPercepcionesPercepcion.....	127
9.12.16	Clase NominaPercepcionesPercepcionAccionesOTitulos.....	127
9.12.17	Clase NominaPercepcionesPercepcionHorasExtra	127
9.12.18	Clase NominaPercepcionesSeparacionIndemnizacion	127
9.12.19	Clase NominaReceptor	127
9.12.20	Clase NominaReceptorSubContratacion	128
9.13	Referencia del <i>Namespace</i> SifeiSignature3_3.Complementos.Pagos10	128
9.13.1	Clases	128
9.13.2	Clase Pagos	128
9.13.3	Clase PagosPago.....	129
9.13.4	Clase PagosPagoDoctoRelacionado.....	129
9.13.5	Clase PagosPagoImpuestos.....	129
9.13.6	Clase PagosPagoImpuestosRetencion	130
9.13.7	Clase PagosPagoImpuestosTraslado	130
9.14	Referencia del <i>Namespace</i> SifeiSignature3_3.Complementos.cartaporte20	130
9.14.1	Clases	130
9.14.2	Clase CartaPorte	130
9.14.3	Clase CartaPorteMercancias	131
9.14.4	Clase CartaPorteMercanciasAutotransporte.....	131



9.14.5	Clase CartaPorteMercanciasAutotransporteIdentificacionVehicular.....	131
9.14.6	Clase CartaPorteMercanciasAutotransporteRemolque	132
9.14.7	Clase CartaPorteMercanciasAutotransporteSeguros.....	132
9.14.8	Clase CartaPorteMercanciasMercancia.....	132
9.15	Referencia del <i>Namespace SifeiSignature3_3.srvSifeiCancelacion2018</i>	133
9.15.1	Clases	133
9.15.2	Enumeraciones.....	133
9.15.3	Clase folios.....	133
9.16	Códigos de Respuesta de Cancelación	133
9.17	Códigos de Error de la Matriz de validaciones para CFDI versión 4.0	134
9.18	Codigos de Error del SAT para el metodo Consulta CFDI Relacionado.	134
9.19	Codigos de error de SAT para el metodo obtener Peticiones de Cancelacion Pendientes.....	134
9.20	Codigos de error del SAT para el metodo Aceptacion/Rechazo de cancelacion (procesa Respuesta).	134
9.21	Codigos de respuesta del servicio de consulta CFDI.	134
10.	Información de Contacto con SIFEI	136



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
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 tanto debe ser cambiado a 1	22
Imagen 30 Error al emitir un Cfdi cuando Mysql reside un servidor Unix y la variable de sistema <code>lower_case_table_names</code> tiene valor 0.....	23
Imagen 31 Configurando el valor de <code>lower_case_table_names= 1</code>	24

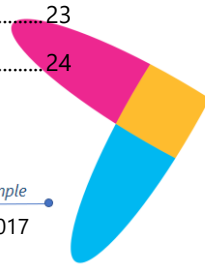


Imagen 32 Consulta de variable del sistema lower_case_table_names , donde se muestra el cambio del valor de 0 a 1 (en un Centos) tras reiniciar Mysql Server, notar que mysql client pierde la conexión y se conecta de nuevo debido al reinicio de Mysql Server.	25
Imagen 33 Atributos de CFDI agregados.....	26
Imagen 34 Nodo InformacionGlobal agregado	27
Imagen 35 Nodo CfdiRelacionados agregado.....	27
Imagen 36 Nodo Emisor agregado.....	28
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 Resultado de agregar un impuesto traslado al concepto.....	30
Imagen 42 Resultado de agregar un impuesto traslado al concepto.....	30
Imagen 42 Resultado de agregar un impuesto traslado al concepto.....	30
Imagen 41 Nodo Impuestos agregado a comprobante	31
Imagen 42 Impuesto traslado agregado	31
Imagen 43 Autenticandose para el uso de Sifei Signature.....	31
Imagen 44 Error de autenticación.....	32
Imagen 45 Comprobante Sellado	35
Imagen 46 Comprobante timbrado	36
Imagen 47 Atributos de Retenciones agregados	63
Imagen 48 Nodo CfdiRetenRelacionados agregado.....	63
Imagen 48 Nodo Emisor agregado	63
Imagen 49 Nodo Receptor agregado.....	64
Imagen 50 Nodo Periodo agregado	64
Imagen 51 Nodo Totales agregado.....	64
Imagen 52 Nodo ImpRetenidos agregado.....	65
Imagen 53 Comprobante Sellado	67
Imagen 54 Comprobante timbrado	68



1. Generalidades

1.1 Introducción

El presente documento es el *Manual Técnico* para el uso e integración de la biblioteca *SIFEI Signature* versión 4.0 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 4.0.
- 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)	http://omawww.sat.gob.mx/tramitesyservicios/Paginas/anexo_20.htm
Manual técnico de Cancelación Sifei.	https://www.sifei.com.mx/slides/slide/sifei-web-service-manual-tecnico-del-nuevo-servicio-de-cancelacion-99

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 4.0
- Comprobante de Retención e Información de Pagos

1.9.1 Complementos para CFDI 4.0

- Comercio Exterior 1.1
- Comercio Exterior 2.0
- Nómina 1.2 Rev. C



- Detallista
- Pagos 2.0
- Impuesto Local
- IEDU
- Carta Porte 2.0
- Carta Porte 3.0
- Donatarias 1.1

1.9.2 Complementos para Comprobante de Retención e Información de Pagos

- Servicios de Plataformas Tecnológicas

1.9.3 Métodos de Timbrado CFDI 4.0

- 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 4.0

- 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.

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

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

XPath:
 Cfdi:Impuestos

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



3. Primeros pasos (*getting started*)

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

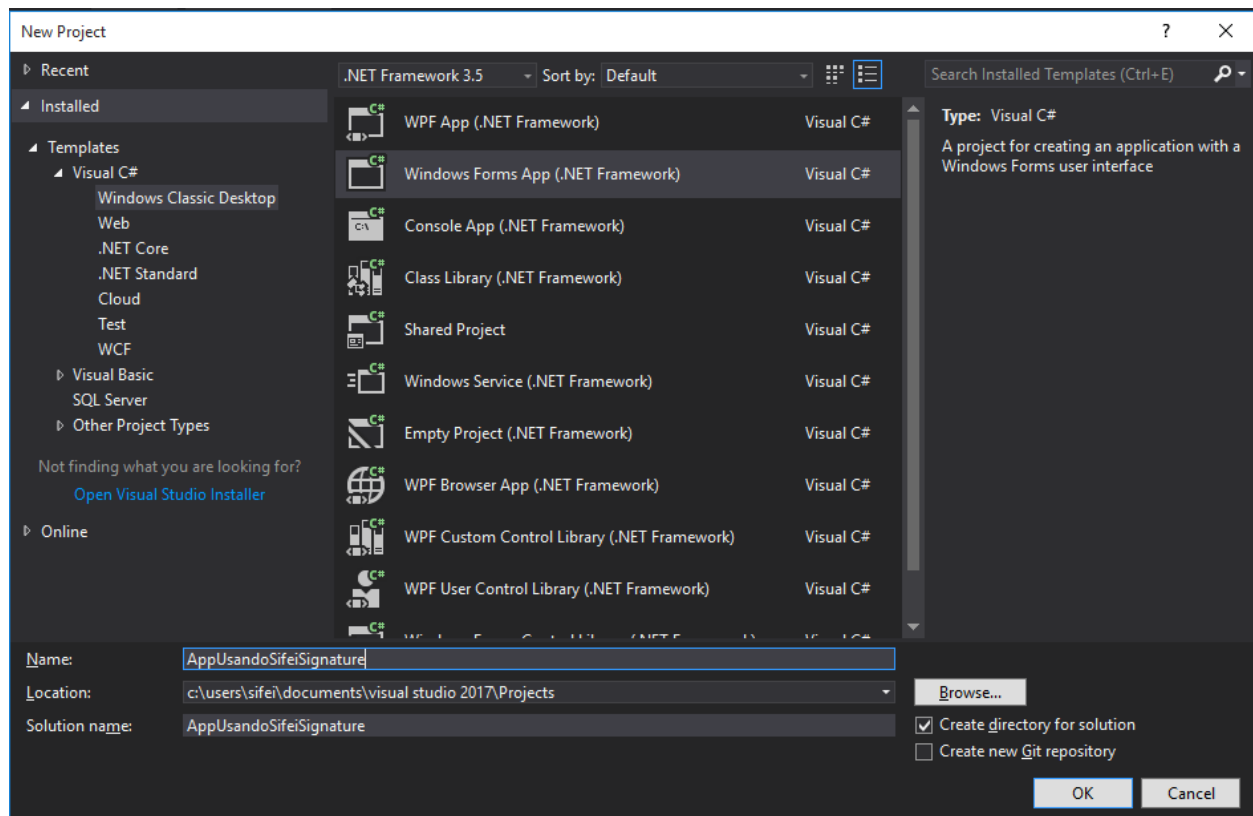


Imagen 1 Creación de proyecto .Net Framework 3.5

Lo cual debe generar algo similar a la siguiente imagen:

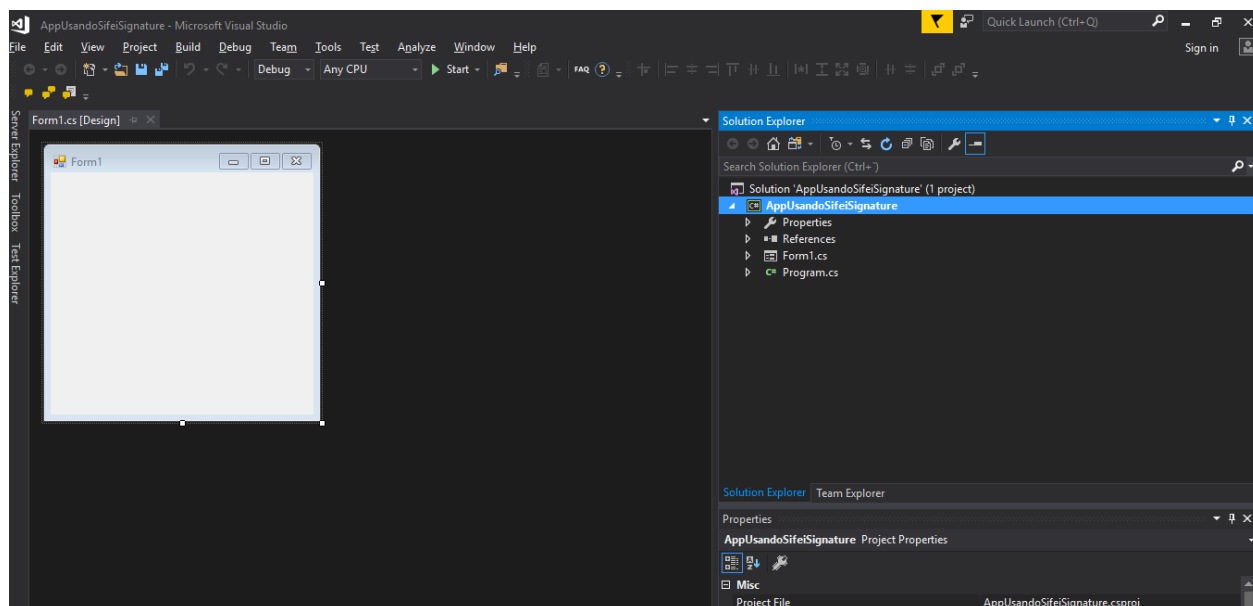


Imagen 2 Proyecto recién creado

3.1 Agregar recursos

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

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

3.1.1 Copiar la carpeta SAT y las librerías

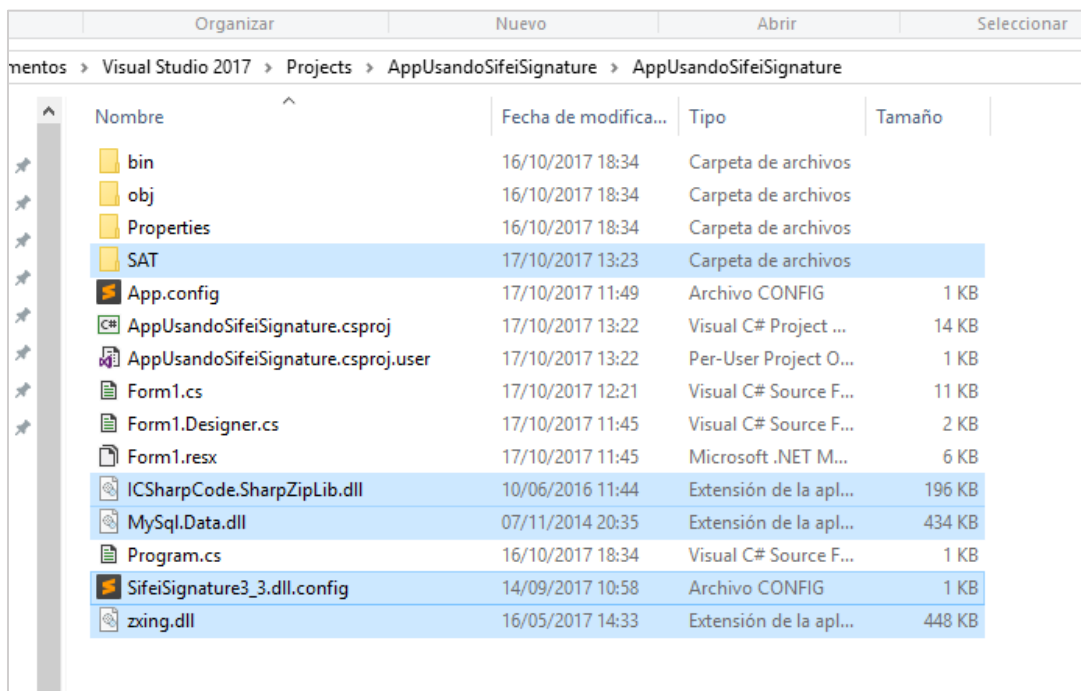


Imagen 3 Copiando los recursos al proyecto

3.1.2 Incluir la carpeta SAT y d11 en el proyecto

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



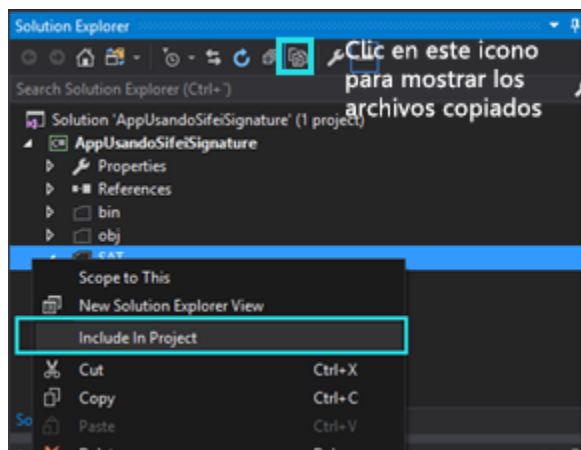


Imagen 4 Inclusión de carpeta SAT en el proyecto

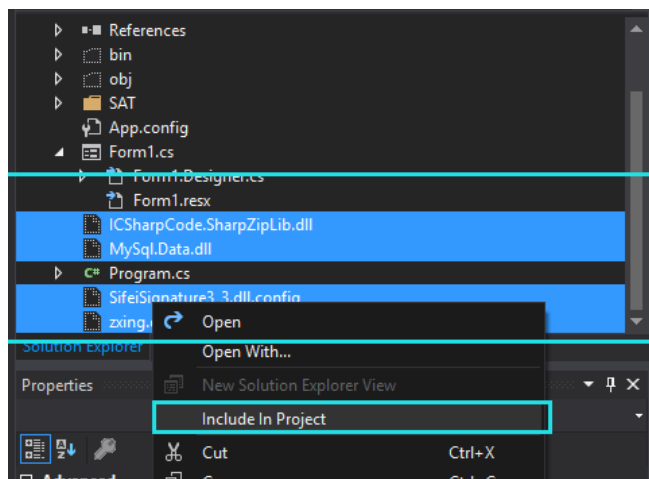


Imagen 5 Inclusión del resto de bibliotecas

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.



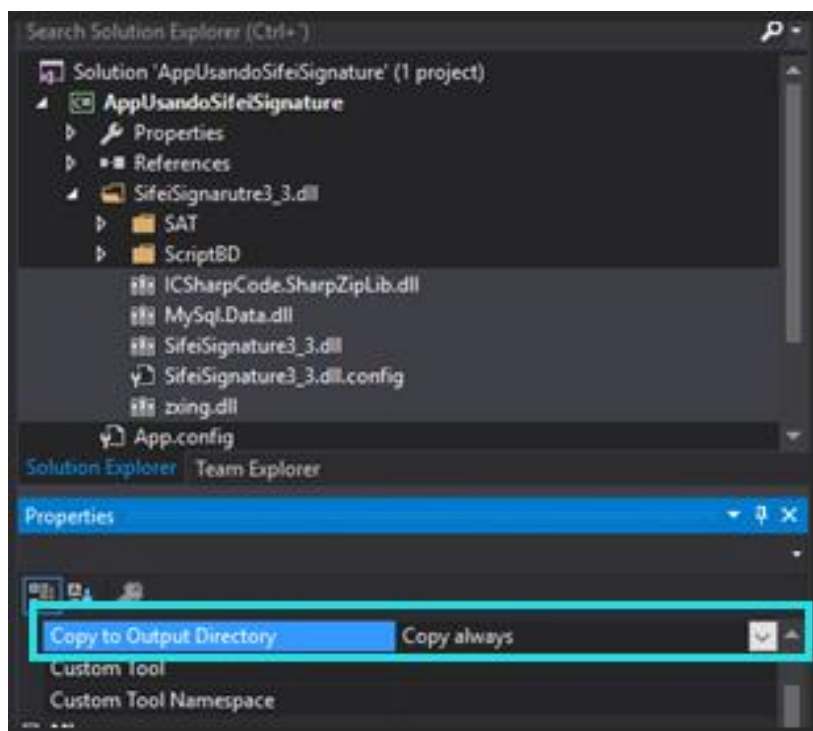


Imagen 6 Modificando la propiedad "Copiar al directorio de Salida"

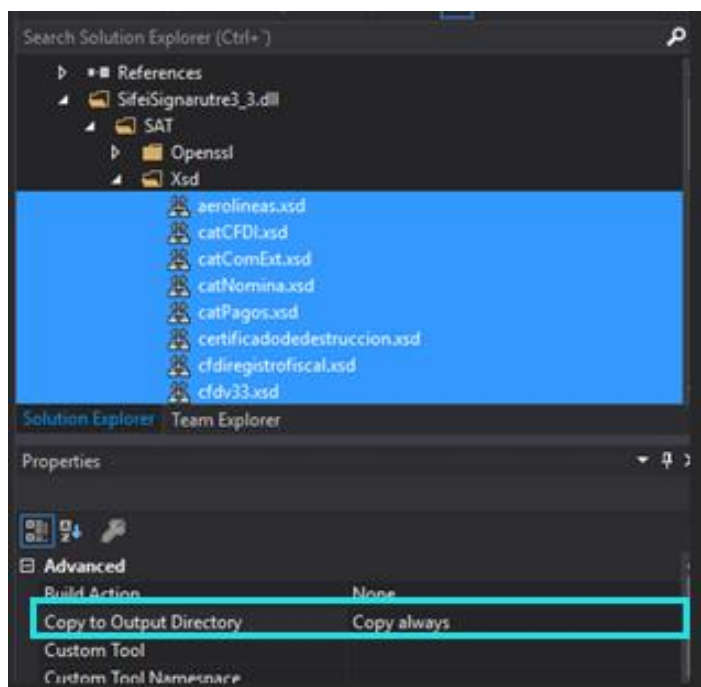


Imagen 7 Modificando la propiedad "Copiar al directorio de salida" a "Copiar siempre"

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).



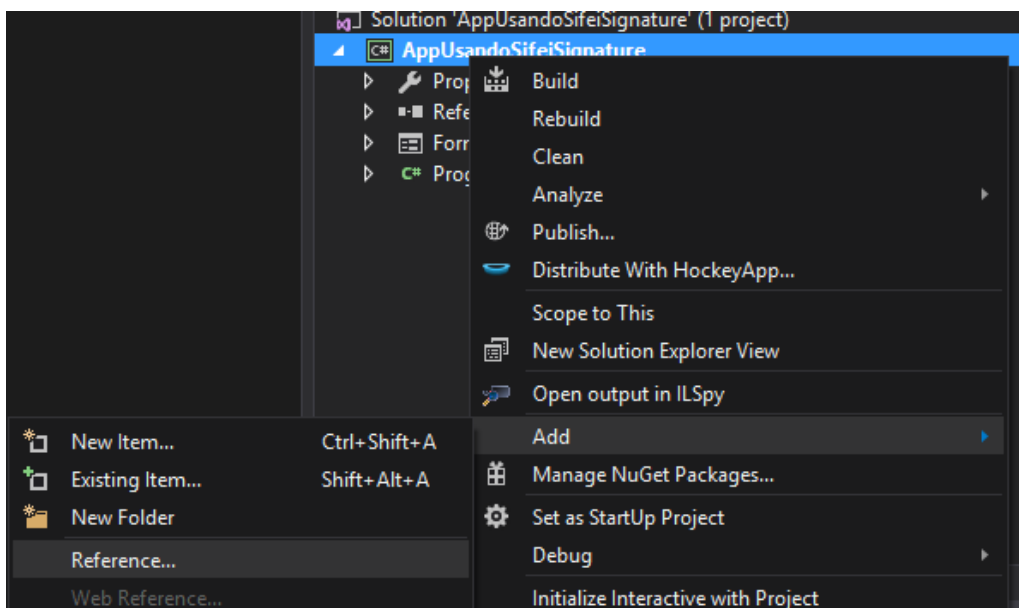


Imagen 8 Agregar referencia al proyecto

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

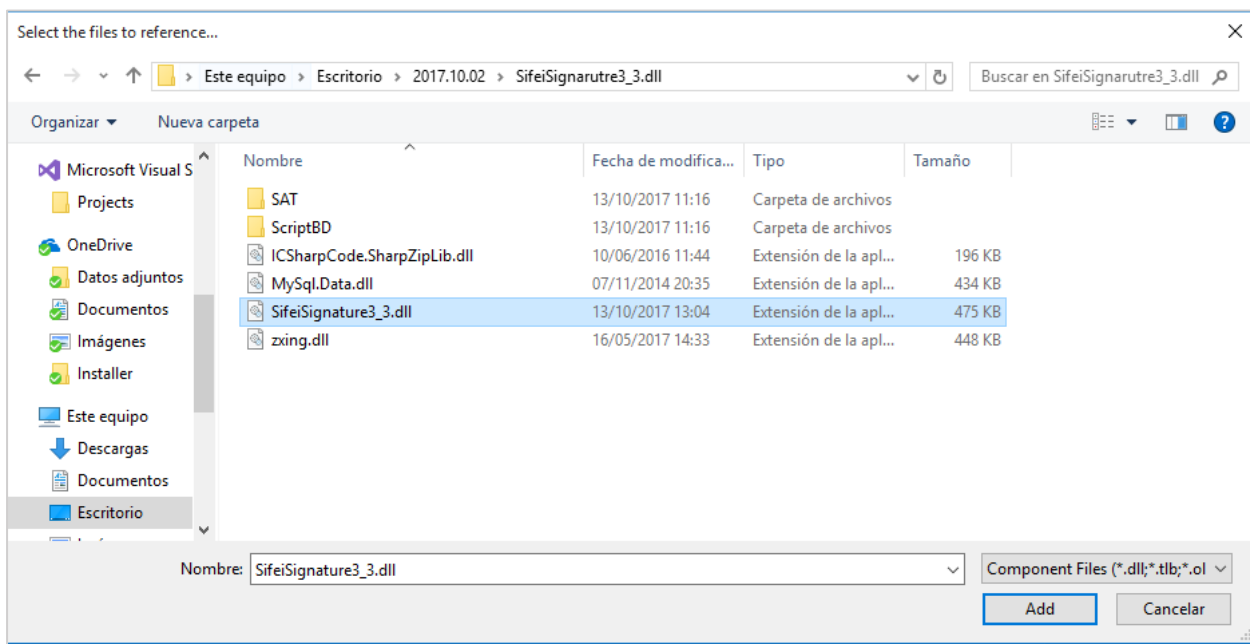


Imagen 9 Carpeta donde está SifeiSignature



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–.

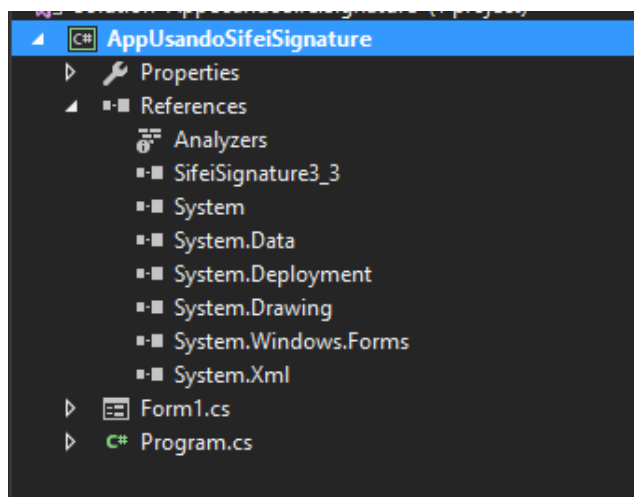


Imagen 10 Referencia de biblioteca agregada

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://devcfdi.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 5 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".



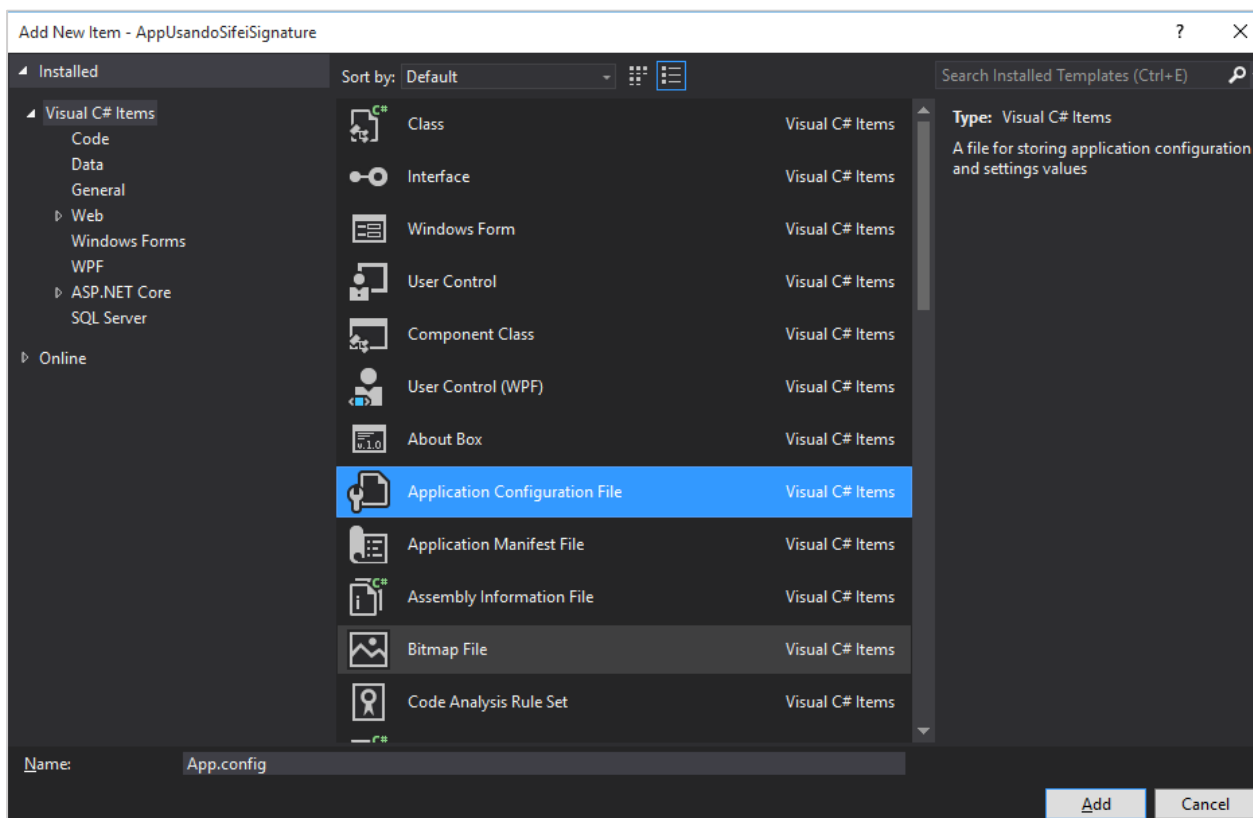


Imagen 12 Seleccionado Archivo de Configuración de Aplicación

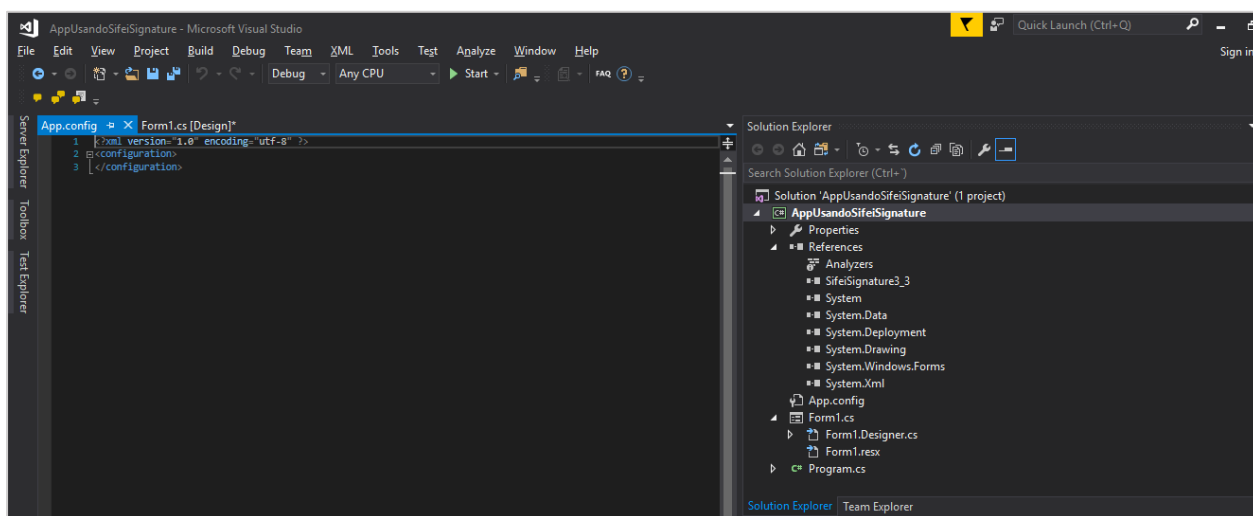
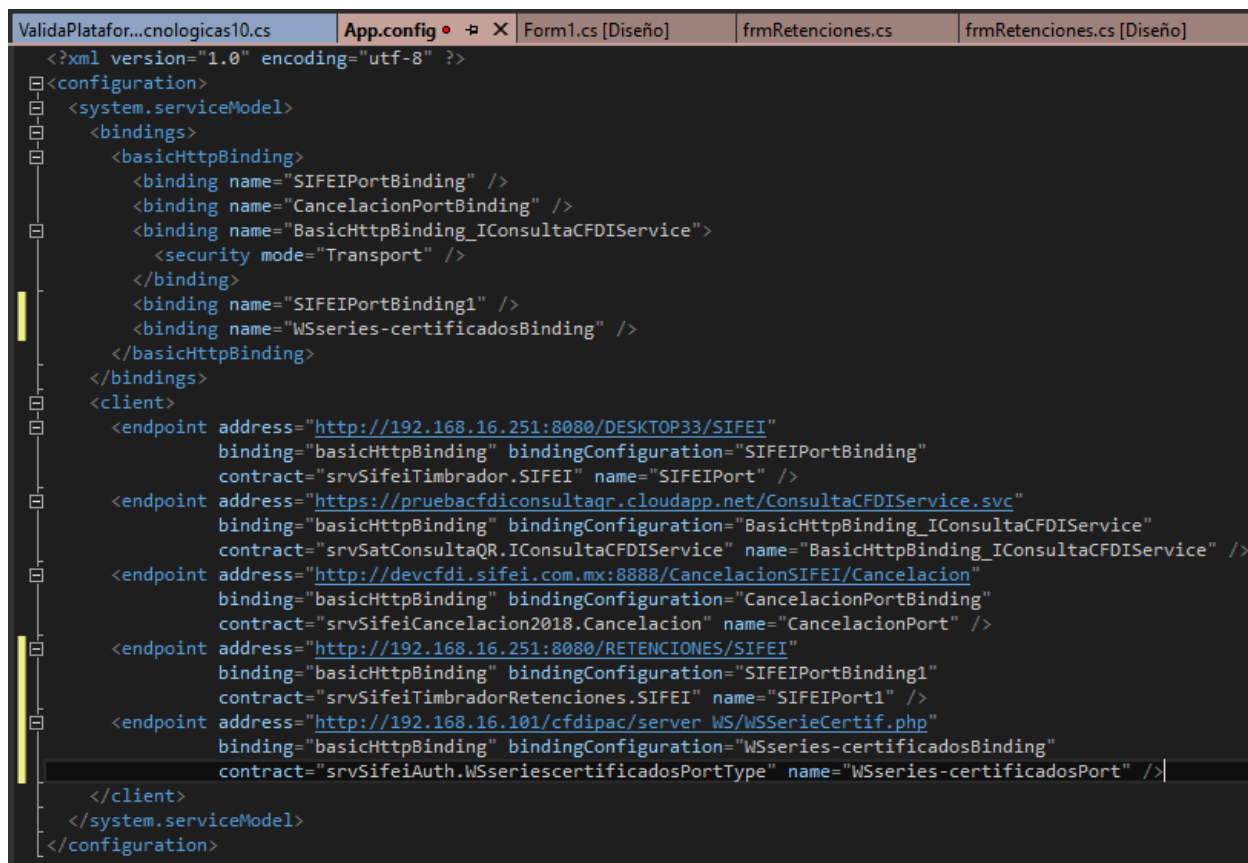


Imagen 13 Archivo de configuración limpio

Copiar el `system.servicemodel`.

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





```
<?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

```
MariaDB [(none)]>
```

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).

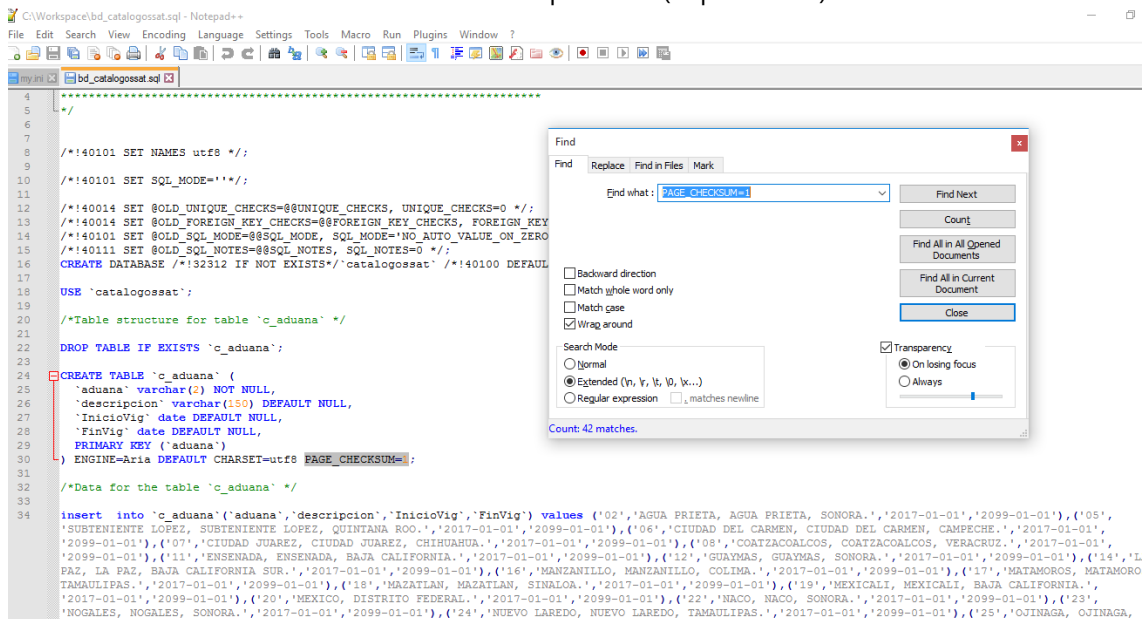


Imagen 24 Vista del catalogossat que muestra PAGE_CHECKSUM y el numero de ocurrencias

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



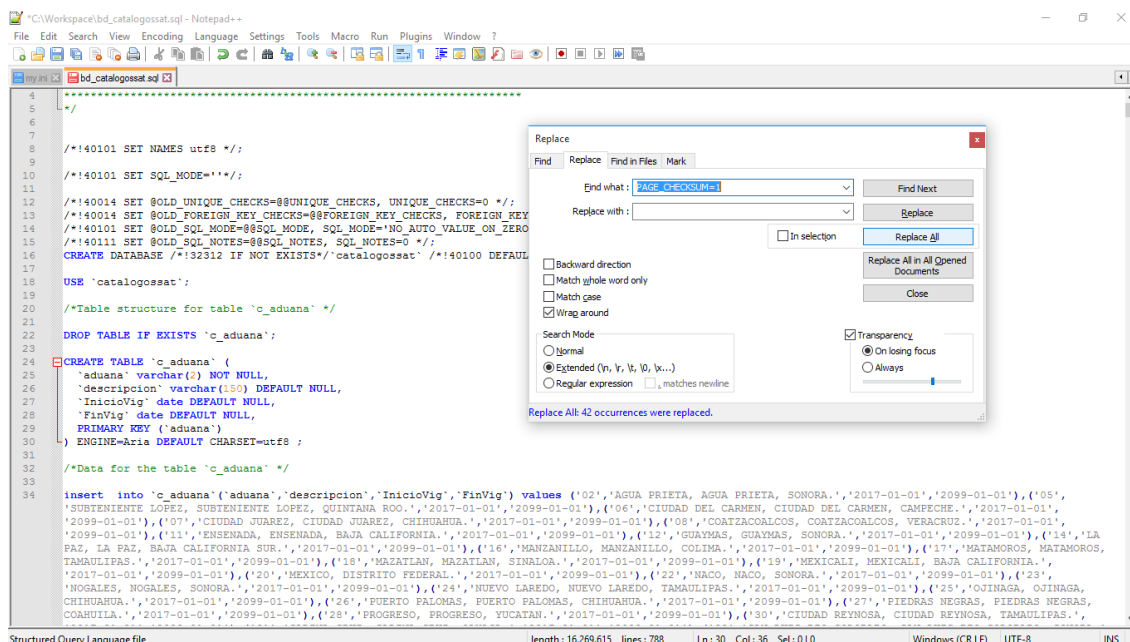
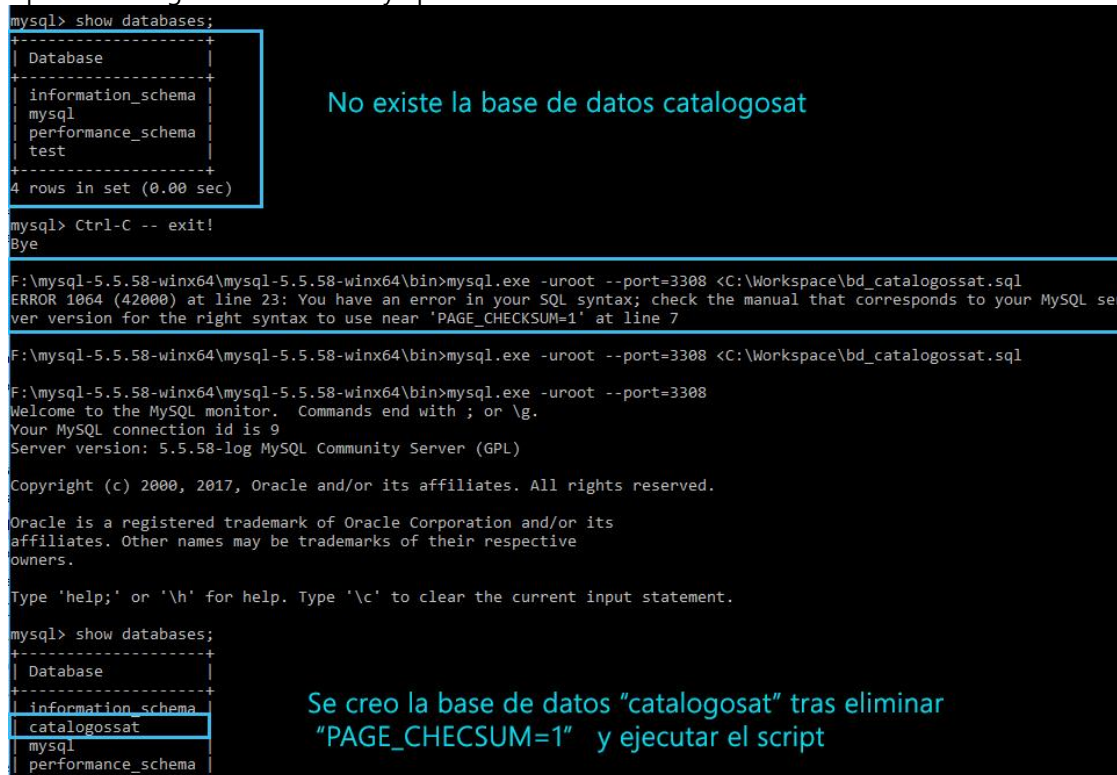


Imagen 25 Catalogosat sin PAGE_CHECKSUM

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='catalogossat';
```



```
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_patenteaduanal	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_tiponomina	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:

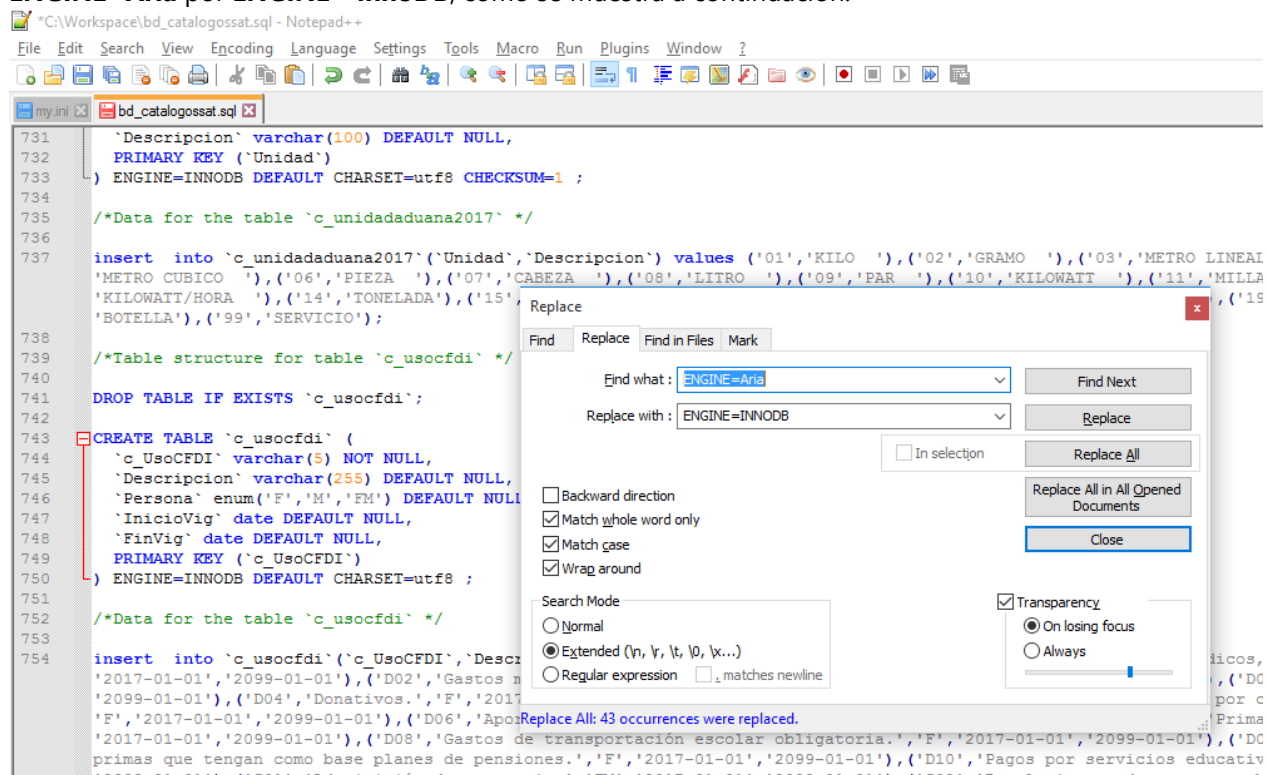


Imagen 27 Captura de reemplazar todas las ocurrencias de "ENGINE=Aria" por "ENGINE=InnoDB"

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
-------	-------------

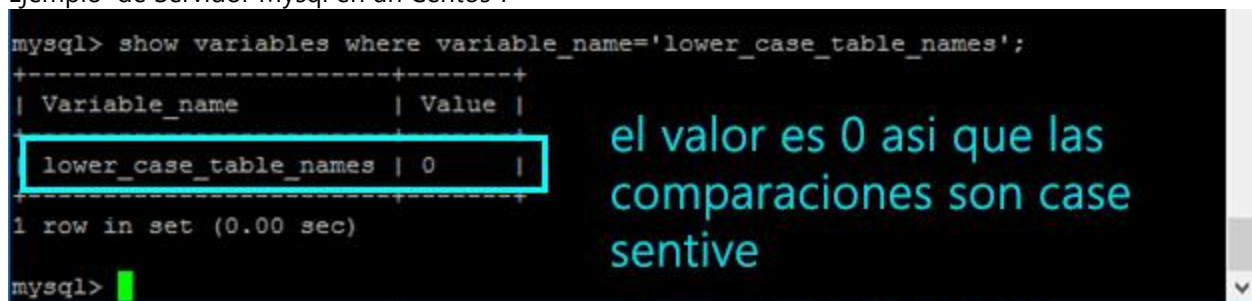
0	Los nombre de tablas y base de datos son almacenados respetando el lettercase (tal cual estan escritas) de las sentencias the CREATE TABLE or CREATE DATABASE . La compacion de nombres es sensible a minisculas y mayusculas.
1	Los nombre de tablas son almacenadas en minisculas en el disco y las comparaciones de nombres son insensibles . 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 CREATE TABLE o CREATE DATABASE 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.**



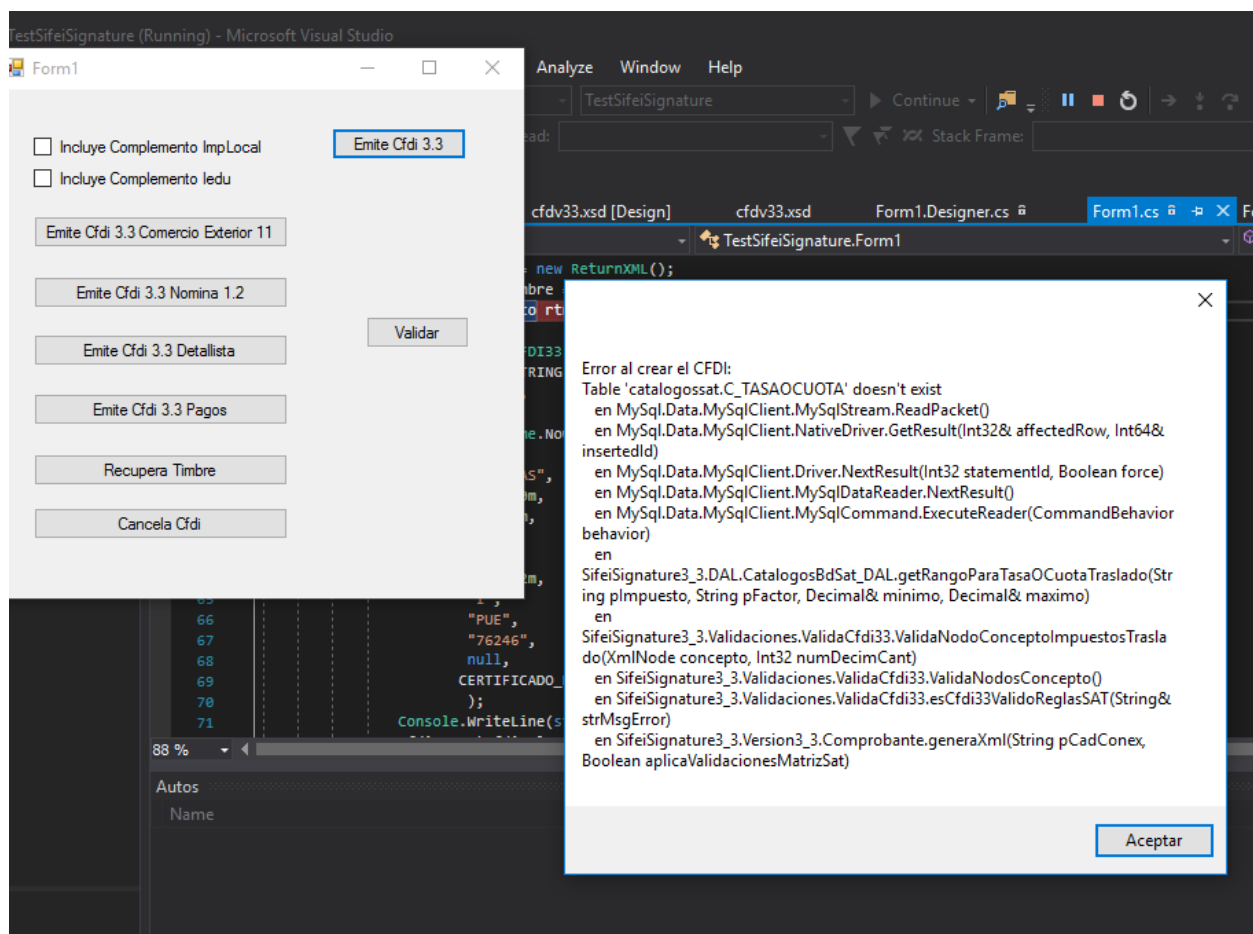
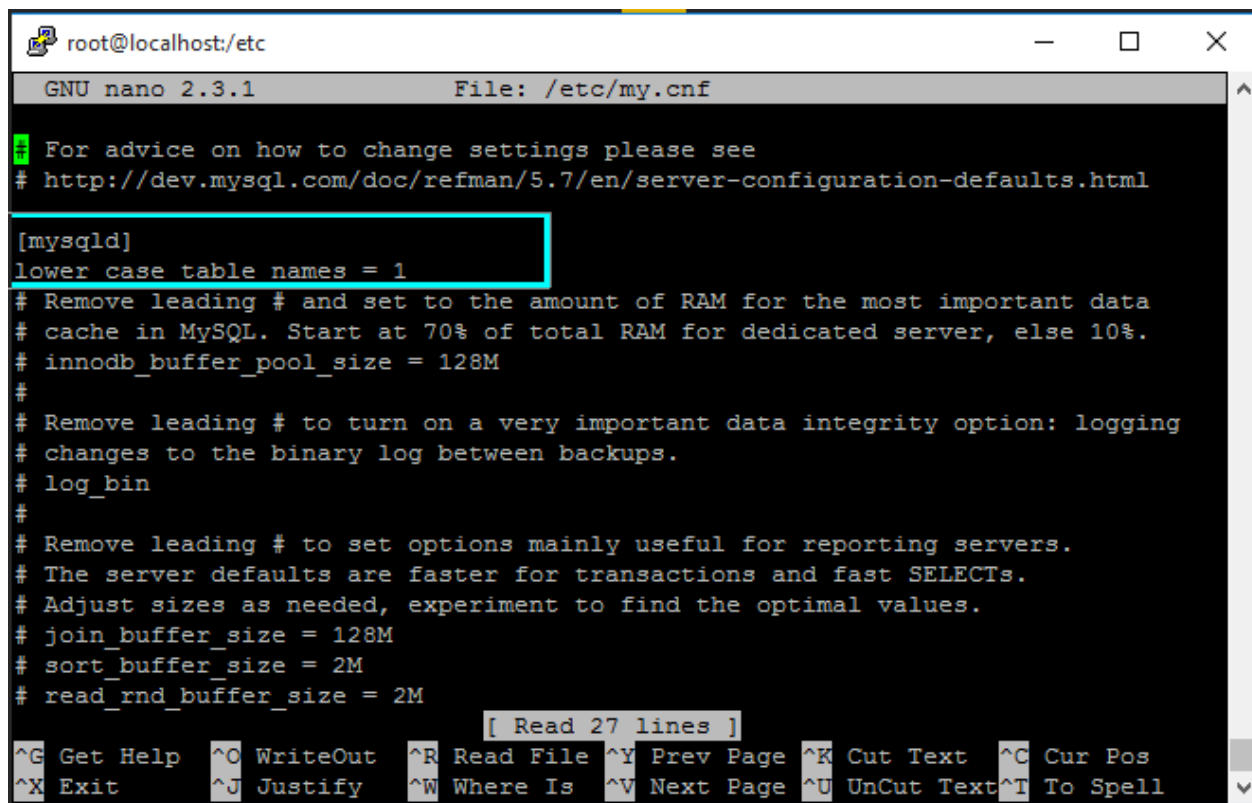


Imagen 30 Error al emitir un Cfdi cuando Mysql reside un servidor Unix y la variable de sistema **lower_case_table_names** tiene valor 0.

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

[mysqld]
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 CFDI40

La clase [CFDI40](#) es la clase principal para comenzar a emitir CFDIs, en este ejemplo se le pasa como primer argumento la cadena de conexión a la base de datos CatálogoSAT, el resto de los argumentos son los atributos que poseerá el CFDI.

```
01 cfdi40 = new
02 CFDI40("Server=localhost;Port=3306;Database=catalogossat;Uid=root;Pwd=xxx;", //connString
03      "BZNPLUS", //pSerie
04      "196", //pFolio
05      fechaCfdi, //pFecha
06      "99", //pFormaPago
07      "", //pCondPago
08      2750.00m, //pSubtotal
09      0.00m, //pDescuento
10      "MXN", //pMoneda
11      1m, //pTipoCambio
12      3190.00m, //pTotal
13      "I", //pTipoCompr
14      "01", //pExportacion
15      "PPD", //pMetodoPago
16      "72010", //pLugarExped
17      null, //pConfirmacion
18      this.rutaCert //rutaCert
19      );
20
```

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/4"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xmlns:iedu="http://www.sat.gob.mx/iedu"
5   xsi:schemaLocation="http://www.sat.gob.mx/cfd/4 http://www.sat.gob.mx/sitio internet
6   Version="4.0"
7   Serie="BZNPLUS"
8   Folio="196"
9   Fecha="2023-05-14T17:00:17"
10  Sello="@@"
11  FormaPago="99"
12  NoCertificado="30001000000400002463"
13  Certificado="MIIFwTCCA6mgAwIBAgIUMzAwMDEwMDAwMDAwMDAwMDI0NjMwDQYJKoZIhvcNAQELBQAwggE
14  SubTotal="2750.00"
15  Descuento="0.00"
16  Moneda="MXN"
17  TipoCambio="1"
18  Total="3190.00"
19  TipoDeComprobante="I"
20  Exportacion="01"
21  MetodoPago="PPD"
22  LugarExpedicion="72010">
```

Imagen 33 Atributos de CFDI agregados



4.2 Establecer información del nodo InformacionGlobal

Utilizando la instancia de CFDI40 anterior invocamos el método "setInformacionGlobal()" y le pasamos la información correspondiente al nodo InformacionGlobal.

```
01 //Estableciendo el nodo InformacionGlobal
02 cfdi40.setInformacionGlobal("04", //pPeriodicidad
03 "05", //pMeses
04 2021); //pAño
```

Lo cual produce el nodo InformacionGlobal:

```
23 <cfdi:InformacionGlobal Periodicidad="04"
24 Meses="05"
25 Año="2021"/>
```

Imagen 34 Nodo InformacionGlobal agregado

4.3 Agregar CfdiRelacionados

Utilizando la instancia de [CFDI40](#) anterior invocamos el método "addCfdirelacionados()" y le pasamos el tipo de relación y los UUIDs de los CFDIs Relacionados, lo cual genera un primer nodo CfdiRelacionados, con los UUIDs indicados.

```
01 //Estableciendo el nodo CfdiRelacionados
02 cfdi40.addCfdirelacionados("04", //pTipoRelacion
03 "DB4A59FB-5BD5-4744-87B6-0F50E7DE2951", //pUUID
04 "806EA594-A850-48C3-B56E-1EB0C4D570C2"); //pUUID
```

Lo cual produce el nodo CfdiRelacionados y los nodos CfdiRelacionado correspondientes a cada uno de los UUIDs indicados en el método anterior (ya que si existe CfdiRelacionados debe existir al menos un CfdiRelacionado):

```
26 <cfdi:CfdiRelacionados TipoRelacion="04">
27 <cfdi:CfdiRelacionado UUID="DB4A59FB-5BD5-4744-87B6-0F50E7DE2951"/>
28 <cfdi:CfdiRelacionado UUID="806EA594-A850-48C3-B56E-1EB0C4D570C2"/>
29 </cfdi:CfdiRelacionados>
```

Imagen 35 Nodo CfdiRelacionados agregado

En caso de requerir agregar más CFDIs relacionados con distinto valor para TipoRelacion, se deberá utilizar el mismo método tantas veces sea necesario.

4.4 Establecer información del Emisor

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

```
01 // Estableciendo el Emisor
02 cfdi40.setEmisor(this.rfcEmisor, //pRfc
03 "HERRERIA & ELECTRICOS SA DE CV", //pNombre
04 "626"); //pRegFiscal
05
```

El nodo agregado es Emisor:


```

34 <cfdi:Emisor Rfc="H&amp;E951128469"
35
36 Nombre="HERRERIA &amp; ELECTRICOS SA DE CV"
37 RegimenFiscal="626"/>

```

Imagen 36 Nodo Emisor agregado

4.5 Establecer información del Receptor

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

```

01 // Estableciendo el Receptor
02 cfdi40.setReceptor(this.rfcReceptor, //pRfc
03 "TRANSPORTADORA CER-MA DE AHUAZOTEPEC", //pNombre
04 "73180", //pDomicilioFiscalReceptor
05 null, //pResidenciaFiscal
06 null, //pNumRegIdTrib
07 "626", //pRegimenFiscalReceptor
"501"); //pUsoCfdi

```

El nodo agregado es Receptor, el cual tiene los atributos con los valores que se les paso al método `setReceptor()`:

```

37 <cfdi:Receptor Rfc="TCM970625MB1"
38
39 Nombre="TRANSPORTADORA CER-MA DE AHUAZOTEPEC"
40 DomicilioFiscalReceptor="73180"
41 RegimenFiscalReceptor="626"
42 UsoCFDI="S01"/>

```

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 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 = cfdi40.addConcepto("78101800", //pCveProdServ
02 "", //pNoIdentif
03 1m, //pCantidad
04 "E48", //pCveUnidad
05 "F", //pUnidad
06 "VIAJE DE PARAFINA DEL 26 DE NOVIEMBRE DE 2021 CON 17,000
07 KG DE PESO", //pDescripcion
08 2750.00m, //pValorUnit
09 2750.00m, //pImporte
10 0.00m, //pDescuento
"02"); //pObjetoImp

```

El nodo agregado es concepto con los atributos que se le paso al método:

```

43 <cfdi:Concepto ClaveProdServ="78101800"
44 Cantidad="1"
45 ClaveUnidad="E48"
46 Unidad="F"
47 Descripcion="VIAJE DE PARAFINA DEL 26 DE NOVIEMBRE DE 2021 CON 17,000 KG DE PESO"
48 ValorUnitario="2750.00"
49 Importe="2750.00"
50 Descuento="0.00"
51 ObjetoImp="02">

```

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 Cfdi40.addInformacionAduaneraToConcepto(indexConc, //pIndexConc seria= 1
03                                     "15 43 3740 6000564" //pNumPedimento
04 );
05
06
```

Modifica el nodo concepto apuntado por `@indexConc` agregándole como hijo el nodo información Aduanera:

```
72 <cfdi:InformacionAduanera NumeroPedimento="15 43 3740 6000564"/>
```

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 traslados por medio del método `addImpTrasladoToConcepto()`. El concepto se indica con el parámetro `@indexConc`.

```
01 // agregando impuesto traslado al concepto
02 cfdi40.addImpTrasladoToConcepto(indexConc, //pIndexConc
03                               2750.00m, //pBase
04                               "002", //pImpuesto
05                               "Tasa", //pTipoFactor
06                               0.160000m, //pTasaOCuota
07                               440.00m); //pImporte
```

Agrega el traslado al concepto:

```
52 <cfdi:Impuestos>
53 <cfdi:Traslados>
54 <cfdi:Traslado Base="2750.00"
55 Impuesto="002"
56 TipoFactor="Tasa"
57 TasaOCuota="0.160000"
58 Importe="440.00"/>
59 </cfdi:Traslados>
60 </cfdi:Impuestos>
```

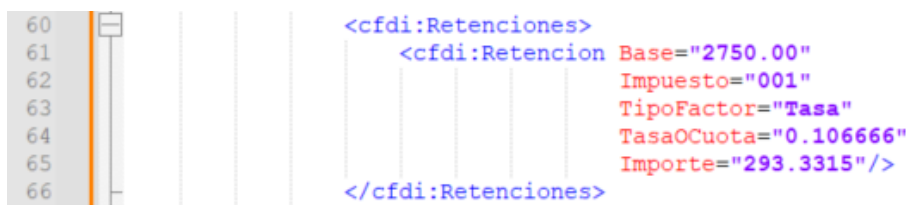
Imagen 40 Resultado de agregar un impuesto traslado al concepto

4.9 Agregar un impuesto retenido a un concepto

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

```
01 // agregando impuesto retenido al concepto
02 cfdi40.addImpRetencionToConcepto(indexConc, //pIndexConc
03                               2750.00m, //pBase
04                               "001", //pImpuesto
05                               "Tasa", //pTipoFactor
06                               0.106666m, //pTasaOCuota
07                               293.3315m); //pImporte
```

Agrega la retención al concepto:



```

60 <cfdi:Retenciones>
61 <cfdi:Retencion Base="2750.00"
62 Impuesto="001"
63 TipoFactor="Tasa"
64 TasaOCuota="0.106666"
65 Importe="293.3315"/>
66 </cfdi:Retenciones>

```

Imagen 41 Resultado de agregar un impuesto traslado al concepto

4.10 Establecer información de A Cuenta de Terceros

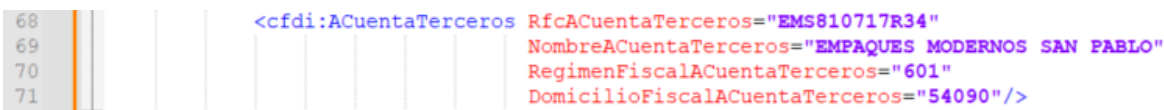
Cuando se ha definido un concepto al CFDI, es posible indicar la información correspondiente al nodo A Cuenta de Terceros por medio del método `setACuentaTercerosToConcepto()`. El concepto se indica con el parámetro `@indexConc`.

```

01 // agregando información ACuentaTerceros al concepto
02 cfdi40.setACuentaTercerosToConcepto(indexConc, //pIndexConc
03 "EMS810717R34", //pRfcACuentaTerceros
04 "EMPAQUES MODERNOS SAN PABLO", //pNombreACuentaTerceros
05 "601", //pRegimenFiscalACuentaTerceros
06 "54090"); //pDomicilioFiscalACuentaTerceros

```

Agrega el nodo `ACuentaTerceros` al concepto:



```

68 <cfdi:ACuentaTerceros RfcACuentaTerceros="EMS810717R34"
69 NombreACuentaTerceros="EMPAQUES MODERNOS SAN PABLO"
70 RegimenFiscalACuentaTerceros="601"
71 DomicilioFiscalACuentaTerceros="54090"/>

```

Imagen 42 Resultado de agregar un impuesto traslado al concepto

4.11 Agregar Cuenta Predial al concepto

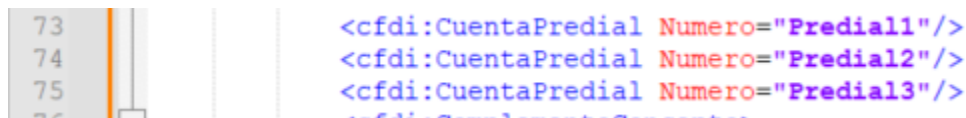
Cuando se ha definido un concepto al CFDI, es posible indicar la información correspondiente a la Cuenta Predial por medio del método `addCuentaPredialToConcepto()`. El concepto se indica con el parámetro `@indexConc`.

```

01 // agregando información Cuenta Predial al concepto
02 cfdi40.addCuentaPredialToConcepto(indexConc, //pIndexConc
03 "Predial1", //pLsNumeros
04 "Predial2",
05 "Predial3");
06

```

Agrega los nodos `CuentaPredial` al concepto:



```

73 <cfdi:CuentaPredial Numero="Predial1"/>
74 <cfdi:CuentaPredial Numero="Predial2"/>
75 <cfdi:CuentaPredial Numero="Predial3"/>

```

Imagen 43 Resultado de agregar un impuesto traslado al concepto

4.12 Establecer información del total impuestos de comprobante

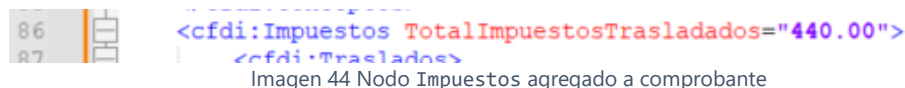
Definiendo el total impuestos con el método `setTotalImpuestos()`.

```

01 // estableciendo el total de impuestos
02 cfdi40.setTotalImpuestos(null, //pTotImpRetenidos
03 440.00m); //pTotImpTrasladados
04
05

```

Lo cual agrega el nodo Impuestos:



```

86 <cfdi:Impuestos TotalImpuestosTrasladados="440.00">
87   <cfdi:Traslados>

```

Imagen 44 Nodo Impuestos agregado a comprobante

4.13 Agregar un impuesto traslado a impuestos de comprobante

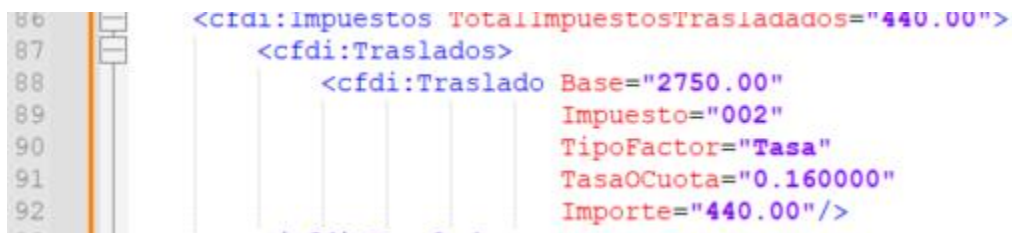
Solo si se ha definido el nodo Impuestos al comprobante (Xpath: `cfdi:Comprobante/cfdi:Impuestos`), es posible indicar el impuesto Traslado por medio del método `addImpTraslado()`.

```

01 //estableciendo un impuesto traslado
02 cfdi40.addImpTraslado(2750.00m, //pBase
03 "002", //pImpuesto
04 "Tasa", //pTipoFactor
05 0.160000m, //pTasaCuota
440.00m); //pImporte

```

Este agrega los nodos traslados y traslado al nodo "impuestos"



```

86 <cfdi:Impuestos TotalImpuestosTrasladados="440.00">
87   <cfdi:Traslados>
88     <cfdi:Traslado Base="2750.00"
89       Impuesto="002"
90       TipoFactor="Tasa"
91       TasaOCuota="0.160000"
92       Importe="440.00"/>

```

Imagen 45 Impuesto traslado agregado

4.14 Autorización para el uso de Sifei Signature

Antes de utilizar los métodos para generar los complementos, generar el XML del comprobante CFDI 4.0 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:

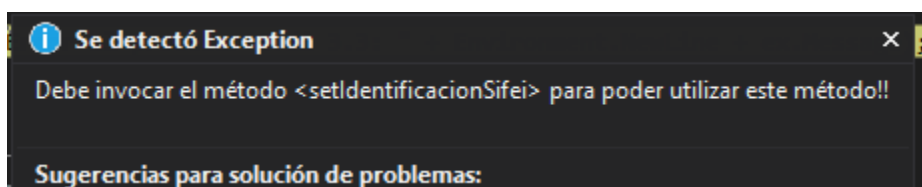


Imagen 46 Autenticandose para el uso de Sifei Signature

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

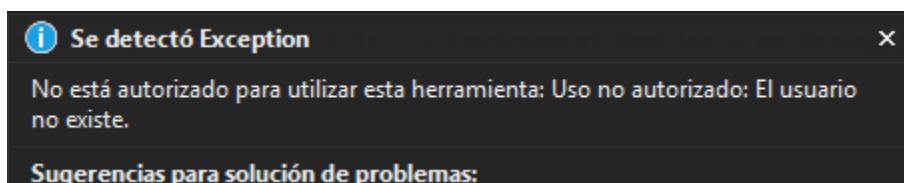


Imagen 47 Error de autenticación

4.15 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 = cfdi40.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/4"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:iedu="http://www.sat.gob.mx/iedu"
  xsi:schemaLocation="http://www.sat.gob.mx/cfd/4
http://www.sat.gob.mx/sitio_internet/cfd/4/cfdv40.xsd http://www.sat.gob.mx/iedu
http://www.sat.gob.mx/sitio_internet/cfd/iedu/iedu.xsd"
  Version="4.0"
  Serie="BZNPLUS"
  Folio="196"
  Fecha="2023-05-15T11:28:26"
  Sello="@ "
  FormaPago="99"
  NoCertificado="30001000000400002463"
  Certificado="MIIFwTCCA6mgAwIBAgIUMzA..."
  SubTotal="2750.00"
  Descuento="0.00"
  Moneda="MXN"
  TipoCambio="1"
  Total="3190.00"
  TipoDeComprobante="I"
  Exportacion="01"
```

```

    MetodoPago="PPD"
    LugarExpedicion="72010">
<cfdi:InformacionGlobal Periodicidad="04"
    Meses="05"
    Año="2021"/>
<cfdi:CfdiRelacionados TipoRelacion="04">
    <cfdi:CfdiRelacionado UUID="DB4A59FB-5BD5-4744-87B6-0F50E7DE2951"/>
    <cfdi:CfdiRelacionado UUID="806EA594-A850-48C3-B56E-1EB0C4D570C2"/>
</cfdi:CfdiRelacionados>
<cfdi:CfdiRelacionados TipoRelacion="01">
    <cfdi:CfdiRelacionado UUID="DB4A59FB-5BD5-4744-87B6-0F50E7DE2953"/>
    <cfdi:CfdiRelacionado UUID="806EA594-A850-48C3-B56E-1EB0C4D570C4"/>
</cfdi:CfdiRelacionados>
<cfdi:Emisor Rfc="H&E951128469"
    Nombre="HERRERIA & ELECTRICOS SA DE CV"
    RegimenFiscal="626"/>
<cfdi:Receptor Rfc="TCM970625MB1"
    Nombre="TRANSPORTADORA CER-MA DE AHUAZOTEPEC"
    DomicilioFiscalReceptor="73180"
    RegimenFiscalReceptor="626"
    UsoCFDI="S01"/>
<cfdi:Conceptos>
    <cfdi:Concepto ClaveProdServ="78101800"
    Cantidad="1"
    ClaveUnidad="E48"
    Unidad="F"
    Descripcion="VIAJE DE PARAFINA DEL 26 DE NOVIEMBRE DE
2021 CON 17,000 KG DE PESO"
    ValorUnitario="2750.00"
    Importe="2750.00"
    Descuento="0.00"
    ObjetoImp="02">
    <cfdi:Impuestos>
    <cfdi:Traslados>
    <cfdi:Traslado Base="2750.00"
    Impuesto="002"
    TipoFactor="Tasa"
    TasaOCuota="0.160000"
    Importe="440.00"/>
    </cfdi:Traslados>
    <cfdi:Retenciones>
    <cfdi:Retencion Base="2750.00"
    Impuesto="001"
    TipoFactor="Tasa"
    TasaOCuota="0.106666"
    Importe="293.3315"/>
    </cfdi:Retenciones>
    </cfdi:Impuestos>
    <cfdi:ACuentaTerceros RfcACuentaTerceros="EMS810717R34"
    NombreACuentaTerceros="EMPAQUES MODERNOS
SAN PABLO"
    RegimenFiscalACuentaTerceros="601"
    DomicilioFiscalACuentaTerceros="54090"/>
<cfdi:InformacionAduanera NumeroPedimento="15 43 3740
6000564"/>
<cfdi:CuentaPredial Numero="Predial1"/>
<cfdi:CuentaPredial Numero="Predial2"/>
<cfdi:CuentaPredial Numero="Predial3"/>
<cfdi:ComplementoConcepto>
    <iedu:instEducativas version="1.0"
    nombreAlumno="SOLIS HERRERA VALERIA"
    CURP="SOHV031025MHGLRLA4"
    nivelEducativo="Primaria"
  
```

```

autRVOE="13PPR0034X"
rfcPago="AALE5706229F6"/>
    </cfdi:ComplementoConcepto>
  </cfdi:Concepto>
</cfdi:Conceptos>
<cfdi:Impuestos TotalImpuestosTrasladados="440.00">
  <cfdi:Traslados>
    <cfdi:Traslado Base="2750.00"
      Impuesto="002"
      TipoFactor="Tasa"
      TasaOCuota="0.160000"
      Importe="440.00"/>
  </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 4.0](#).

4.16 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 = cfdi40.sellaComprobante(rtnCfdi.XML,
03  @"C:\Workspace\DatosBuzonMariaDB\Cert\DEM020170725123153.cer",
04  @"C:\Workspace\DatosBuzonMariaDB\Cert\DEM020170725123153.key",
05  "12345678a",
06  false,
07  true);
08
09  //comprobamos que se halla sellado
10  if (rtnCfdi.EstaSellado) {
11    //escribimos el Comprobante sellado:
12
13    rtnCfdi.XML.Save(@"C:\Workspace\SifeiSignature\CfdiGeneradosTest\cfdi_sellado.xml");
14  }
15

```

Comprobante sellado atributo sello (línea 10, marcado en púrpura):



```

1  <?xml version="1.0" encoding="utf-8"?>
2  <cfdi:Comprobante xmlns:cfdi="http://www.sat.gob.mx/cfd/4"
3      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4      xmlns:iedu="http://www.sat.gob.mx/iedu"
5      xsi:schemaLocation="http://www.sat.gob.mx/cfd/4 http://www.sat.gob.mx/sitio_internet/cfd/4/cfdv
6      Version="4.0"
7      Serie="BZNPLUS"
8      Folio="196"
9      Fecha="2023-05-15T11:56:35"
10     Sello="4fv1SkeLGEW9FjmwfGMm/+UR4vkTtvOM/5qGOXja2RND6+6SQzUX7XkK0dCHiisCsgW7vg5jFq9cHARnL8Retq6
11     FormaPago="99"
12     NoCertificado="30001000000400002463"
13     Certificado="MIIFwTCCA6mgAwIBAgIUMzAwMDEwMDAwMDA0NDI0NjMwDQYJKoZIhvcNAQELBQAwggErMQ8wDQYDVQ
14     SubTotal="2750.00"
15     Descuento="0.00"
16     Moneda="MXN"
17     TipoCambio="1"
18     Total="3190.00"
19     TipoDeComprobante="I"
20     Exportacion="01"
21     MetodoPago="PPD"
22     LugarExpedicion="72010">
23     <cfdi:CfdiRelacionados TipoRelacion="04">
24         <cfdi:CfdiRelacionado UUID="DB4A59FB-5BD5-4744-87B6-0F50E7DE2951"/>
25         <cfdi:CfdiRelacionado UUID="806EA594-A850-48C3-B56E-1EB0C4D570C2"/>
26     </cfdi:CfdiRelacionados>
27     <cfdi:CfdiRelacionados TipoRelacion="01">
28         <cfdi:CfdiRelacionado UUID="DB4A59FB-5BD5-4744-87B6-0F50E7DE2953"/>
29         <cfdi:CfdiRelacionado UUID="806EA594-A850-48C3-B56E-1EB0C4D570C4"/>
30     </cfdi:CfdiRelacionados>
31     <cfdi:Emisor Rfc="H&E951128469"
32         Nombre="HERRERIA &ELECTRICOS SA DE CV"
33         RegimenFiscal="626"/>

```

Imagen 48 Comprobante Sellado

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

4.17 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 = cfdi40.timbraComprobante("A",
03 "VPjftCMG1beV+/wkNwc8lkaFetxTrC/rt20oZgbn130S6SD6kPAmLozt7CMkqzZK3L7+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 `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 CFDI contendrá el nodo **TimbreFiscalDigital** (líneas 78 a 86):

```

70         <cfdi:Traslado Base="2750.00"
71             Impuesto="002"
72             TipoFactor="Tasa"
73             TasaOCuota="0.160000"
74             Importe="440.00"/>
75     </cfdi:Traslados>
76 </cfdi:Impuestos>
77 <cfdi:Complemento>
78     <tfd:TimbreFiscalDigital xmlns:tfd="http://www.sat.gob.mx/TimbreFiscalDigital"
79         xsi:schemaLocation="http://www.sat.gob.mx/TimbreFiscalDigital http://www.sat.gob.mx/sitio
80             Version="1.1"
81             UUID="8D13C013-C6CA-45D1-849C-551F2CD1D201"
82             FechaTimbrado="2023-05-16T12:00:38"
83             RfcProvCertif="SPR190613I52"
84             SelloCFD="4fv1sKeLGEW9FjmwfGMM/+Ur4vkTtvOM/5qGOXja2RND6+6SQzUX7XkK0dChiisCsgW7vgSjFq9cHARr
85             NoCertificadoSAT="30001000000400002495"
86             SelloSAT="cpUSGWYFGClOuVLSrDhB71EHIFWhlI2vzRvzdEofr1/z6Jke9EbMyEpfwsboN1ZF5GJKQzPulbe5G2s
87     </cfdi:Complemento>
88 </cfdi:Comprobante>

```

Imagen 49 Comprobante timbrado

4.18 Recuperar timbre

```
01 //creamos el XmlDocument y cargamos el comprobante
02 System.Xml.XmlDocument cfdi = new System.Xml.XmlDocument();
03 cfdi.Load(@"C:\Workspace\comprobante.xml");
04 ReturnTimbre recuperado = CFDI40.recuperaTimbre(
05     "JGE080430LN8", //usuario
06     "0e84ec92", //password
07     cfdi, //objeto XmlDocument
08     CFDI33.tipoTimbrado.Pruebas
09 );
10 //verificamos si esta timbrado (si existe un timbre)
11 if (recuperado.IsTimbrado) {
12     //escribimos el comprobante
13     recuperado.Xml.Save(@"C:\Workspace\hash_cfdi_timbrado.xml");
14
15     // escribimos el código de barra bidimensional en un archivo jpg
16     ByteArrayToFile(@"C:\Workspace\hash_cbb.jpg", recuperado.Cbb);
17     Console.WriteLine(string.Format("{0} - Proceso finalizado", DateTime.Now));
18 }
19 }
```

5. Crear Complementos

Con la clase `Complemento` podemos crear varios tipos de complemento.

```
01 //Instanciamos la clase complemento, pasándole el string de conexión.
02 Complemento complCfdi = new
03 Complemento("Server=localhost;Port=3306;Database=catálogosat;Uid=root;Pwd=123456;",
04 fechaCfdi);
05
```

5.1 Complemento Comercio Exterior

5.1.1 Definir complemento Comercio Exterior

```
01 //=====>>>>>>> Inclusión de complemento CCE1.1
02 complCfdi.Cce11_Define_ComercioExterior(
03     "",                //pMotivoTrasl
04     "2",               //pTipoOperacion
05     "A1",              //pClavePedim
06     0,                 //pCertifOrigen
```



```

07         null,           //pNumCertifOrigen
08         null,           //pNumExportadorConf
09         "CIF",          //pIncoterm
10         0,              //pSubdivision
11         null,           //pObservaciones
12         17.473200m,     //pTipoCambioUSD
13         242.11m);       //pTotalUSD

```

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     "7103910100",      //pFraccionAranc
04     null,               //pCantidadAduana
05     null,               //pUnidadAduana
06     null,               //pValorUnitAduana
07     242.11m);          //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         cfdi40.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
07         cfdi40.addNamespace(complCfdi.NsComercioExterior1_1);
08     }
09     // ahora para generar el comprobante, sellar y timbrar ver la clase CFDI40
10

```

Para generar el comprobante, sellar y timbrar ver la clase [CFDI40](#)

5.2 Complemento Comercio Exterior 2.0

5.2.1 Definir complemento Comercio Exterior

```
01 //=====>>>>>>> Inclusión de complemento CCE2.0
02 complCfdi.CCE20_Define_ComercioExterior(
03     "",                //pMotivoTrasl
04     "A1",              //pClavePedim
05     0,                 //pCertifOrigen
06     null,              //pNumCertifOrigen
07     null,              //pNumExportadorConf
08     "CIF",             //pIncoterm
09     null,              //pObservaciones
10     17.133500m,        //pTipoCambioUSD
11     242.11m);          //pTotalUSD
```

5.2.2 Establecer Emisor de Comercio Exterior

```
01 complCfdi.CCE20_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", //pDomPaís  
12     "94300"); //pDomCP
```

5.2.3 Definir Receptor de Comercio Exterior

```
01 complCfdi.CCE20_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
```

5.2.4 Agregar mercancía de Comercio Exterior

```
01 indexMerc = complCfdi.CCE20_addMercancia(  
02     "A-RTGT-56",           //pNoIdentificacion  
03     "7103910100",         //pFraccionAranc  
04     null,                  //pCantidadAduana  
05     null,                  //pUnidadAduana  
06     null,                  //pValorUnitAduana  
07     242.1100m);           //pValorDolares
```

5.2.5 Generar el Complemento Comercio Exterior

```

01  /// generamos el complemento
02  ReturnXmlComplemento rtnXmlCompl = complCfdi.CCE20_generaComplemento();
03  ///verificamos que se haya creado.
04  if (rtnXmlCompl.IsXmlComplementoCreated) {
05  // le pasamos a al objeto CFDI40 el complemento y el namespace de comercio exterior
06      cfdi40.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
07      cfdi40.addNamespace(complCfdi.NsComercioExterior2_0);
08  }
09  // ahora para generar el comprobante, sellar y timbrar ver la clase CFDI40
10

```

Para generar el comprobante, sellar y timbrar ver la clase [CFDI40](#)

5.3 Complemento Nómina

5.3.1 Definir complemento Nómina

```

01  //complCfdi es de tipo Complemento
02  complCfdi.Nomina12_DefineNomina("E", //pTipoNomina
03      new DateTime(2022, 12, 23), //pFechaPago
04      new DateTime(2022, 01, 26), //fechaInicPago
05      new DateTime(2022, 02, 20), //fechaFinPago
06      7, //numDiasPag
07      16744.39m, //totPercepciones
08      296.79m, //totDeducciones
09      10m); //totOtrosPagos
10
11

```

5.3.2 Agregar un emisor a Nómina

```

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

```

5.3.3 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, //esSindicalizado
09      "01", //pTipoJornada
10      "02", //pTipoRegimen
11      "49", //pNumEmpleado
12      "1", //pDepartamento
13      "ENC ALMACEN DE MAQUI", //pPuesto
14      "5", //pRiesgo
15      "99", //pPeriodiciadPago
16      "072", //pBanco
17      null, //pCuentaBanco
18      210.00m, //pSalarioBaseCotApor
19      231.52m, //pSalarioDiario
20      "SIN"); //pEstado
21

```

5.3.4 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

```

5.3.5 Agregar una percepción a percepciones de Nómina

```

01      complCfdi.Nomina12_addPercepcion(
02          "029",              //pTipoPercepcion
03          "500100010004",    //pClave
04          "Vales de Despensa", //pConcepto
05          0.00m,              //pImporteGravado
06          462.00m);          //pImporteExento
07

```

5.3.6 Agregar una Separación indemnización a percepciones de Nómina

```

01      complCfdi.Nomina12_setSeparacionIndemnizacion(
02          70160.94m,          //pTotalPagado
03          3,                  //pNumAñosServicio
04          6384m,              //pUltimoSueldoMensOrd
05          6384.00m,          //pIngresoAcumulable
06          63776.94m);        //pIngresoNoAcumulable
07

```

5.3.7 Establecer deducciones a Nómina

```

01      complCfdi.Nomina12_setDeducciones(
02          null,                //pTotalOtrasDeduc
03          3781.05m);          //pTotalImpuestosReten

```

5.3.8 Agregar una deducción a deducciones de Nómina

```

01      complCfdi.Nomina12_addDeducion(
02          "002",                //pTipoDeduc
03          "20010004",          //pClave
04          "ISR",                //pConcepto
05          3781.05m);            //pImporte
06

```

5.3.9 Generar el Complemento Nómina

Una vez agregado al objeto CFDI40 ya se puede generar el XML, sellar y timbrar con los respectivos métodos de esta clase (CFDI40).

```

01      ReturnXmlComplemento rtnXmlCompl = complCfdi.Nomina12_generaComplemento();
02          //verificamos que se haya creado el complemento y si es así lo agregamos
03      al comprobante, tambien debemos agregar el namespace para nómina
04          if (rtnXmlCompl.IsXmlComplementoCreated) {
05              cfdi40.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
06              cfdi40.addNamespace(complCfdi.NsNomina12);
07          }

```

5.4 Complemento Detallista

5.4.1 Definir complemento Detallista

```
01 //=====>>>>>>> Inclusión de complemento Detallista
02 complCfdi.Det131_Define_Detallista(detallistaDocumentStatus.ORIGINAL);
03 //DocumentStatus
04
```

5.4.2 Establecer el requestForPaymentIdentification a Detallista

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

5.4.3 Agregar un specialInstruction a Detallista

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

5.4.4 Establecer el orderIdentification a Detallista

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

5.4.5 Agregar un referenceIdentification al AdditionalInformation de Detallista

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

5.4.6 Establecer un DeliveryNote a Detallista

```
01 complCfdi.Det131_SetDeliveryNote(
02         new DateTime(2016, 04, 07), // reference date
03         "0000001" //referenceIdIndetification
04     );
```

5.4.7 Definir el *Buyer* (comprador) de Detallista

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

5.4.8 Definir el *Seller* (vendedor) a Detallista

```
01 complCfdi.Det131_SetSeller(  
02     "7244020982223", //g1  
03     detallistaSellerAlternatePartyIdentificationType.IEPS REFERENCE,
```

```
04      "12382317" //alternatePartyIdentification
05      );
06
```

5.4.9 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 );
```

5.4.10 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 );
```

5.4.11 Agregar un Customs a Detallista

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

5.4.12 Agregar un currency a Detallista

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

5.4.13 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
```



5.4.14 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 );
```

5.4.15 Definir el totalAmount a Detallista

01	
02	complCfdi.Det131_setTotalAmount(1.0m);
03	

5.4.16 Definir el TotalAllowanceCharge a Detallista para definir el monto total de cargos o descuentos

```
01  
02 complCfdi.Det131_AddTotalAllowanceCharge(detallistaTotalAllowanceChargeAllowanceOrCharge  
03 eType.CHARGE,  
04         detallistaTotalAllowanceChargeSpecialServicesType.ADS, 1.00m);  
05  
06
```

5.4.17 Generar el complemento Detallista

```
01 //generamos el complemento detallista
02     ReturnXmlComplemento rtnXmlCompl =
03     complCfdi.Det131_generaComplemento();
04     //verificamos que se creo
05     if (rtnXmlCompl.IsXmlComplementoCreated) {
06         //le pasamos el complemento generado a CFDI40 y agregamos su
07     correspondiente Namespace
08         cfdi40.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
09         cfdi40.addNamespace(complCfdi.NsDetallista131);
10     }
11
```

5.5 Complemento Recepción de Pagos 2.0

5.5.1 Definir el nodo Totales

[illegible]

5.5.2 Agregar un Pago

```

01 complCfdi.PAGOS20_ADD_Pagos_Pago(
02     new DateTime(2020, 12, 15, 12, 0, 0), //pFechaPago
03     "03", //pFormaDePago
04     "MXN", //pMoneda
05     1m, //pTipoCambio
06     49153.09m, //pMonto
07     null, //pNumOperacion
08     "BSM970519DU8", //pRfcEmisorCtaOrd
09     "BANCO SANTANDER MEXICO SA", //pNomBancoOrdExt
10     "014628655019049888", //pCtaOrdenante
11     "BSM970519DU8", //pRfcEmisorCtaBen
12     "014650655057442596", //pCtaBeneficiario
13     null, //pTipoCadPago
14     null, //pCertPago
15     null, //pCadPago
16     null); //pSelloPago

```

5.5.3 Agregar un DoctoRelacionado al último Pago agregado

```

01 complCfdi.PAGOS20_ADD_Pagos_Pago_DoctoRelacionado_UltimoPago(
02     "6AD9904B-4954-48C7-980C-FFAF9B09642B", //pIdDocumento
03     "A", //pSerie
04     "19225", //pFolio
05     "MXN", //pMonedaDR
06     1m, //pEquivalenciaDR
07     1, //pNumParcialidad
08     47703.41m, //pImpSaldoAnt
09     47703.41m, //pImpPagado
10     0.00m, //pImpSaldoInsoluto
11     "02"); //pObjetoImpDR

```

5.5.4 Agregar un Impuesto TrasladoDR / RetencionDR al último DoctoRelacionado agregado

```

01 complCfdi.PAGOS20_ADD_Pagos_Pago_DoctoRelacionado_TrasladoDR_UltimoDoctoRelacionado
02 (
03     1m, //pBaseDR
04     "002", //pImpuestoDR
05     "Tasa", //pTipoFactorDR
06     0.160000m, //pTasaOCuotaDR
07     0.16m); //pImporteDR
08

```

El código anterior agregará un nodo TrasladoDR al último nodo DoctoRelacionado. Para agregar un nodo RetencionDR, se debe usar el método PAGOS20_ADD_Pagos_Pago_DoctoRelacionado_RetencionDR_UltimoDoctoRelacionado con los mismos parámetros que se utilizan para los Traslados.



5.5.5 Agregar un Impuesto TrasladoP al ultimo Pago agregado.

```
01 complCfdi.PAGOS20_ADD_Pagos_Pago_TrasladoP_UltimoPago(1m, //pBaseP
02 "002", //pImpuestoP
03 "Tasa", //pTipoFactorP
04 0.160000m, //pTasaOCuotaP
05 0.16m); //pImporteP
06
07
```

5.5.6 Agregar un Impuesto RetencionP al ultimo Pago agregado.

```
01 complCfdi.PAGOS20_ADD_Pagos_Pago_RetencionP_UltimoPago("001", //pImpuestoP
02 0.16m); //ImporteP
03
```

5.5.7 Generar el Complemento Pago

```
01 //generamos el complemento pago
02 ReturnXmlComplemento rtnXmlCompl = complCfdi.Pagos20_generaComplemento();
03 //verificamos que se haya creado.
04 if (rtnXmlCompl.IsXmlComplementoCreated) {
05     // agregamos el complemento al comprobante
06     cfdi40.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
07     //agregamos el namespace del complemento pago al comprobante
08     cfdi40.addNamespace(complCfdi.NsPagos20);
09 }
10
11
```

5.6 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 VALENTINA", //nombre del alumno
05     "SOHV031025MHGLRLA0", //CURP
06     instEducativasNivelEducativo.Primaria,
07     "13PPR0034A", //autRVOE
08     "AALE5706229F0" //rfcPago
09 );
10 //comprobamos que se creó el complemento
11 if (rtnXmlCompl.IsXmlComplementoCreated) {
12     //agregamos el complemento al concepto indicado por @indexConc
13     cfdi40.addComplementoConcepto(indexConc, rtnXmlCompl.XMLComplemento);
14     //agregamos el namespace de Iedu al comprobante
15     cfdi40.addNamespace(complCfdi.NsIedu10);
16 }
```



5.7 Definir y generar el complemento Hidrocarburos y Petrolíferos 1.0 (a un concepto)

```

01 //debido a que el complemento Hidrocarburos y Petrolíferos es un solo nodo, el método
02 HidroYPetro10_generaComplemento define y genera el complemento a la vez.
03 ReturnXmlComplemento rtnXmlCompl = complCfdi.HidroYPetro10_generaComplemento (
04     "PER03", //TipoPermiso
05     "PL/364/DIS/OM/2015", //NumeroPermiso
06     "15101505", //ClaveHYP
07     "SP18" //SubProductoHYP
08 );
09 //comprobamos que se creó el complemento
10 if (rtnXmlCompl.IsXmlComplementoCreated) {
11     //agregamos el complemento al concepto indicado por @indexConc
12     cfdi40.addComplementoConcepto(indexConc, rtnXmlCompl.XMLComplemento);
13     //agregamos el namespace de Iedu al comprobante
14     cfdi40.addNamespace(complCfdi.NsHYP10);
15 }
16

```

5.8 Definir y generar el complemento Donatarias 1.1

```

01 //debido a que el complemento IEDU es un solo nodo, el método Donat11_generaComplemento
02 define y genera el complemento a la vez.
03 string leyenda = "Este comprobante ampara un donativo...";
04 ReturnXmlComplemento rtnXmlCompl = complCfdi.Donat11_generaComplemento("325-SAT-28-
05 I-(21)-4109", new DateTime(2017, 1, 9), leyenda);
06 //comprobamos que se creó el complemento
07 if (rtnXmlCompl.IsXmlComplementoCreated) {
08     //agregamos el complemento al comprobante
09     cfdi40.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
10     //agregamos el namespace de Iedu al comprobante
11     cfdi40.addNamespace(complCfdi.NsDonat11);
12 }
13
14

```

5.9 Complemento Carta Porte 2.0

5.9.1 Definir el complemento Carta Porte

```

01 complCfdi.CP20_SET_CartaPorte("No", //TranspInternac
02     null, //EntradaSalidaMerc
03     null, //PaisOrigenDestino
04     null, //ViaEntradaSalida
05     52.61m); //TotalDistRec

```

5.9.2 Agregar una Ubicación

```

01 int indx = complCfdi.CP20_ADD_CartaPorte_Ubicacion("Origen", //TipoUbicacion
02     "OR000001", //IDUbicacion
03     "PASF690721EL6", //RFCRemitenteDestinatario
04     "", //NombreRemitenteDestinatario
05     "", //NumRegIdTrib
06     "", //ResidenciaFiscal
07     "", //NumEstacion
08     "", //NombreEstacion
09     "", //NavegacionTrafico
10

```

```

11      new DateTime(2021, 12, 01, 19, 17, 52), //FechaHoraSalidallegada
12      "", //TipoEstacion
13      null); //DistanciaRecorrida

```

5.9.3 Establecer el Domicilio para una ubicación

```

01 complCfdi.CP20_SET_CartaPorte_Ubicacion_Domicilio(indx, //indexUbicacion
02     new CartaPorte10_Domicilio
03     {
04         Calle = "ORQUIDEAS 6320",
05         Colonia = "0491",
06         Localidad = "03",
07         Municipio = "014",
08         Estado = "CMX",
09         Pais = "MEX",
10         CodigoPostal = "03010"
11     });
12
13

```

5.9.4 Establecer información de Mercancías

```

01 complCfdi.CP20_SET_CartaPorte_Mercancias(17000m, //PesoBrutoTotal
02     "KGM", //UnidadPeso
03     null, //PesoNetoTotal
04     1, //NumTotalMercancias
05     null); //CargoPorTasacion
06
07

```

5.9.5 Agregar una Mercancía

```

01 complCfdi.CP20_ADD_CartaPorte_Mercancias_Mercancia(
02     "12181503", //BienesTransp
03     null, //ClaveSTCC
04     "BULTOS DE PARAFINA 50 KG", //Descripcion
05     340m, //Cantidad
06     "XGY", //ClaveUnidad
07     "", //Unidad
08     null, //Dimensiones
09     null, //MaterialPeligroso
10     null, //CveMaterialPeligroso
11     null, //Embalaje
12     null, //DescripEmbalaje
13     17000m, //PesoEnKg
14     null, //ValorMercancia
15     "", //Moneda
16     null, //FraccionArancelaria
17     null); //UUIDComercioExt
18

```



5.9.6 Agregar CantidadTransporta a la última Mercancía

```
01 complCfdi.CP20_ADD_CartaPorte_Mercancias_Mercancia_CantidadTransporta_UltimaMercanc
02 ia(
03     340m,          //Cantidad
04     "OR000001",    //IDOrigen
05     "DE000001",    //IDDestino
06     "");           //CvesTransporte
07
```

5.9.7 Establecer elemento Autotransporte para Mercancias

```
01 complCfdi.CP20_SET_CartaPorte_Mercancias_Autotransporte(
02     "TPAF01",      //PermSCT
03     "2141TAL31052011021001063", //NumPermisoSCT
04     "VL",          //ConfigVehicular
05     "465AU2",      //PlacaVM
06     2013);         //AnioModeloVM
07
```

5.9.8 Establecer información de Seguros para el Autotransporte

```
01 complCfdi.CP20_SET_CartaPorte_Mercancias_Autotransporte_Seguros(
02     "SEGUROS ATLAS S.A", //AseguraRespCivil
03     "P00-1-7-118240-18", //PolizaRespCivil
04     "", //AseguraMedAmbiente
05     "", //PolizaMedAmbiente
06     "", //AseguraCarga
07     "", //PolizaCarga
08     null); //PrimaSeguro
09
```

5.9.9 Agregar un Tipo de Figura para FiguraTransporte

```
01 complCfdi.CP20_ADD_CartaPorte_TiposFigura(
02     "01", //TipoFigura
03     "AAMI760614929", //RFCFigura
04     "PUE0036473", //NumLicencia
05     "", //NombreFigura
06     "", //NumRegIdTribFigura
07     null, //ResidenciaFiscalFigura
08     new CartaPorte10_Domicilio
09     {
10         Calle = "",
11         NumeroExterior = "",
12         NumeroInterior = "",
13         Colonia = "",
14         Localidad = "",
15         Referencia = "",
16         Municipio = "",
17         Estado = "CMX",
18         Pais = "MEX",
19        CodigoPostal = "01985"
20     });
21
```

5.9.10 Generar el Complemento Carta Porte 2.0

```
01 ReturnXmlComplemento rtnXmlCompl = complCfdi.CartaPorte20_generaComplemento();
02 //comprobamos que se creó el complemento
03 if (rtnXmlCompl.IsXmlComplementoCreated) {
04     //agregamos el complemento al comprobante
05     cfdi40.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
06     //agregamos el namespace de Impuestos Locales al comprobante
07     cfdi40.addNamespace(complCfdi.NsCartaPorte20);
08 }
```

5.10 Complemento Carta Porte 3.0

5.10.1 Definir el complemento Carta Porte

```
01 complCfdi.CP30_SET_CartaPorte("CCBCD94-870A-4332-A52A-A52AA52AA52A", //IdCCP
02     "Sí", //TranspInternac
03     "IMD", //RegimenAduanero
04     "Entrada", //EntradaSalidaMerc
05     "USA", //PaisOrigenDestino
06     "04", //ViaEntradaSalida
07     525.11m, //TotalDistRec
08     "Sí", //RegistroISTMO
09     "05", //UbicacionPoloOrigen
10     "02"); //UbicacionPoloDestino
```

5.10.2 Agregar una Ubicación

```
01 int indx = complCfdi.CP30_ADD_CartaPorte_Ubicacion("Origen", //TipoUbicacion
02     "OR458563", //IDUbicacion
03     rfcReceptor, //RFCRemitanteDestinatario
04     "Cori$, S. A. de C. V.", //NombreRemitanteDestinatario
05     "", //NumRegIdTrib
06     "", //ResidenciaFiscal
07     "R0000", //NumEstacion
08     "PIEDRAS NEGRAS", //NombreEstacion
09     "", //NavegacionTrafico
10     new DateTime(2021, 10, 30, 12, 00, 00), //FechaHoraSalidaLlegada
11     "01", //TipoEstacion
12     null); //DistanciaRecorrida
13
```

5.10.3 Establecer el Domicilio para una ubicación

```
1 complCfdi.CP30_SET_CartaPorte_Ubicacion_Domicilio(indx,
02     new CartaPorte10_Domicilio
03     {
04         Calle = "CARRETERA REMITENTE NACIONAL",
05         NumeroExterior = "234325",
06         NumeroInterior = "545",
07         Colonia = "6003",
08         Localidad = "10",
09         Referencia = "KM9 A UN COSTADO DE LA CARRETERA FEDERAL",
10         Municipio = "087",
11         Estado = "VER",
12         Pais = "MEX",
13     }
```

14	CodigoPostal = "91013"
15	});

5.10.4 Establecer información de Mercancías

01	complCfdi.CP30_SET_CartaPorte_Mercancias(30000m, //PesoBrutoTotal
02	"MC", //UnidadPeso
03	28.026M, //PesoNetoTotal
04	2, //NumTotalMercancias
05	null, //CargoPorTasacion
06	null); //LogisticaInversaRecoleccionDevolucion
07	

5.10.5 Agregar una Mercancía

01	complCfdi.CP30_ADD_CartaPorte_Mercancias_Mercancia(
02	"11121701", //BienesTransp
03	"2429320", //ClaveSTCC
04	"ASERRÍN", //Descripcion
05	15000m, //Cantidad
06	"XBX", //ClaveUnidad
07	"CAJA", //Unidad
08	null, //Dimensiones
09	null, //MaterialPeligroso
10	null, //CveMaterialPeligroso
11	null, //Embalaje
12	null, //DescripEmbalaje
13	"05", //SectorCOFEPRIS
14	"nombre común del ingrediente activo de los precursores, químicos de uso
15	dual, plaguicidas o fertilizantes que se trasladan a través de los distintos medios
16	de transporte.", //NombreIngredienteActivo
17	"", //NomQuimico
18	"", //DenominacionGenericaProd
	"", //DenominacionDistintivaProd
	"", //Fabricante
	null, //FechaCaducidad
	"", //LoteMedicamento
	"", //FormaFarmaceutica
	"", //CondicionesEspTransp
	"", //RegistroSanitarioFolioAutorizacion
	"", //PermisoImportacion
	"48565254685654862548", //FolioImpoVUCEM
	"", //NumCAS
	"", //RazonSocialEmpImp
	"9851316113231546565131321312329816513214", //NumRegSanPlagCOFEPRIS
	"Atributo condicional para registrar el país y nombre o razón social de
	quien produce o fabrica el ingrediente activo del plaguicida o fertilizante.",
	//DatosFabricante
	"DATOSFORMULADOR", //DatosFormulador
	"DATOSMAQUILADOR", //DatosMaquilador
	"Atributo condicional para registrar el uso autorizado del plaguicida o
	fertilizante de acuerdo a la regulación del país", //UsoAutorizado
	15000m, //PesoEnKg
	50000m, //ValorMercancia
	"MXN", //Moneda
	"0101300100", //FraccionArancelaria

```

null,          //UUIDComercioExt
"05",          //TipoMateria
"Atributo para expresar la descripción de la
materi"); //DescripcionMateria

```

5.10.6 Agregar DocumentacionAduanera a la última Mercancía

```

01 complCfdi.CP30_ADD_CartaPorte_Mercancias_mercancia_DocumentacionAduanera_UltimaMerc
02 ancia(
03         "01",          //TipoDocumento
04         "21 34 3913 1123456", //NumPedimento
05         "",            //IdentDocAduanero
06         "SIF0403229F9"); //RFCImpo
07

```

5.10.7 Agregar CantidadTransporta a la última Mercancía

```

01 complCfdi.CP30_ADD_CartaPorte_Mercancias_Mercancia_CantidadTransporta_UltimaMercanc
02 ia(
03         340m,          //Cantidad
04         "OR000001",    //IDOrigen
05         "DE000001",    //IDDestino
06         "");           //CvesTransporte
07

```

5.10.8 Establecer elemento Autotransporte para Mercancias

```

01 complCfdi.CP30_SET_CartaPorte_Mercancias_Autotransporte(
02         "TPAF01",      //PermSCT
03         "2141TAL31052011021001063", //NumPermisoSCT
04         "VL",          //ConfigVehicular
05         12.5m,         //PesoBrutoVehicular
06         "465AU2",      //PlacaVM
07         2013);         //AnioModeloVM

```

5.10.9 Establecer información de Seguros para el Autotransporte

```

01 complCfdi.CP30_SET_CartaPorte_Mercancias_Autotransporte_Seguros(
02         "SEGUROS ATLAS S.A", //AseguraRespCivil
03         "P00-1-7-118240-18", //PolizaRespCivil
04         "", //AseguraMedAmbiente
05         "", //PolizaMedAmbiente
06         "", //AseguraCarga
07         "", //PolizaCarga
08         null); //PrimaSeguro
09

```

5.10.10 Agregar un Tipo de Figura para FiguraTransporte

```

01 complCfdi.CP30_ADD_CartaPorte_TiposFigura(
02         "01",          //TipoFigura
03         "AAMI760614929", //RFCFigura
04         "PUE0036473",    //NumLicencia
05         "",              //NombreFigura
06         "",              //NumRegIdTribFigura
07         null,            //ResidenciaFiscalFigura

```

```

08      new CartaPorte10_Domicilio
09      {
10          Calle = "",
11          NumeroExterior = "",
12          NumeroInterior = "",
13          Colonia = "",
14          Localidad = "",
15          Referencia = "",
16          Municipio = "",
17          Estado = "CMX",
18          Pais = "MEX",
19          CodigoPostal = "01985"
20      };
21

```

5.10.11 Generar el Complemento Carta Porte 3.0

```

01 ReturnXmlComplemento rtnXmlCompl = complCfdi.CartaPorte30_generaComplemento();
02 //comprobamos que se creó el complemento
03 if (rtnXmlCompl.IsXmlComplementoCreated) {
04     //agregamos el complemento al comprobante
05     cfdi40.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
06     //agregamos el namespace de Impuestos Locales al comprobante
07     cfdi40.addNamespace(complCfdi.NsCartaPorte30);
08 }

```

5.11 Complemento Carta Porte 3.1

5.11.1 Definir el complemento Carta Porte

```

01 complCfdi.CP31_SET_CartaPorte("CCBCD94-870A-4332-A52A-A52AA52AA52A", //IdCCP
02     "Sí", //TranspInternac
03     "Entrada", //EntradaSalidaMerc
04     "USA", //PaisOrigenDestino
05     "04", //ViaEntradaSalida
06     525.11m, //TotalDistRec
07     "Sí", //RegistroISTMO
08     "05", //UbicacionPoloOrigen
09     "02"); //UbicacionPoloDestino
10

```

5.11.2 Agregar un Régimen Aduanero

```

01 complCfdi.CP31_ADD_CartaPorte_RegimenAduaneroCCP("IMD"); //RegimenAduanero

```

5.11.3 Agregar una Ubicación

```

01 int indx = complCfdi.CP31_ADD_CartaPorte_Ubicacion("Origen", //TipoUbicacion
02     "OR458563", //IDUbicacion
03     rfcReceptor, //RFCRemitenteDestinatario
04     "Cori$, S. A. de C. V.", //NombreRemitenteDestinatario
05     "", //NumRegIdTrib

```



```

06      "", //ResidenciaFiscal
07      "R0000", //NumEstacion
08      "PIEDRAS NEGRAS", //NombreEstacion
09      "", //NavegacionTrafico
10      new DateTime(2021, 10, 30, 12, 00, 00), //FechaHoraSalidaLlegada
11      "01", //TipoEstacion
12      null); //DistanciaRecorrida
13

```

5.11.4 Establecer el Domicilio para una ubicación

```

1  complCfdi.CP31_SET_CartaPorte_Ubicacion_Domicilio(indx,
02      new CartaPorte10_Domicilio
03      {
04          Calle = "CARRETERA REMITENTE NACIONAL",
05          NumeroExterior = "234325",
06          NumeroInterior = "545",
07          Colonia = "6003",
08          Localidad = "10",
09          Referencia = "KM9 A UN COSTADO DE LA CARRETERA FEDERAL",
10          Municipio = "087",
11          Estado = "VER",
12          Pais = "MEX",
13          CodigoPostal = "91013"
14      });
15

```

5.11.5 Establecer información de Mercancías

```

01  complCfdi.CP31_SET_CartaPorte_Mercancias(30000m, //PesoBrutoTotal
02      "MC", //UnidadPeso
03      28.026M, //PesoNetoTotal
04      2, //NumTotalMercancias
05      null, //CargoPorTasacion
06      null); //LogisticaInversaRecoleccionDevolucion
07

```

5.11.6 Agregar una Mercancía

```

01  complCfdi.CP31_ADD_CartaPorte_Mercancias_Mercancia(
02      "11121701", //BienesTransp
03      "2429320", //ClaveSTCC
04      "ASERRÍN", //Descripcion
05      15000m, //Cantidad
06      "XBX", //ClaveUnidad
07      "CAJA", //Unidad
08      null, //Dimensiones
09      null, //MaterialPeligroso
10      null, //CveMaterialPeligroso
11      null, //Embalaje
12      null, //DescripEmbalaje
13      "05", //SectorCOFEPRIS
14      "nombre común del ingrediente activo de los precursores, químicos de uso
15      dual, plaguicidas o fertilizantes que se trasladan a través de los distintos medios
16      de transporte.", //NombreIngredienteActivo
17      "", //NomQuimico
18      ""); //DenominacionGenericaProd

```

	"",	//DenominacionDistintivaProd
	"",	//Fabricante
	null,	//FechaCaducidad
	"",	//LoteMedicamento
	"",	//FormaFarmaceutica
	"",	//CondicionesEspTransp
	"",	//RegistroSanitarioFolioAutorizacion
	"",	//PermisoImportacion
	"48565254685654862548",	//FolioImpoVUCEM
	"",	//NumCAS
	"",	//RazonSocialEmpImp
	"9851316113231546565131321312329816513214",	//NumRegSanPlagCOFEPRIS
	"Atributo condicional para registrar el país y nombre o razón social de quien produce o fabrica el ingrediente activo del plaguicida o fertilizante.",	
	//DatosFabricante	
	"DATOSFORMULADOR",	//DatosFormulador
	"DATOSMAQUILADOR",	//DatosMaquilador
	"Atributo condicional para registrar el uso autorizado del plaguicida o fertilizante de acuerdo a la regulación del país",	
	//UsoAutorizado	
	15000m,	//PesoEnKg
	50000m,	//ValorMercancia
	"MXN",	//Moneda
	"0101300100",	//FraccionArancelaria
	null,	//UUIDComercioExt
	"05",	//TipoMateria
	"Atributo para expresar la descripción de la materi");	
	//DescripcionMateria	

5.11.7 Agregar DocumentacionAduanera a la última Mercancía

01	complCfdi.CP31_ADD_CartaPorte_Mercancias_mercancia_DocumentacionAduanera_UltimaMerc	
02	ancia(
03	"01",	//TipoDocumento
04	"21 34 3913 1123456",	//NumPedimento
05	"",	//IdentDocAduanero
06	"SIF0403229F9");	//RFCImpo
07		

5.11.8 Agregar CantidadTransporta a la última Mercancía

01	complCfdi.CP31_ADD_CartaPorte_Mercancias_Mercancia_CantidadTransporta_UltimaMercanc	
02	ia(
03	340m,	//Cantidad
04	"OR000001",	//IDOrigen
05	"DE000001",	//IDDestino
06	""	//CvesTransporte
07);	

5.11.9 Establecer elemento Autotransporte para Mercancias

01	complCfdi.CP31_SET_CartaPorte_Mercancias_Autotransporte(
02	"TPAF01",	//PermSCT
03	"2141TAL31052011021001063",	//NumPermisoSCT
04	"VL",	//ConfigVehicular
05	12.5m,	//PesoBrutoVehicular
06	"465AU2",	//PlacaVM
07	2013);	//AnioModeloVM

5.11.10 Establecer información de Seguros para el Autotransporte

```
01 complCfdi.CP31_SET_CartaPorte_Mercancias_Autotransporte_Seguros(
02     "SEGUROS ATLAS S.A",           //AseguraRespCivil
03     "P00-1-7-118240-18",         //PolizaRespCivil
04     "",                           //AseguraMedAmbiente
05     "",                           //PolizaMedAmbiente
06     "",                           //AseguraCarga
07     "",                           //PolizaCarga
08     null);                        //PrimaSeguro
09
```

5.11.11 Agregar un Tipo de Figura para FiguraTransporte

```
01 complCfdi.CP31_ADD_CartaPorte_TiposFigura(
02     "01",                          //TipoFigura
03     "AAMI760614929",              //RFCFigura
04     "PUE0036473",                //NumLicencia
05     "",                           //NombreFigura
06     "",                           //NumRegIdTribFigura
07     null,                         //ResidenciaFiscalFigura
08     new CartaPorte10_Domicilio
09     {
10         Calle = "",
11         NumeroExterior = "",
12         NumeroInterior = "",
13         Colonia = "",
14         Localidad = "",
15         Referencia = "",
16         Municipio = "",
17         Estado = "CMX",
18         Pais = "MEX",
19         CodigoPostal = "01985"
20     });
21
```

5.11.12 Generar el Complemento Carta Porte 3.1

```
01 ReturnXmlComplemento rtnXmlCompl = complCfdi.CartaPorte31_generaComplemento();
02 //comprobamos que se creó el complemento
03 if (rtnXmlCompl.IsXmlComplementoCreated) {
04     //agregamos el complemento al comprobante
05     cfdi40.addComplementoComprobante(rtnXmlCompl.XMLComplemento);
06     //agregamos el namespace de Impuestos Locales al comprobante
07     cfdi40.addNamespace(complCfdi.NsCartaPorte31);
08 }

```

5.12 Complemento Impuestos Locales

5.12.1 Definir el complemento Impuestos Locales

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

```

5.12.2 Agregar Impuesto Local Traslado al complemento

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

5.12.3 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 comprobante
05     cfdi40. addComplementoComprobante(rtnXmlCompl.XMLComplemento);
06     //agregamos el namespace de Impuestos Locales al comprobante
07     cfdi40.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 `cancelaCfdiNuevoProceso()`

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](#).

La lista de UUIDs a cancelar debe conformarse con un string para cada UUID a cancelar con el formato `[UUID a cancelar|MotivoCancelacion|UUID que sustituye]`, en caso de que no sea necesario incluir el UUID que sustituye, se deberá dejar el espacio en blanco conservando los caracteres pipe especificados.

```
01 Cancelacion2018 cancelacion2018 = new Cancelacion2018(
02     "JGE080430LN8",           //Usuario Sifei
03     "0e84ec92",               //Password Sifei
04     CFDI33.tipoTimbrado.Pruebas); //Ambiente
05 string msg = "";
06 //Lista de uuids del comprobante a cancelar
07 List<string> uuids = new List<string>();
08 uuids.Add("XXXXXX-447C-4F10-BCD7-D1949CDF636CXXX|02|");
09 try {
10     ReturnCancelacion2018 returnCancelacion2018 =
11     cancelacion2018.cancelaCfdi(ref msg,
12     uuids.ToArray(), //arreglo de Uuid
13     "MAR980114GQA", //rfcEmisor
14     CERTIFICADO_PATH, //rutaCert
15     CERTIFICADOKEY_PATH, //rutaKey
```

```

17 "12345678a"); //pass del certificado
18 if (!returnCancelacion2018.Procesado) {
19     Console.WriteLine(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 `cancelaCfdi()`

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         uuid.motivoCancelacion = "02";
17         uuid.uuidSustituye = "";
18         structUuids.Add(uuid);
19
20         string msg = "";
21
22         ReturnCancelacion2018 result = canc2018.cancelaCfdiNuevoProceso(ref msg, s
23         tructUuids.ToArray(), "S&S051221SE2",
24         @" CSD_S_& SOFTWARE_S.A_DE_C.V_S&S051221SE2_20190627_135134s.cer",
25         @" CSD_S_& SOFTWARE_S.A_DE_C.V_S&S051221SE2_20190627_135134.key",
26         "12345678a");
27
28         if (result.Procesado)
29         {
30             XmlDocument xmlAc = new XmlDocument();
31             xmlAc.LoadXml(result.AcuseBE.AcuseXml);
32             xmlAc.Save(@"acuseCanc.xml");
33         }
34         else
35         {
36             MessageBox.Show("Error al procesar cancelación: " + result.CodigoError
37             + " - " + result.ErrorDetalle);
38             return;
39         }

```

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.WriteLine(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 9.18 [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

```



```

32         foreach (var uuid in returnPeticionesPendientes.UUID) {
33             //en este ejemplo se usa un ListView para mostrar el UUID, pero es posible que
34             se desee guardar a BD como registro
35             listView2.Items.Add(new ListViewItem(uuid));
36             uuidPeticionesPendientes.Add(uuid);
37         }
38         //guardadndo el acuse a archivo(ejemplo)
39         XmlDocument xml = new XmlDocument();
40         xml.LoadXml(returnPeticionesPendientes.AcuseXml);
41
42         xml.Save(WORK_DIR+"\\AcusePeticionesPendientes_"+DateTime.Now.ToString("yyyy-MM-
43         ddTHH.mm.ss")+".xml");
44     }
45     //1101 no existen peticiones pendientes
46     else if (returnPeticionesPendientes.CodEstatus == "1101") {
47         Console.WriteLine("No hay peticiones pendientes");
48     }
49 }
50 }
51 catch (Exception ex) {
52     MessageBox.Show("Error al obtener peticiones Pendinetes de Cancelacion: " +
53     ex.Message);
54 }

```

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
06 //lista de objeto folios requerido por el servicio, incluye el UUID y la
07 respuesta:aceptacion o rechazo
08 List<folios> lstFolios = new List<folios>();
09 //Se establecen los UUID de las peticiones
10 List<string> uuidSeleccionados= getUUIDSeleccionados();
11
12 foreach (string uuid in uuidSeleccionados) {
13     //se crea objeto clase folio , que requiere el UUID y la respuesta
14     folios f = new folios();
15     //se asigna el UUID de la peticion a la que se le dara respuesta
16     f.uuid = uuid;
17     //se asigna la respuesta, que es donde se indica si se acepta o rechaza
18     f.respuesta = respuestaParaUUIDMarcados;
19 //se indica que el valor respuesta no es uno por defecto,sino que de lado del
20 cliente, realmente se esta estableciendo este valor
21 f.respuestaSpecified = true;
22 lstFolios.Add(f);//agregamos la respuesta
23 }
24 string msg="";
25 try {
26     // con este metodo, como receptor se acepta o rechaza
27     ReturnAceptacionRechazoResult returnAceptacionRechazoResult =
28     cancelacion2018.procesaRespuestaPeticionesCancelacion(
29     "MAR980114GQA",
30     CERTIFICADO_PATH,

```

```

31 CERTIFICADO_KEY_PATH,
32 "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 //en caso de ok
40 if (returnAceptacionRechazoResult.Procesado) {
41     if (returnAceptacionRechazoResult.CodEstatus == "1000") {
42         Console.WriteLine(returnAceptacionRechazoResult.AcuseXML);
43     }
44     XmlDocument xml = new XmlDocument();
45     xml.LoadXml(returnAceptacionRechazoResult.AcuseXML);
46     xml.Save(WORK_DIR + "\\AcuseProcesaRespuestaAceptacionRechazo_" +
47 DateTime.Now.ToString("yyyy-MM-ddTHH.mm.ss") + ".xml");
48 }
49 }
50 }
51 catch (Exception ex) {
52     MessageBox.Show("Error al procesar respuesta: " + ex.Message);
53 }
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 `Reten20_Define_Retenciones()`:

```

01 ret.Reten20_Define_Retenciones(
02     this.rutaCert,          //pRutaCert
03     "BZNPLUS339",          //pFolioInt
04     DateTime.Now,           //pFechaExp
05     "70780",                //pLugarExpRetenc
06     "26",                   //pCveRetenc
07     "",                      //pDescRetenc
08 );

```

Lo cual produce un XML con lo siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<retenciones:Retenciones xmlns:retenciones="http://www.sat.gob.mx/esquemas/retencionpago/2"
  xmlns:plataformasTecnologicas="http://www.sat.gob.mx/esquemas/retencionpago/1/PlataformasTecnologicas10"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.sat.gob.mx/esquemas/retencionpago/2 http://www.sat.gob.mx/esquemas/retenci
  Version="2.0"
  FolioInt="BZNPLUS339"
  Sello="EL1Sxz+VSKHO9dhVJnAOYnDkgNbcwKf0tyFC5sBfGGYznhg0VXpobf77K366G7Hsav93jJI3XCirKotsAFmv/PHm19/s+uEZ6l
  NoCertificado="300010000000500003431"
  Certificado="MIIFrTCCA5WgAwIBAgIUMzAwMDEwMDAwMDA1MDAwMDM0MzEwDQYJKoZIhvcNAQELBQAwggErMQ8wDQYDVQQDDAZBQyBVM
  FechaExp="2023-07-18T12:20:15"
  LugarExpRetenc="70780"
  CveRetenc="26">
```

Imagen 50 Atributos de Retenciones agregados

7.3 Definir CfdiRetenRelacionados

Para establecer los datos del comprobante relacionado se debe utilizar el método Reten20_Define_CfdiRetenRelacionados().

```
01 ret.Reten20_Define_CfdiRetenRelacionados("04", //pTipoRelacion
02 "AD61BF2E-2D20-478A-9E59-9AE0C9EC2301"); //pUUID
03
04
```

El nodo generado es Emisor:

```
<retenciones:CfdiRetenRelacionados TipoRelacion="04"
  UUID="AD61BF2E-2D20-478A-9E59-9AE0C9EC2301"/>
<retenciones:Emisor RfcE="H&amp;E951128469"
```

Imagen 51 Nodo CfdiRetenRelacionados agregado

7.4 Definir el Emisor

Para establecer los datos del Emisor del comprobante se debe utilizar el método Reten20_Set_Emisor().

```
01 ret.Reten20_Set_Emisor(this.rfcEmisor, //pRfcE
02 "HERRERIA & ELECTRICOS SA DE CV", //pNomDenRazSocE
03 "601" //pRegimenFiscale
04 );
```

El nodo generado es Emisor:

```
<retenciones:Emisor RfcE="H&amp;E951128469"
  NomDenRazSocE="HERRERIA &amp; ELECTRICOS SA DE CV"
  RegimenFiscale="601"/>
<retenciones:Receptor NacionalidadR="Nacional">
```

Imagen 52 Nodo Emisor agregado

7.5 Definir Receptor

Para establecer los datos de Receptor, se puede utilizar el método Reten20_Set_ReceptorNacional() o Reten20_Set_ReceptorExtranjero() según sea el caso, en el siguiente ejemplo se utiliza el primer método:

```
01 ret.Reten20_Set_ReceptorNacional("CACX7605101P8", //pRfcR
02 "Pruebas", //pNomDenRazSocR
03 "", //pCurpR
04 "94310" //pDomicilioFiscalR
```

);
--	----

El nodo generado es Receptor:

```
<retenciones:Receptor NacionalidadR="Nacional">
  <retenciones:Nacional RfcR="CACX7605101P8"
    NomDenRazSocR="Pruebas"
    DomicilioFiscalR="94310"/>
</retenciones:Receptor>
```

Imagen 53 Nodo Receptor agregado

7.6 Definir el Periodo

Para establecer el periodo del comprobante se debe utilizar el método Reten20_Set_Periodo():

```
01 ret.Reten20_Set_Periodo("01", //pMesIni
02 "01", //pMesFin
03 "2023"); //pEjercicio
04
```

El nodo generado es Periodo:

```
<retenciones:Periodo MesIni="01"
  MesFin="01"
  Ejercicio="2023"/>
```

Imagen 54 Nodo Periodo agregado

7.7 Definir Totales

Para establecer los totales del comprobante de Retenciones e Información de Pagos, se debe utilizar el método Reten20_Set_Totales():

```
01 ret.Reten20_Set_Totales(1681.06m, //pMontoTotOperacion
02 1681.06m, //pMontoTotGrav
03 0.00m, //pMontoTotExent
04 151.29m, //pMontoTotRet
  null, //pUtilidadBimestral
  null); //pISRCorrespondiente
```

Lo cual genera el siguiente nodo Totales:

```
<retenciones:Periodo MesIni="01" MesFin="01" Ejercicio="2023"/>
<retenciones:Totales MontoTotOperacion="1681.06" MontoTotGrav="1681.06" MontoTotExent="0.00" MontoTotRet="151.29">
  <retenciones:ImpRetenidos BaseRet="1681.06" ImpuestoRet="001" MontoRet="16.81" TipoPagoRet="03"/>
</retenciones:Totales>
```

Imagen 55 Nodo Totales agregado

7.8 Agregar Impuesto Retenido

Para agregar un impuesto retenido se debe utilizar el método Reten20_Add_ImpRetenidos ():

```
01 ret.Reten20_Add_ImpRetenidos(1681.06m, //pBaseRet
02 "001", //pImpuestoRet
03 16.81m, //pMontoRet
04 "03"); //pTipoPagoRet
```

Lo cual genera el nodo ImpRetenidos:

```
<retenciones:Totales MontoTotOperacion="1681.06" MontoTotGrav="1681.06" MontoTotExent="0.00" MontoTotRet="151.29">
  <retenciones:ImpRetenidos BaseRet="1681.06" ImpuestoRet="001" MontoRet="16.81" TipoPagoRet="03"/>
</retenciones:Totales BaseRet="268.96" ImpuestoRet="002" MontoRet="134.48" TipoPagoRet="01"/>
```

Imagen 56 Nodo ImpRetenidos agregado

7.9 Autorización para el uso de Sifei Signature

La clase `retencion` incluye el método `setIdenticacionSifei()` que debe invocarse antes de invocar a los métodos `generaXmlRetencion20()` y `sellaRetencion20()`, referirse a la misma sección para CFDI 3.3.

7.10 Generar el XML

Para generar el `XmlDocument` debemos invocar al método `generaXmlRetencion20()`, 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 result = this.generaXmlRetencion20();
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:plataformasTecnologicas="http://www.sat.gob.mx/esquemas/retencionpago/1/PlataformasTecnologicas10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.sat.gob.mx/esquemas/retencionpago/2
http://www.sat.gob.mx/esquemas/retencionpago/2/retencionpagov2.xsd
http://www.sat.gob.mx/esquemas/retencionpago/1/PlataformasTecnologicas10
http://www.sat.gob.mx/esquemas/retencionpago/1/PlataformasTecnologicas10/ServiciosPlataformasTecnologicas10.xsd"
Version="2.0"
FolioInt="BZNPLUS339"
Sello="@
NoCertificado="30001000000500003431"
Certificado="MIIFrTCCA5WgA..."
FechaExp="2023-07-18T13:45:39"
```

```

    LugarExpRetenc="70780"
    CveRetenc="26"
    xmlns:retenciones="http://www.sat.gob.mx/esquemas/retencionpago/2">
<retenciones:CfdiRetenRelacionados TipoRelacion="04"
    UUID="AD61BF2E-2D20-478A-9E59-9AE0C9EC2301"/>
<retenciones:Emisor RfcE="H&amp;E951128469"
    NomDenRazSocE="HERRERIA &amp; ELECTRICOS SA DE CV"
    RegimenFiscale="601"/>
<retenciones:Receptor NacionalidadR="Nacional">
    <retenciones:Nacional RfcR="CACX7605101P8"
    NomDenRazSocR="Pruebas"
    DomicilioFiscalR="94310"/>
</retenciones:Receptor>
<retenciones:Periodo MesIni="01"
    MesFin="01"
    Ejercicio="2023"/>
<retenciones:Totales MontoTotOperacion="1681.06"
    MontoTotGrav="1681.06"
    MontoTotExent="0.00"
    MontoTotRet="151.29">
    <retenciones:ImpRetenidos BaseRet="1681.06"
    ImpuestoRet="001"
    MontoRet="16.81"
    TipoPagoRet="03"/>
    <retenciones:ImpRetenidos BaseRet="268.96"
    ImpuestoRet="002"
    MontoRet="134.48"
    TipoPagoRet="01"/>
</retenciones:Totales>
</retenciones:Retenciones>

```

7.11 Sellar el Comprobante de Retención e Información de Pagos

Es posible sellar un Comprobante de Retención e Información de Pagos a través del método `sellaRetencion20()` 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. sellaRetencion20(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):

```
<?xml version="1.0" encoding="utf-8"?>
<retenciones:Retenciones xmlns:plataformasTecnologicas="http://www.sat.gob.mx/esquemas/retencionpago/1/PlataformasTecnologicas10"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.sat.gob.mx/esquemas/retencionpago/2 http://www.sat.gob.mx/esquemas/retenci
Version="2.0"
FolioInt="BZNPLUS339"
Sello="ELISxz+VSKHO9dhVJnAOYndkgNbcwKf0tyFC5sBfGGYznhg0VXpoBf77K366G7Hsav93jJI3XCirKoTsAFmv/PHml9/s+uEZf
NoCertificado="30001000000500003431"
Certificado="MIIFrTCCA5WgAwIBAgIUMzAwMDEwMDAwMDA1MDAwMDM0MzEwDQYJKoZIhvcNAQELBQAwggErMQ8wDQYDVQQDDAZBQyB1
FechaExp="2023-07-18T12:20:15"
LugarExpRetenc="70780"
CveRetenc="26"
xmlns:retenciones="http://www.sat.gob.mx/esquemas/retencionpago/2">
<retenciones:Emisor RfcE="H&E951128469"
NomDenRazSocE="HERRERIA &ELECTRICOS SA DE CV"
RegimenFiscalE="601"/>
<retenciones:Receptor NacionalidadR="Nacional">
<retenciones:Nacional RfcR="CACX7605101P8"
NomDenRazSocR="Pruebas"
DomicilioFiscalR="94310"/>
```

Imagen 57 Comprobante Sellado

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

7.12 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                                     //verificamos que haya sido timbrado:
06 if (rtnTimbre.IsTimbrado) {
07     //escribimos el cfdi timbrado
08     rtnTimbre.Xml.Save(@"C:\Workspace\SifeiSignature\ret_timbrado.xml");
09
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 Comprobante de Retenciones contendrá el nodo `TimbreFiscalDigital` (líneas 12 a 19):





Imagen 58 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     1,                   //pNumServ
05     1681.06m,            //pMonTotServSIVA
06     268.9696m,          //pTotalIVATrasladado
07     134.48m,             //pTotalIVARetenido
08     16.81m,              //pTotalISRRetenido
09     134.4896m,           //pDifIVAEntregadoPrestServ
10     14.66m,              //pMonTotalporUsoPlataforma
11     null,                //pMonTotalContribucionGubernamental
12 );
```

8.1.2 Agregar DetallesDelServicio

```
01 ret.Plat_Add_DetallesDelServicio(
02     "02",                //pFormaPagoServ
03     "05",                //pTipoDeServ
04     "",                  //pSubTipServ
05     "",                  //pRFCTerceroAutorizado
06     new DateTime(2023, 1, 6), //pFechaServ
07     1681.06m,            //pPrecioServSinIVA
08     true,                //integraImpTrasServ
09     1681.06m,            //pImpTras_Base
10     "02",                //pImpTras_Impuesto
11     retencion.tipoTasaOCuota.IVA_16, //pImpTras_TasaCuota
12     268.9696m,          //pImpTras_Importe
13     true,                //integraComServ
14     1681.06m,            //pComServ_Base
15     null,                //pComServ_Porcentaje
16     14.66m,              //pComServ_Importe
17     false,               //integraContribucionGubernamental
18     null,                //pContrGub_ImpContrib
19     "",                  //pContrGub_EntidadDondePaga
```


);
--	----

8.1.3 Generar Complemento Servicios Plataformas Tecnológicas

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



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
CFDI40	Clase que representa el comprobante CFDI versión 4.0, 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 CFDI40.generaXml() , por tanto, esta clase no contiene métodos. Si algún error ocurre al generar el comprobante cuando no es válido, esta clase contendrá los códigos de error y mensajes correspondientes a la matriz de errores, en caso de éxito contiene el comprobante.
ReturnTimbre	Clase que representa el resultado de obtener el timbre CFDI40.recuperaTimbre() , por tanto, no contiene métodos.
ReturnXmlComplemento	Clase que representa el resultado de generar un complemento a través del método Pagos20.generaComplemento de clase Complemento , por tanto, esta clase no tiene métodos.

9.3 Namespace SifeiSignature3_3.Version3_3

Clases:

- [CFDI40](#)



9.3.1 Clase CFDI33

9.3.1.1 Constructor

Constructor	Descripción
<pre>public CFDI40(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 pExportacion, string pMetodoPago, string pLugarEped, string pConfirmacion, string rutaCert)</pre>	Inicializa el objeto CFDI40 con los atributos principales del comprobante

9.3.2 Métodos

Método de CFDI40	Descripción
<pre>public static ReturnTimbre recuperaTimbre(string usuario, string password, XmlDocument xml, tipoTimbrado TipoTimbrado);</pre>	Recupera un timbre del servicio web de sifei, el cual si es correcto es devuelto en el objeto de tipo ReturnTimbre . Para más detalle ver la clase ReturnTimbre .
<pre>public void addAddenda(XmlElement pAddenda);</pre>	Agrega un nodo hijo addenda de tipo any (cualquiera) al nodo addenda del comprobante. XPath: Cfdi:Addenda/cfdi:any
<pre>public void addCfdirelacionados(string pTipoRelacion, params string[] pLsUUIDs)</pre>	Crea el nodo CfdiRelacionados con el atributo TipoRelacion indicado en el parámetro @pTipoRelacion y agrega los nodos CfdiRelacionado con cada UUID especificado en el array @pLsUUIDs. XPath: Cfdi:CfdiRelacionados/cfdi:Cfdirelacionado Excepciones: Exception Si @pUUID no cumple con el formato y restricciones establecidas por el SAT.
<pre>public void addComplementoComprobante(XmlEle ment pComplemento);</pre>	Agrega un complemento al nodo complemento.
<pre>public bool addComplementoConcepto(</pre>	Agrega el nodo @complemento como hijo al Concepto apuntado por @indexConcepto.

Método de CFDI40	Descripción
<pre>int indexConcepto, XmlElement complemento);</pre>	<p>xPath: Cfdi:Conceptos/cfdi:concepto[@indexConcepto]/ComplementoConcepto</p> <p>Excepciones: Exception 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 pObjetoImp);</pre>	<p>Crea y agrega un nodo hijo concepto al nodo conceptos (Cfdi:Conceptos).</p> <p>xPath: Cfdi:Conceptos/cfdi:concepto</p> <p>Returns Índice del nodo concepto que se agregó</p> <p>Excepciones: Exception Si los parámetros no cumplen el formato y restricciones definidas por el SAT</p> <p>Returns Índice del nodo Concepto</p>
<pre>public void addCuentaPredialToConcepto(int pIndexConc, params string[] pLsNumeros);</pre>	<p>Crea y agrega un nodo hijo CuentaPredial al nodo Concepto).</p> <p>xPath: Cfdi:Conceptos/cfdi:concepto/cfdi:CuentaPredial</p> <p>Excepciones: Exception Si los parámetros no cumplen el formato y restricciones definidas por el SAT Si el concepto apuntado por @ pIndexConc no existe</p>
<pre>public void addImpRetencion(string pImpuesto, decimal pImporte);</pre>	<p>Crea y agrega un nodo Retencion con el impuesto e importe indicado al nodo Impuestos del comprobante, verifica que exista el impuesto en el catálogo impuesto y que el importe corresponda con tipo importe (definido por el estándar del SAT).</p> <p>XPath: cfdi:Impuestos/cfdi:Retenciones/cfdi:Retencion</p> <p>Excepciones: Exception Si el nodo Impuestos no existe (es null), no ha sido definida. Lanza excepciones los parámetros no cumplen el formato y restricciones definidas por el SAT. ArgumentException Si alguno de los parámetros: @pImpuesto no existe en su catálogo.</p>
<pre>public void addImpRetencionToConcepto(int pIndexConc, decimal pBase, string pImpuesto,</pre>	<p>Crea y agrega un nodo Retencion al Concepto indicado por el valor @pIndexConc, el nodo retencion tendrá los valores: pBase, pImpuesto, pTipoFactor, pTasaOCuota, pImporte.</p> <p>XPath:</p>

Método de CFDI40	Descripción
<pre>string pTipoFactor, decimal pTasaOCuota, decimal pImporte);</pre>	<p>Cfdi:Conceptos/cfdi:concepto[@pIndexConc]/cfdi:Impuestos/cfdi:Retenciones/cfdi:Retencion</p> <p>Excepciones: Exception Si el concepto al que apunta @pIndexConc no existe (es null o está fuera de rango). Lanza excepciones los parámetros no cumplen el formato y restricciones definidas por el SAT. ArgumentException Si alguno de los parámetros: @pImpuesto, @pTipoFactor no existe en su catálogo.</p>
<pre>public void addImpTraslado(decimal pBase, string pImpuesto, string pTipoFactor, decimal? pTasaCuota, decimal? pImporte);</pre>	<p>Crea y agrega un nodo Traslado con el @pImpuesto, @ptipofactor, @pTasaCuota e @pimporte indicado al nodo Impuestos del comprobante, verifica que exista el impuesto en el catálogo impuesto y que el importe corresponda con tipo importe (definido por el estándar del SAT).</p> <p>XPath: cfdi:Impuestos/cfdi:Traslados/cfdi:Traslado</p> <p>Excepciones: Exception Lanza excepciones los parámetros no cumplen el formato y restricciones definidas por el SAT. ArgumentException Si alguno de los parámetros: @pImpuesto, @pTipoFactor no existe en su catálogo.</p>
<pre>public void addImpTrasladoToConcepto(int pIndexConc, decimal pBase, string pImpuesto, string pTipoFactor, decimal? pTasaOCuota, decimal? pImporte);</pre>	<p>Crea y agrega un nodo Traslado al Concepto indicado por el valor @pIndexConc, el nodo traslado tendrá los valores: @pBase, @pImpuesto, @pTipoFactor, @pTasaOCuota, @pImporte.</p> <p>XPath: Cfdi:Conceptos/cfdi:concepto[@pIndexConc]/cfdi:Impuestos/cfdi:Traslados/cfdi:Traslado</p> <p>Excepciones: Exception Si el concepto apuntado por @pIndexConc no existe. Lanza excepciones si no existen los valores en el catálogo o si no cumplen los tipos.</p>
<pre>public void addInformacionAduaneraToConcepto(int pIndexConc, string pNumPedimento);</pre>	<p>Crea y agrega un nodo InformacionAduanera al Concepto indicado por el valor @pIndexConc, el nodo InformacionAduanera tendrá los valores: @pNumPedimento.</p> <p>Lanza excepciones si no existen los valores en el catálogo o si no cumplen los tipos.</p> <p>XPath: Cfdi:Conceptos/cfdi:concepto[@pIndexConc]/cfdi:InformacionAduanera</p> <p>Excepciones:</p>

Método de CFDI40	Descripción
	Exception Lanza excepciones si no existen los valores en el catálogo o si no cumplen los tipos.
<pre>public void addNamespace(namespaces ns);</pre>	Agrega un <i>namespace</i> a la lista de <i>namespace</i> que contendrá el comprobante
<pre>public ReturnXML generaXml(string pCadConex, bool aplicaValidacionesMatrizSat = true, MidpointRounding pTipoRedondeo = MidpointRounding.AwayFromZero);</pre>	Genera el comprobante con los datos de la instancia, y retorna un objeto tipo <i>ReturnXML</i> , el cual contiene información acerca del resultado de generar el XML, contiene una referencia al objeto <i>xmlDocument</i> .
<pre>public List<namespaces> getNamespaces();</pre>	Obtiene la lista de <i>namespaces</i> del comprobante.
<pre>public ReturnXML sellaComprobante(XmlDocument xml, string rutaCert, string rutaKey, string pwdCert, bool asignaFechaEmisionActual, bool validaXsds);</pre>	Recibe una referencia del de tipo <i>XmlDocument</i> (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.
<pre>public void setACuentaTercerosToConcepto(int pIndexConc, string pRfcACuentaTerceros, string pNombreACuentaTerceros, string pRegimenFiscalACuentaTerceros, string pDomicilioFiscalACuentaTerceros);</pre>	Crea y agrega un nodo <i>ACuentaTerceros</i> al Concepto indicado por el valor <i>@pIndexConc</i> , el nodo <i>ACuentaTerceros</i> tendrá los valores: <i>@pRfcACuentaTerceros</i> , <i>@pNombreACuentaTerceros</i> , <i>@pRegimenFiscalACuentaTerceros</i> y <i>@pDomicilioFiscalACuentaTerceros</i> . Lanza excepciones si no existen los valores en el catálogo o si no cumplen los tipos. XPath: <i>Cfdi:Conceptos/cfdi:concepto[@pIndexConc]/cfdi:ACuentaTerceros</i> Excepciones: Exception Lanza excepciones si no existen los valores en el catálogo o si no cumplen los tipos.
<pre>public void setEmisor(string pRfc, string pNombre, string pRegFiscal);</pre>	Crea y establece un nodo <i>Emisor</i> como hijo directo al comprobante, verifica que existe el régimen fiscal en el catálogo y lanza una excepción sino existe. XPath: <i>Cfdi:Emisor</i> Excepciones: Exception Si los parámetros no cumplen con el formato y restricciones establecidas por el SAT. ArgumentException Si alguno de los parámetros: <i>@pRegFiscal</i> no existe en su catálogo.
<pre>public void setInformacionGlobal(string pPeriodicidad, string pMeses, short pAño);</pre>	Crea y establece un nodo <i>InformacionGlobal</i> como hijo directo al comprobante. XPath: <i>Cfdi:InformacionGlobal</i>

Método de CFDI40	Descripción
);	Excepciones: Exception Si los parámetros no cumplen con el formato y restricciones establecidas por el SAT.
<pre>public void setReceptor(string pRfc, string pNombre, string pDomicilioFiscalReceptor, string pResidenciaFiscal, string pNumRegIdTrib, string pRegimenFiscalReceptor string pUsoCfdi);</pre>	Crea y establece un nodo Receptor como hijo directo al comprobante. XPath: Cfdi:Receptor Excepciones: Exception Si los parámetros no cumplen con el formato y restricciones establecidas por el SAT. ArgumentException Si alguno de los parámetros: @pUsoCfdi, @pResidenciaFiscal no existe en su catálogo.
<pre>public void setTotalImpuestos(decimal? pTotImpRetenidos, decimal? pTotImpTrasladados);</pre>	Crea y establece un nodo Impuestos (ver XPath abajo) como hijo directo al comprobante, los parámetros @pTotImpRetenidos y @pTotImpTrasladados son opcionales, pero si se pasan deben cumplir con el tipo t_Importe , en caso contrario se lanza una excepción. XPath: Cfdi: Impuestos Excepciones: Exception Si los parámetros son establecidos (no null) y no cumplen los formatos y restricciones del SAT.
<pre>public ReturnTimbre timbraComprobante(string serie, string idEquipo, XmlDocument cfdi);</pre>	Genera el timbre del comprobante a partir de XML. Retorna un objeto ReturnTimbre con el Comprobante timbrado y el cbb. @cfdi XmlDocument con el comprobante a timbrar.
<pre>public void setIdenticacionSifei(string pUsuarioSifei, string pPasswordSifei, tipoTimbrado pTipo)</pre>	Establece los valores para autenticarse en los servidores de Sifei. Estos valores se utilizarán para las operaciones de generación de XML, sellado y timbrado. @usuarioSifei asignado para usar <i>SifeiSignature</i> @passwordSifei asignado para usar <i>SifeiSignature</i> @Tipo es un valor de enumerado tipoTimbrado para indicar si es de tipo pruebas o productivo.
<pre>public ReturnTimbre depositaFacturaGlobal(string serie, string idEquipo, XmlDocument cfdi, ref string msg)</pre>	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. @cfdi XmlDocument con el comprobante a timbrar.

Método de CFDI40	Descripción
<pre>public static ReturnTimbre recuperaTimbreFacturaGlobal(string 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 Enumeracion tipoTimbrado

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

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
public string UUID { get; set; }
public string SelloPAC { get; set; }
public string NoCertificado { get; set; }
public string FechaTimbre { get; set; }
public string CadenaOriginalTimbre { get; set; }
public string RfcProvCertif { get; set; }
public string Version { get; set; }
public string Leyenda { get; set; }
public string SelloCFD { get; set; }

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 1.1
<code>public namespaces NsComercioExterior2_0 { get}</code>	Obtiene el <i>namespace</i> de Comercio Exterior versión 2.0
<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 NsHYP10{ get}</code>	Obtiene el <i>namespace</i> de Hidrocarburos y Petrolíferos 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
<code>public namespaces NsCartaPorte20 { get}</code>	Obtiene el <i>namespace</i> de Complemento Carta Porte versión 2.0
<code>public namespaces NsCartaPorte30 { get}</code>	Obtiene el <i>namespace</i> de Complemento Carta Porte versión 3.0
<code>public namespaces NsCartaPorte31 { get}</code>	Obtiene el <i>namespace</i> de Complemento Carta Porte versión 3.1
<code>public namespaces NsPagos20 { get}</code>	Obtiene el <i>namespace</i> de Complemento pagos versión 2.0
<code>public namespaces NsDonat11{ get}</code>	Obtiene el <i>namespace</i> de Complemento Donatarias versión 1.1

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 Excepciones: Exception Esta excepción ocurre si los parámetros no cumplen con el formato válido establecido por el SAT. También cuando no se ha definido la instancia Comercio Exterior.
<code>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,</code>	Crea el nodo Destinatario y su hijo Domicilio . Se hacen las validaciones correspondientes. XPath: Cfdi:Complemento/ cce11:ComercioExterior/Destinatario Excepciones:

Métodos complemento	Descripción
<pre>string pDomReferencia, string pDomMunicipio, string pDomEstado, string pDomPais, string pDomCP)</pre>	<p>Exception Esta excepción ocurre si no cumple con el formato válido establecido por el SAT.</p> <p>ArgumentException Ocurre cuando pDomPais no existe en el catálogo.</p> <p>Returns: Índice del nodo Destinatario que se agregó.</p>
<pre>public void Cce11_addDestinatarioDomicilio(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 Domicio al nodo Destinatario.</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior/Destinatario/Domicio</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 Mercancia a Mercancias.</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior/Mercancias/Mercancia</p> <p>Returns: Índice del nodo Mercancia que se agregó.</p>
<pre>public void Cce11_addPropietario(string pNumRegIdTrib, string pResidenciaFiscal)</pre>	<p>Crea y establece un nodo Propietario como hijo a ComercioExterior.</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior/Propietario</p>
<pre>public void Cce11_Define_ComercioExterior(string pMotivoTrasl, string pTipoOperacion, string pClavePedim, int? pCertifOrigen, string pNumCertifOrigen, string pNumExportadorConf, string pIncoterm, int? pSubdivision, string pObservaciones, decimal? pTipoCambioUSD, decimal? pTotalUSD)</pre>	<p>Define el complemento ComercioExterior.</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior</p> <p>Excepciones: Exception Esta excepción ocurre si no cumple con el formato válido establecido por el SAT. ArgumentException Ocurre cuando un parámetro no existe en el catálogo.</p>
<pre>public ReturnXmlComplemento Cce11_generaComplemento()</pre>	<p>Genera el complemento de comercio exterior, lanza una excepción si no se ha definido la instancia comercio exterior (con el método Cce11_Define_ComercioExterior())</p>
<pre>public void Cce11_setEmisor(string pCarp, string pDomCalle, string pDomNumExt, string pDomNumInt, string pDomColonia, string pDomLocalidad, string pDomReferencia, string pDomMunicipio, string pDomEstado, string pDomPais,</pre>	<p>Crea y establece un nodo Emisor (como hijo de ComercioExterior).</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior/Emisor</p> <p>Excepciones: Exception Si algún atributo no cumple con el formato definido por el SAT, o si no se ha definido el complemento ComercioExterior11.</p>

Métodos complemento	Descripción
<pre>string pDomCP) 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) public void CCE20_addDescripcionEspecifica(int pIndexMercancia, string pMarca, string pModelo, string pSubmodelo, string pNumSerie);</pre>	<p>Crea y establece un nodo Receptor (como hijo de ComercioExterior).</p> <p>Xpath: Cfdi:Complemento/cce11:ComercioExterior/Emisor</p> <p>Excepciones: Exception Si algún atributo no cumple con el formato definido por el SAT, o si no se ha definido el complemento ComercioExterior11</p> <p>Agrega la descripción al complemento de Comercio Exterior.</p> <p>xPath: Cfdi:Complemento/cce20:ComercioExterior/Mercancias/Mercancia/DescripcionesEspecificas</p> <p>Excepciones: Exception Esta excepción ocurre si los parámetros no cumplen con el formato válido establecido por el SAT. También cuando no se ha definido la instancia Comercio Exterior.</p>
<pre>public int CCE20_addDestinatario(string pNumRegIdTrib, string pNombre, 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 Destinatario y su hijo Domicilio. Se hacen las validaciones correspondientes.</p> <p>Xpath: Cfdi:Complemento/cce20:ComercioExterior/Destinatarior</p> <p>Excepciones: Exception Esta excepción ocurre si no cumple con el formato válido establecido por el SAT. ArgumentException Ocurre cuando pDomPais no existe en el catálogo.</p> <p>Returns: Índice del nodo Destinatario que se agregó.</p>
<pre>public void CCE20_addDestinatarioDomicilio(int pIndexDestinatario, string pDomCalle, string pDomNumExt, string pDomNumInt, string pDomColonia, string pDomLocalidad, string pDomReferencia, string pDomMunicipio, string pDomEstado, string pDomPais, string pDomCP);</pre>	<p>Crea y establece un nodo hijo Domicio al nodo Destinatario.</p> <p>Xpath: Cfdi:Complemento/cce20:ComercioExterior/Destinatarior/Domicio</p>
<pre>public int CCE20_addMercancia(string pNoIdentificacion, string pFraccionAranc, decimal? pCantidadAduana, string pUnidadAduana, decimal? pValorUnitAduana, decimal pValorDolares)</pre>	<p>Crea y establece un nodo hijo Mercancia a Mercancias.</p> <p>Xpath: Cfdi:Complemento/cce20:ComercioExterior/Mercancias/Mercancia</p> <p>Returns: Índice del nodo Mercancia que se agregó.</p>

Métodos complemento	Descripción
<pre>public void CCE20_addPropietario(string pNumRegIdTrib, string pResidenciaFiscal)</pre>	<p>Crea y establece un nodo Propietario como hijo a ComercioExterior.</p> <p>Xpath: Cfdi:Complemento/cce20:ComercioExterior/Propietario</p>
<pre>public void CCE20_Define_ComercioExterior(string pMotivoTrasl, string pClavePedim, int pCertifOrigen, string pNumCertifOrigen, string pNumExportadorConf, string pIncoterm, string pObservaciones, decimal pTipoCambioUSD, decimal pTotalUSD)</pre>	<p>Define el complemento ComercioExterior.</p> <p>Xpath: Cfdi:Complemento/cce20:ComercioExterior</p> <p>Excepciones: Exception Esta excepción ocurre si no cumple con el formato válido establecido por el SAT. ArgumentException Ocurre cuando un parámetro no existe en el catálogo.</p>
<pre>public ReturnXmlComplemento CCE20_generaComplemento()</pre>	<p>Genera el complemento de comercio exterior, lanza una excepción si no se ha definido la instancia comercio exterior (con el método CCE20_Define_ComercioExterior())</p>
<pre>public void CCE20_setEmisor(string pCurp, string pDomCalle, string pDomNumExt, string pDomNumInt, string pDomColonia, string pDomLocalidad, string pDomReferencia, string pDomMunicipio, string pDomEstado, string pDomPais, string pDomCP)</pre>	<p>Crea y establece un nodo Emisor (como hijo de ComercioExterior).</p> <p>Xpath: Cfdi:Complemento/cce20:ComercioExterior/Emisor</p> <p>Excepciones: Exception Si algún atributo no cumple con el formato definido por el SAT, o si no se ha definido el complemento ComercioExterior11.</p>
<pre>public void CCE20_setReceptor(string pNumRegIdTrib, string pDomCalle, string pDomNumExt, string pDomNumInt, string pDomColonia, string pDomLocalidad, string pDomReferencia, string pDomMunicipio, string pDomEstado, string pDomPais, string pDomCP)</pre>	<p>Crea y establece un nodo Receptor (como hijo de ComercioExterior).</p> <p>Xpath: Cfdi:Complemento/cce20:ComercioExterior/Emisor</p> <p>Excepciones: Exception Si algún atributo no cumple con el formato definido por el SAT, o si no se ha definido el complemento ComercioExterior11</p>
<pre>public void Det131_AddAllowanceCharge(detallistaAllowanceChargeAllowanceCharge Type allChargeType, detallistaAllowanceChargeSettlementType settType, string numConsec, detallistaAllowanceChargeSpecialServicesType? specialServType, decimal? porcent)</pre>	<p>Crea y establece un nodo allowanceCharge (como hijo de Detallista).</p> <p>Xpath: Cfdi:Complemento/Detallista/ allowanceCharge</p> <p>Excepciones: Exception Si algún atributo no cumple con el formato definido por el SAT, o si no se ha definido el complemento Detallista (ver Det131_Define_Detallista()), o si se ha sobrepasado el límite de 99.</p>
<pre>public void Det131_addCurrency(detallistaCurrencyCurrencyISOCode isoCode, /*enumeración */ detallistaCurrencyCurrencyFunction funcion1,</pre>	<p>Crea y agrega un nodo currency como hijo al nodo detallista.</p> <p>Xpath: Cfdi:Complemento/Detallista/ currency[]</p>

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

Métodos complemento	Descripción
<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 TotalAllowanceCharge como hermano al nodo TotalAllowanceCharge.</p> <p>Se debe indicar si es descuento o cargo; es opcional indicar el tipo de servicio y el total de los cargos o descuentos.</p> <p>Xpath: Cfdi:Complemento/Detallista/ TotalAllowanceCharge[]</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos (30) referenceIdentification. Por último, si no se ha establecido orderIdentification (que es requerido).</p>
<pre>public int Det131_AgregaSpecialInstruction(detallistaSpecialInstructionCode Code,/* tipo de instrucciones comerciales*/ string text)</pre>	<p>Crea y agrega un nodo specialInstruction como hermano al nodo specialInstruction.</p> <p>Xpath: Cfdi:Complemento/Detallista/ specialInstruction[]</p> <p>Returns: Índice del nodo specialInstruction que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista o si se superó el número máximo de nodos (4) specialInstruction.</p>
<pre>public void Det131_AgregaTextASpecialInstruct ion(int indexSpecialInstr, /*index del nodo specialInstruction */ string text)</pre>	<p>Crea y agrega un nodo text a specialInstruction (como hijo) dado un índice (el cual indica a qué nodo specialInstruction será el padre).</p> <p>Xpath: Cfdi:Complemento/Detallista/ specialInstruction[indexSpecialInstr]/text</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista, si no existe el nodo specialInstruction o si se superó el número máximo de nodos (15) text.</p>
<pre>public void Det131_Agrega_RequestForPaymentId entification_Detallista(detallistaRequestForPaymentIdenti ficationEntityType EntityType /* tipo de transacción*/)</pre>	<p>Crea y agrega (o reemplaza si existe) un requestForPaymentIdentification a detallista (como hijo) para indicar la transacción.</p> <p>Xpath: Cfdi:Complemento/Detallista/ requestForPaymentIdentification/ entityType</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista.</p>
<pre>public void Det131_Define_Detallista(detallis taDocumentStatus DocumentStatus)</pre>	<p>Define el complemento detallista del comprobante.</p> <p>Xpath: Cfdi:Complemento/Detallista</p>

Métodos complemento	Descripción
<pre>public ReturnXmlComplemento Det131_generaComplemento()</pre>	<p>Retorna el complemento detallista en el objeto ReturnXmlComplemento si todo fue bien.</p> <p>Excepciones: Exception Ocurre si no se ha establecido en complemento Detallista.</p>
<pre>public void Det131_SetBuyer(string gln, /* el número global de localización */ string personOrDepartmentName /* contacto de compras */)</pre>	<p>Crea y agrega (o reemplaza si existe) un buyer a detallista (como hijo) para indicar la información del comprador.</p> <p>Xpath: Cfdi:Complemento/Detallista/ buyer/gln, buyer/contactInformation/ personOrDepartmentName/text</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista.</p>
<pre>public void Det131_SetDeliveryNote(DateTime? referenceDate, /*opcional, pasar null para no agregar atributo*/ string ReferenceIdentification)</pre>	<p>Crea y establece un DeliveryNote a detallista (como hijo) para indicar la información recepción de mercancía.</p> <p>Xpath: Cfdi:Complemento/Detallista/ DeliveryNote</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido en complemento Detallista.</p>
<pre>public void Det131_SetInvoiceCreator(string gln, detallistaInvoiceCreatorAlternate PartyIdentificationType? tipo, /* tipo de identificación secundaria se le asigno al proveedor */ string value, string name, string streetAddressOne,/* domicilio del emisor de la factura */ string city, string cp /* codigo postal del domicilio del emisor de la factura */)</pre>	<p>Crea y establece el InvoiceCreator a detallista (como hijo).</p> <p>Xpath: Cfdi:Complemento/Detallista/ InvoiceCreator/nameAndAddress, InvoiceCreator/ alternatePartyIdentification</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Detallista.</p>
<pre>public void Det131_SetOrderIdentification(DateTime? referenceDate, string ReferenceIdentification)</pre>	<p>Crea y establece el orderIdentification a detallista (como hijo).</p> <p>Xpath: Cfdi:Complemento/Detallista/ orderIdentification/referenceIdentification</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Detallista.</p>
<pre>public void Det131_SetPaymentTerms(detallistaPaymentTermsPaymentTerm sEvent? termsEvent,</pre>	<p>Crea y establece el paymentTerms a detallista (como hijo) para especificar los términos de pago de la factura</p> <p>Xpath:</p>

Métodos complemento	Descripción
<pre> dettallistaPaymentTermsPaymentTerm sRelationTime? termsRelationTime, dettallistaPaymentTermsNetPaymentN etPaymentTermsType? netPaymntTermsType, /*para el nodo paymentTimePeriod*/ dettallistaPaymentTermsNetPaymentP aymentTimePeriodTimePeriodDueTime Period? pTimePeriod, /*para el nodo paymentTimePeriod*/ string valueTimePeriod, dettallistaPaymentTermsDiscountPay mentDiscountType? descType, /*para el nodo discountPayment */ string porcent /*para el nodo discountPayment */) public void Det131_SetSeller(string strGln, dettallistaSellerAlternatePartyIde ntificationType tipo, string alternatePartyIdentification) </pre>	<p>Cfdi:Complemento/Detallista/ paymentTerms/ y/o Cfdi:Complemento/Detallista/ paymentTerms/paymentTimePeriod y/o Cfdi:Complemento/Detallista/ paymentTerms/discountPayment</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Detallista o si se estableció @pTimePeriod o valueTimePeriod cuando se pasó @netPaymntTermsType como null (ya que estos pertenecen a dicho nodo).</p>
<pre> public void Det131_SetSeller(string strGln, dettallistaSellerAlternatePartyIde ntificationType tipo, string alternatePartyIdentification) </pre>	<p>Crea y establece el seller a dettallista (como hijo) para especificar la información del vendedor.</p> <p>Xpath: Cfdi:Complemento/Detallista/ seller</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Detallista.</p>
<pre> public void Det131_SetShipTo(string gln, string name, string streetAddressOne, string city, string cp) </pre>	<p>Crea y establece el shipTo a dettallista (como hijo) para especificar la ubicación donde debe realizarse la entrega de la mercancía.</p> <p>Xpath: Cfdi:Complemento/Detallista/ shipTo</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Detallista.</p>
<pre> public void Det131_setTotalAmount(decimal amnt) </pre>	<p>Crea y establece el totalAmount a dettallista (como hijo).</p> <p>Xpath: Cfdi:Complemento/Detallista/ totalAmount</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Detallista.</p>
<pre> public ReturnXmlComplemento Iedu10_generaComplemento(string nombreAlumno, string curp, instEducativasNivelEducativo nivelEducativo, string autRVOE, string RfcPago) </pre>	<p>Genera el complemento concepto de instituciones educativas privadas. El complemento viene la propiedad XMLComplemento de ReturnXmlComplemento.</p> <p>Excepciones: Exception Esta excepción ocurre si los parámetros no cumplen la especificación del SAT en formato o restricción: @curp, @rfcPago, @autRVOE.</p>
<pre> public ReturnXmlComplemento HidroYPetro10_generaComplemento(s tring pTipoPermiso, string </pre>	<p>Genera el complemento concepto de Hidrocarburos y Petrolíferos versión 1.0. El complemento viene la propiedad XMLComplemento de ReturnXmlComplemento.</p>

Métodos complemento	Descripción
<p>pNumeroPermiso, string pClaveHYP, string pSubProductoHYP)</p>	<p>Excepciones: Exception Esta excepción ocurre si los parámetros no cumplen la especificación del SAT respecto a valores requeridos.</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 RetencionesLocales a ImpuestosLocales (como hijo).</p> <p>Xpath: Cfdi:Complemento/ImpuestosLocales/RetencionesLocales</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento ImpuestosLocales. Si @nombre es null o cadena vacía.</p>
<pre>public void ImpLocal10_Agrega_TrasladoImpuest oLocal(string nombre, decimal importe, decimal tasa) </pre>	<p>Crea y establece el RetencionesLocales a ImpuestosLocales (como hijo).</p> <p>Xpath: Cfdi:Complemento/ImpuestosLocales/RetencionesLocales</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento ImpuestosLocales. Si @nombre es null o cadena vacía.</p>
<pre>public void ImpLocal10_Define_ImpuestoLocal(decimal totalTraslado, decimal totalRetencion) </pre>	<p>Crea y establece el complemento ImpuestosLocales.</p> <p>Xpath: Cfdi:Complemento/ImpuestosLocales</p> <p>Excepciones: Exception Ninguna.</p>
<pre>public ReturnXmlComplemento ImpLocal10_generaComplemento() </pre>	<p>Genera el complemento impuestos locales que es devuelto en el objeto ReturnXmlComplemento, si existe un error no fatal es incluido en el mismo objeto. Ver propiedades:</p> <ul style="list-style-type: none"> ReturnXmlComplemento.IsXmlComplementoCreated ReturnXmlComplemento.ErrorDetalle <p>Excepciones: Exception Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación. Si no se pudo formar el nodo IEDU.</p>
<pre>public void Nomina12_addDeducccion(string pTipoDeducc, string pClave, string pConcepto, decimal pImporte) </pre>	<p>Crea y establece una Deducccion a Deducciones (como hijo).</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12: Deducciones/Deducccion[]</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Nómina 1.2.</p>

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

Métodos complemento	Descripción
<pre>public int Nomina12_addPercepcion(string pTipoPercepcion, string pClave, string pConcepto, decimal pImporteGravado, decimal pImporteExento)</pre>	<p>Crea y establece un nodo Percepcion como hijo a Percepciones.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:Percepciones/Percepcion[]</p> <p>Returns: Índice del nodo Percepcion que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha establecido el complemento Nómina 1.2. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p> <p>ArgumentException si @pTipoPercepcion no existe en el catálogo definido por el SAT</p>
<pre>public void Nomina12_addSubContratacion(string pRfcLabora, decimal pPorcentajeTiempo)</pre>	<p>Crea y establece un nodo SubContratacion como hijo a Receptor.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:Receptor/SubContratacion[]</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void Nomina12_DefineNomina(string pTipoNomina, DateTime pFechaPago, DateTime fechaInicPago, DateTime fechaFinPago, decimal numDiasPag, decimal? totPercepciones, decimal? totDeducciones, decimal? totOtrosPagos)</pre>	<p>Define (establece) el complemento Nómina versión 1.2.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/</p> <p>Excepciones: Exception Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación. ArgumentException Si @pTipoNomina no existe en el catálogo definido por el SAT.</p>
<pre>public ReturnXmlComplemento Nomina12_generaComplemento()</pre>	<p>Genera el complemento Nómina versión 1.2 que es devuelto en el objeto ReturnXmlComplemento, si existe un error no fatal es incluido en el mismo objeto. Ver propiedades:</p> <ul style="list-style-type: none"> ReturnXmlComplemento.IsXmlComplementoCreated ReturnXmlComplemento.ErrorDetalle <p>Excepciones: Exception Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación. Si la instancia complemento Nómina no está definida.</p>
<pre>public void Nomina12_setAccionesOTitulos(int indexPercepcion, decimal pValorMercado,</pre>	<p>Crea y establece el nodo AccionesOTitulos como hijo a la Percepcion apuntada por @indexPercepcion.</p> <p>Xpath:</p>

Métodos complemento	Descripción
<pre>decimal pPrecioAlOtorgarse)</pre>	<p>Cfdi:Complemento/nomina12: Nomina/nomina12:Percepcion[@indexPercepcion]/AccionesOTitulos</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si la Percepcion padre apuntada no existe. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void Nomina12_setCompensacionSaldosAFavor(int indexOtroPago, decimal pSaldoAFavor, short pAño, decimal pRemanenteAFavor)</pre>	<p>Crea y define la CompensacionSaldosAFavor al OtroPago apuntado por @indexOtroPago.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/OtrosPagos/OtroPago [@indexOtroPago]/CompensacionSaldosAFavor</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si OtrosPagos no está definida. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void Nomina12_setDeducciones(decimal? pTotalOtrasDeducc, decimal? pTotalImpuestosReten)</pre>	<p>Crea y define el nodo Deducciones como hijo a nomima.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Deducciones</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2.</p>
<pre>public void Nomina12_setEmisor(string pCarp, string pRegPatronal, string pRfcPatronOrigen)</pre>	<p>Crea y define el nodo Emisor como hijo a nomima.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Emisor</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void Nomina12_setEntidadSNCF(string pOrigenRecurso, decimal pMontoRecursoPropio, bool aplicaMontoRecursoPropio)</pre>	<p>Crea y define el nodo EntidadSNCF como hijo a Emisor.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Emisor/EntidadSNCF</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si no se ha definido la instancia Emisor. ArgumentException</p>

Métodos complemento	Descripción
	Si @pOrigenRecurso no existe en el catálogo definido por el SAT.
<pre>public void Nomina12_setJubilacionPensionReti ro(decimal? pTotalUnaExhibicion, decimal? pTotalParcialidad, decimal? pMontoDiario, decimal pIngresoAcumulable, decimal pIngresoNoAcumulable)</pre>	<p>Crea y define el nodo JubilacionPensionRetiro como hijo a Percepciones.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Percepciones/JubilacionPensionRetiro</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si no se ha definido la instancia nodo Percepciones. ArgumentException Ninguna.</p>
<pre>public void Nomina12_setPercepciones(decimal? pTotalSueldos, decimal? pTotalSeparacionIndemn, decimal? pTotalJubilacionPensionRetiro, decimal pTotalGravado, decimal pTotalExento)</pre>	<p>Crea y define el nodo Percepciones como hijo a nomina.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Percepciones</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. ArgumentException Ninguna.</p>
<pre>public void Nomina12_setReceptor(string pCarp, string pNumSegSocial, DateTime? pFechaInicRelLab, string pAntiguedad, 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>Crea y define el nodo Emisor como hijo a Receptor.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Receptor</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p> <p>ArgumentException Si @pTipoContrato, @pTipoRegimen, @pTipoJornada, @pRiesgo, @pEstado, @pPeriodiciadPago o @pBanco no existe en el catálogo del SAT.</p>
<pre>public void Nomina12_setSeparacionIndemnizaci on(decimal pTotalPagado, int pNumAñosServicio, decimal pUltimoSueldoMensOrd, decimal pIngresoAcumulable, decimal pIngresoNoAcumulable)</pre>	<p>Crea y define el nodo SeparacionIndemnizacion como hijo a Percepciones.</p> <p>Xpath: Cfdi:Complemento/nomina12: Nomina/nomina12:/Percepciones/SeparacionIndemnizacion</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si no se ha definido la instancia nodo Percepciones.</p>

Métodos complemento	Descripción
	Si @pNumAñosServicio es mayor a 99. ArgumentException Ninguna.
<pre>public void Nomina12_setSubsidioAlEmpleo(int indexOtroPago, decimal pSubsidioCausado)</pre>	Crea y define el nodo SubsidioAlEmpleo como hijo a OtroPago . Xpath: Cfdi:Complemento/ nomina12: Nomina /nomina12:/ OtrosPagos/OtroPago/SubsidioAlEmpleo Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Nómina 1.2. Si no se ha definido la instancia nodo OtrosPagos . Si @pNumAñosServicio es mayor a 99. ArgumentException Ninguna.
<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>	Crea y define el nodo DoctoRelacionado como hijo de Pago . Xpath: Cfdi:Complemento/ pagos /pago/ DoctoRelacionado[] Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Pago 1.0. Si no se ha definido la instancia nodo pago . Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación. ArgumentException 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>	Crea y define el nodo Pago como hijo de Pagos . Xpath: Cfdi:Complemento/ pagos /pago[] Returns: Índice del nodo pago que se agregó Excepciones: Exception 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. ArgumentException Si @pFormaDePago, @pMoneda ó @pTipoCadPago no existen en el catálogo del SAT.
<pre>public ReturnXmlComplemento Pagos10_generaComplemento()</pre>	Genera el complemento pago 1.0 que es devuelto en el objeto ReturnXmlComplemento , si existe un error no fatal es incluido en el mismo objeto. Ver propiedades: <ul style="list-style-type: none"> ReturnXmlComplemento.IsXmlComplementoCreated ReturnXmlComplemento.ErrorDetalle

Métodos complemento	Descripción
	Excepciones: Exception 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).
<pre>public ReturnXmlComplemento CartaPorte20_generaComplemento()</pre>	Genera el complemento Carta Porte 2.0 que es devuelto en el objeto ReturnXmlComplemento , si existe un error no fatal es incluido en el mismo objeto. Ver propiedades: <ul style="list-style-type: none"> ReturnXmlComplemento.IsXmlComplementoCreated ReturnXmlComplemento.ErrorDetalle Excepciones: Exception Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
<pre>public void CP20_ADD_CartaPorte_Mercancias_ Autotransporte_Remolques(string SubTipoRem, string Placa)</pre>	Crea y define el nodo Remolques como hijo de Autotransporte . Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:Autotransporte/cartaporte20:Remolques Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, el elemento Mercancias o el elemento Autotransporte. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
<pre>public int CP20_ADD_CartaPorte_Mercancias_ Mercancia(string BienesTransp, string ClaveSTCC, string Descripcion, decimal Cantidad, string ClaveUnidad, string Unidad, string Dimensiones, string MaterialPeligroso, string CveMaterialPeligroso, string Embalaje, string DescripEmbalaje, decimal PesoEnKg, decimal? ValorMercancia, string Moneda, string FraccionArancelaria, string UUIDComercioExt)</pre>	Crea y define el nodo Mercancia como hijo de Mercancias . Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:Mercancia Returns: Índice del nodo Mercancia que se agregó Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, el elemento Mercancias. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
<pre>public void CP20_ADD_CartaPorte_Mercancias_ Mercancia_CantidadTransporta_Ul timaMercancia(decimal pCantidad, string pIDOrigen, string pIDDestino, string pCvesTransporte)</pre>	Crea y define el nodo CantidadTransporta como hijo del último nodo Mercancia . Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:Mercancia/cartaporte20:CantidadTransporta Excepciones:

Métodos complemento	Descripción
	<p>Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, el elemento Mercancías o no existe ningún elemento Mercancía.</p> <p>Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP20_ADD_CartaPorte_Mercancias_ Mercancia_GuiasIdentificacion_U ltimaMercancia(string pNumeroGuiaIdentificacion, string pDescripGuiaIdentificacion, decimal pPesoGuiaIdentificacion)</pre>	<p>Crea y define el nodo GuiasIdentificacion como hijo del último nodo Mercancia.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:Mercancia/cartaporte20:GuiasIdentificacion</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, el elemento Mercancias o no existe ningún elemento Mercancia. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP20_ADD_CartaPorte_Mercancias_ Mercancia_Pedimento_UltimaMerca ncia(string pedim)</pre>	<p>Crea y define el nodo Pedimentos como hijo del último nodo Mercancia.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:Mercancia/cartaporte20:Pedimentos</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, el elemento Mercancias o no existe ningún elemento Mercancia. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public int CP20_ADD_CartaPorte_Mercancias_ TransporteFerroviario_Carro(string TipoCarro, string MatriculaCarro, string GuiaCarro, decimal ToneladasNetasCarro)</pre>	<p>Crea y define el nodo Carro como hijo de TransporteFerroviario.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:TransporteFerroviario/cartaporte20:Carro</p> <p>Returns: Índice del nodo Carro que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, el elemento Mercancias o el elemento TransporteFerroviario. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP20_ADD_CartaPorte_Mercancias_ TransporteFerroviario_Carro_Con tenedor_UltimoCarro(string TipoContenedor, decimal PesoContenedorVacio, decimal PesoNetoMercancia)</pre>	<p>Crea y define el nodo Contenedor como hijo del último nodo Carro.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:TransporteFerroviario/cartaporte20:Carro/cartaporte20:Contenedor</p> <p>Excepciones:</p>

Métodos complemento	Descripción
	<p>Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, el elemento Mercancias, elemento TransporteFerroviario, o no existe ningún elemento Carro.</p> <p>Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP20_ADD_CartaPorte_Mercancias_ TransporteFerroviario_DerechosD ePaso(string TipoDerechoDePaso, decimal KilometrajePagado)</pre>	<p>Crea y define el nodo DerechosDePaso como hijo del nodo TransporteFerroviario.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:TransporteFerroviario/cartaporte20:DerechosDePaso</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, el elemento Mercancias, o el elemento TransporteFerroviario. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP20_ADD_CartaPorte_Mercancias_ TransporteMaritimo_Contenedor(s tring MatriculaContenedor, string TipoContenedor, string NumPrecinto)</pre>	<p>Crea y define el nodo Contenedor como hijo del nodo TransporteMaritimo.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:TransporteMaritimo/cartaporte20:Contenedor</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, el elemento Mercancias, o el elemento TransporteMaritimo. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP20_ADD_CartaPorte_TiposFigura (string TipoFigura, string RFCFigura, string NumLicencia, string NombreFigura, string NumRegIdTribFigura, string ResidenciaFiscalFigura, CartaPorte10_Domicilio domicilio)</pre>	<p>Crea y define el nodo TiposFigura como hijo del nodo FiguraTransporte. Si se incluye el parámetro @domicilio, se crea el nodo Domicilio como hijo del nodo TiposFigura.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:FiguraTransporte/cartaporte20:TiposFigura</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP20_ADD_CartaPorte_TiposFigura _PartesTransporte_UltimoTipoFig ura(string partesTransp)</pre>	<p>Crea y define el nodo PartesTransporte como hijo del nodo TiposFigura.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:FiguraTransporte/cartaporte20:TiposFigura/cartaporte20:PartesTransporte</p> <p>Excepciones:</p>

Métodos complemento	Descripción
	Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0 o el elemento FiguraTransporte/TiposFigura. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
<pre>public int CP20_ADD_CartaPorte_Ubicacion(string TipoUbicacion, string IDUbicacion, string RFCRemitenteDestinatario, string NombreRemitenteDestinatario, string NumRegIdTrib, string ResidenciaFiscal, string NumEstacion, string NombreEstacion, string NavegacionTrafico, DateTime FechaHoraSalidaLlegada, string TipoEstacion, decimal? DistanciaRecorrida)</pre>	Crea y define el nodo Ubicacion como hijo del nodo Ubicaciones . Xpath: /cartaporte20:CartaPorte/cartaporte20:Ubicaciones/cartaport e20: Ubicacion Returns: Índice del nodo Ubicacion que se agregó Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
<pre>public void CP20_REMOVEALL_CartaPorte_Merca ncias_Autotransporte_Remolques()</pre>	Remueve todos los elementos Remolques del nodo Autotransporte . Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2, el elemento Mercancias o el elemento Autotransporte.
<pre>public void CP20_SET_CartaPorte(string TranspInternac, string EntradaSalidaMerc, string PaisOrigenDest, string ViaEntradaSalida, decimal? TotalDistRec)</pre>	Crea y define el nodo CartaPorte . Xpath: Cfdi:Complemento/ cartaporte20:CartaPorte
<pre>public void CP20_SET_CartaPorte_Mercancias(decimal PesoBrutoTotal, string UnidadPeso, decimal? PesoNetoTotal, int NumTotalMercancias, decimal? CargoPorTasacion)</pre>	Crea y define el nodo Mercancias como hijo del nodo CartaPorte . Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
<pre>public void CP20_SET_CartaPorte_Mercancias_ Autotransporte(string PermSCT, string NumPermisoSCT, string ConfigVehicular, string PlacaVM, int AnioModeloVM)</pre>	Crea y define el nodo Autotransporte como hijo del nodo Mercancias . y a su vez el nodo IdentificacionVehicular como hijo del nodo Autotransporte . Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaport e20:Autotransporte Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0 o el elemento Mercancias.

Métodos complemento	Descripción
	Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
<pre>public void CP20_SET_CartaPorte_Mercancias_ Autotransporte_Seguros(string AseguraRespCivil, string PolizaRespCivil, string AseguraMedAmbiente, string PolizaMedAmbiente, string AseguraCarga, string PolizaCarga, decimal? PrimaSeguro)</pre>	<p>Crea y define el nodo Seguros como hijo del nodo Autotransporte.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:Autotransporte/cartaporte20:Seguros</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, el elemento Mercancias o el elemento Autotransporte. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP20_SET_CartaPorte_Mercancias_ Mercancia_DetalleMercancia_Ulti maMercancia(string pUnidadPesoMerc, decimal pPesoBruto, decimal pPesoNeto, decimal pPesoTara, int? pNumPiezas)</pre>	<p>Crea y define el nodo DetalleMercancia como hijo del ultimo nodo Mercancia.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:Mercancia/cartaporte20:DetalleMercancia</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, el elemento Mercancias o al menos un elemento Mercancia. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP20_SET_CartaPorte_Mercancias_ TransporteAereo(string PermSCT, string NumPermisoSCT, string MatriculaAeronave, string NombreAseg, string NumPolizaSeguro, string NumeroGuia, string LugarContrato, string CodigoTransportista, string RFCEmbarcador, string NumRegIdTribEmbarc, string ResidenciaFiscalEmbarc, string NombreEmbarcador)</pre>	<p>Crea y define el nodo TransporteAereo como hijo del nodo Mercancias.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:TransporteAereo</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, o el elemento Mercancias. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP20_SET_CartaPorte_Mercancias_ TransporteFerroviario(string TipoDeServicio, string TipoDeTrafico, string NombreAseg, string NumPolizaSeguro)</pre>	<p>Crea y define el nodo TransporteFerroviario como hijo del nodo Mercancias.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:TransporteFerroviario</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, o el elemento Mercancias. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>

Métodos complemento	Descripción
<pre>public void CP20_SET_CartaPorte_Mercancias_ TransporteMaritimo(string PermSCT, string NumPermisoSCT, string NombreAseg, string NumPolizaSeguro, string TipoEmbarcacion, string Matricula, string NumeroOMI, int? AnioEmbarcacion, string NombreEmbarc, string NacionalidadEmbarc, decimal UnidadesDeArqBruto, string TipoCarga, string NumCertITC, decimal? Eslora, decimal? Manga, decimal? Calado, string LineaNaviera, string NombreAgenteNaviero, string NumAutorizacionNaviero, string NumViaje, string NumConocEmbarc, string MatriculaContenedor, string TipoContenedor, string NumPrecinto)</pre>	<p>Crea y define el nodo TransporteMaritimo como hijo del nodo Mercancias.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:Mercancias/cartaporte20:TransporteMaritimo</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, o el elemento Mercancias. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP20_SET_CartaPorte_Ubicacion_D omicilio(int indexUbicacion, CartaPorte10_Domicilio domic)</pre>	<p>Crea y define el nodo Domicilio como hijo del nodo Ubicacion.</p> <p>Xpath: /cartaporte20:CartaPorte/cartaporte20:Ubicaciones/cartaport e20:Ubicacion/cartaporte20:Domicilio</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 2.0, o no existe el nodo Ubicacion en el índice especificado por @indexUbicacion. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public ReturnXmlComplemento CartaPorte30_generaComplemento()</pre>	<p>Genera el complemento Carta Porte 3.0 que es devuelto en el objeto ReturnXmlComplemento, si existe un error no fatal es incluido en el mismo objeto. Ver propiedades:</p> <ul style="list-style-type: none"> ReturnXmlComplemento.IsXmlComplementoCreated ReturnXmlComplemento.ErrorDetalle <p>Excepciones: Exception Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP30_ADD_CartaPorte_Mercancias_ Autotransporte_Remolques(string SubTipoRem, string Placa)</pre>	<p>Crea y define el nodo Remolques como hijo de Autotransporte.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:Autotransporte/cartaporte30:Remolques</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias o el elemento Autotransporte.</p>

Métodos complemento	Descripción
	Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
<pre>public int CP30_ADD_CartaPorte_Mercancias_ Mercancia(string BienesTransp, string ClaveSTCC, string Descripcion, decimal Cantidad, string ClaveUnidad, string Unidad, string Dimensiones, string MaterialPeligroso, string CveMaterialPeligroso, string Embalaje, string DescripEmbalaje, string SectorCOFEPRIS, string NombreIngredienteActivo, string NomQuimico, string DenominacionGenericaProd, string DenominacionDistintivaProd, string Fabricante, DateTime? FechaCaducidad, string LoteMedicamento, string FormaFarmaceutica, string CondicionesEspTransp, string RegistroSanitarioFolioAutorizac ion, string PermisoImportacion, string FolioImpoVUCEM, string NumCAS, string RazonSocialEmpImp, string NumRegSanPlagCOFEPRIS, string DatosFabricante, string DatosFormulador, string DatosMaquilador, string UsoAutorizado, decimal PesoEnKg, decimal? ValorMercancia, string Moneda, string FraccionArancelaria, string UUIDComercioExt, string TipoMateria, string DescripcionMateria)</pre>	<p>Crea y define el nodo Mercancia como hijo de Mercancias.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:Mercancia</p> <p>Returns: Índice del nodo Mercancia que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP30_ADD_CartaPorte_Mercancias_ Mercancia_CantidadTransporta_Ul timaMercancia(decimal pCantidad, string pIDOrigen, string pIDDestino, string pCvesTransporte)</pre>	<p>Crea y define el nodo CantidadTransporta como hijo del último nodo Mercancia.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:Mercancia/cartaporte30:CantidadTransporta</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias o no existe ningún elemento Mercancia. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>

Métodos complemento	Descripción
<pre>public void CP30_ADD_CartaPorte_Mercancias_ mercancia_DocumentacionAduanera _UltimaMercancia(string TipoDocumento, string NumPedimento, string IdentDocAduanero, string RFCImpo)</pre>	<p>Crea y define el nodo DocumentacionAduanera como hijo del último nodo Mercancia.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:Mercancia/cartaporte30: DocumentacionAduanera</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias o no existe ningún elemento Mercancia. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP30_ADD_CartaPorte_Mercancias_ Mercancia_GuiasIdentificacion_U ltimaMercancia(string pNumeroGuiaIdentificacion, string pDescripGuiaIdentificacion, decimal pPesoGuiaIdentificacion)</pre>	<p>Crea y define el nodo GuiasIdentificacion como hijo del último nodo Mercancia.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:Mercancia/cartaporte30:GuiasIdentificacion</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias o no existe ningún elemento Mercancia. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public int CP30_ADD_CartaPorte_Mercancias_ TransporteFerroviario_Carro(string TipoCarro, string MatriculaCarro, string GuiaCarro, decimal ToneladasNetasCarro)</pre>	<p>Crea y define el nodo Carro como hijo de TransporteFerroviario.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:TransporteFerroviario/cartaporte30:Carro</p> <p>Returns: Índice del nodo Carro que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias o el elemento TransporteFerroviario. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP30_ADD_CartaPorte_Mercancias_ TransporteFerroviario_Carro_Con tenedor_UltimoCarro(string TipoContenedor, decimal PesoContenedorVacio, decimal PesoNetoMercancia)</pre>	<p>Crea y define el nodo Contenedor como hijo del último nodo Carro.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:TransporteFerroviario/cartaporte30:Carro/cartaporte30:Contenedor</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias, elemento TransporteFerroviario, o no existe ningún elemento Carro. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>

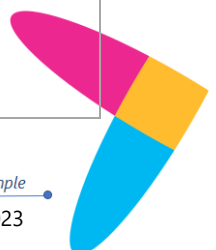
Métodos complemento	Descripción
<pre>public void CP30_ADD_CartaPorte_Mercancias_ TransporteFerroviario_DerechosD ePaso(string TipoDerechoDePaso, decimal KilometrajePagado)</pre>	<p>Crea y define el nodo DerechosDePaso como hijo del nodo TransporteFerroviario.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:TransporteFerroviario/cartaporte30:DerechosDePaso</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias, o el elemento TransporteFerroviario. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP30_ADD_CartaPorte_Mercancias_ TransporteMaritimo_Contenedor(s tring TipoContenedor, string MatriculaContenedor, string NumPrecinto, string IdCCPRelacionado, string PlacaVMCCP, DateTime? FechaCertificacionCCP)</pre>	<p>Crea y define el nodo Contenedor como hijo del nodo TransporteMaritimo.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:TransporteMaritimo/cartaporte30:Contenedor</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias, o el elemento TransporteMaritimo. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP30_ADD_CartaPorte_Mercancias_ TransporteMaritimo_RemolqueCCP(string SubTipoRemCCP, string PlacaCCP)</pre>	<p>Crea y define el nodo RemolquesCCP como hijo del nodo TransporteMaritimo.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:TransporteMaritimo/cartaporte30: RemolquesCCP</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias, o el elemento TransporteMaritimo. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP30_ADD_CartaPorte_TiposFigura (string TipoFigura, string RFCFigura, string NumLicencia, string NombreFigura, string NumRegIdTribFigura, string ResidenciaFiscalFigura, CartaPorte10_Domicilio domicilio)</pre>	<p>Crea y define el nodo TiposFigura como hijo del nodo FiguraTransporte. Si se incluye el parámetro @domicilio, se crea el nodo Domicilio como hijo del nodo TiposFigura.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:FiguraTransporte/cartaporte30:TiposFigura</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>

Métodos complemento	Descripción
<pre>public void CP30_ADD_CartaPorte_TiposFigura _PartesTransporte_UltimoTipoFigura(string partesTransp)</pre>	<p>Crea y define el nodo PartesTransporte como hijo del nodo TiposFigura.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:FiguraTransporte/cartaporte30:TiposFigura/cartaporte30:PartesTransporte</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0 o el elemento FiguraTransporte/TiposFigura. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public int CP30_ADD_CartaPorte_Ubicacion(string TipoUbicacion, string IDUbicacion, string RFCRemitenteDestinatario, string NombreRemitenteDestinatario, string NumRegIdTrib, string ResidenciaFiscal, string NumEstacion, string NombreEstacion, string NavegacionTrafico, DateTime FechaHoraSalidaLlegada, string TipoEstacion, decimal? DistanciaRecorrida)</pre>	<p>Crea y define el nodo Ubicacion como hijo del nodo Ubicaciones.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:Ubicaciones/cartaporte30:Ubicacion</p> <p>Returns: Índice del nodo Ubicacion que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP30_REMOVEALL_CartaPorte_Mercancias_Autotransporte_Remolques()</pre>	<p>Remueve todos los elementos Remolques del nodo Autotransporte.</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias o el elemento Autotransporte.</p>
<pre>public void CP30_REMOVEALL_CartaPorte_Mercancias_TransporteMaritimo_RemolquesCCP()</pre>	<p>Remueve todos los elementos RemolquesCCP del nodo Autotransporte.</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias o el elemento Autotransporte.</p>
<pre>public void CP30_SET_CartaPorte(string IdCCP, string TranspInternac, string RegimenAduanero, string EntradaSalidaMerc, string PaisOrigenDest, string ViaEntradaSalida, decimal? TotalDistRec, string RegistroISTMO, string UbicacionPoloOrigen, string UbicacionPoloDestino)</pre>	<p>Crea y define el nodo CartaPorte.</p> <p>Xpath: Cfdi:Complemento/cartaporte30:CartaPorte</p>
<pre>public void CP30_SET_CartaPorte_Mercancias(decimal PesoBrutoTotal, string UnidadPeso, decimal? PesoNetoTotal, int)</pre>	<p>Crea y define el nodo Mercancias como hijo del nodo CartaPorte.</p> <p>Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias</p>

Métodos complemento	Descripción
NumTotalMercancias, decimal? CargoPorTasacion, string LogisticaInversaRecoleccionDevo lucion)	Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
public void CP30_SET_CartaPorte_Mercancias_ Autotransporte(tring PermSCT, string NumPermisoSCT, string ConfigVehicular, decimal PesoBrutoVehicular, string PlacaVM, int AnioModeloVM)	Crea y define el nodo Autotransporte como hijo del nodo Mercancias . y a su vez el nodo IdentificacionVehicular como hijo del nodo Autotransporte . Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:Autotransporte Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0 o el elemento Mercancias. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
public void CP30_SET_CartaPorte_Mercancias_ Autotransporte_Seguros(string AseguraRespCivil, string PolizaRespCivil, string AseguraMedAmbiente, string PolizaMedAmbiente, string AseguraCarga, string PolizaCarga, decimal? PrimaSeguro)	Crea y define el nodo Seguros como hijo del nodo Autotransporte . Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:Autotransporte/cartaporte30:Seguros Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias o el elemento Autotransporte. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
public void CP30_SET_CartaPorte_Mercancias_ Mercancia_DetalleMercancia_Ulti maMercancia(string pUnidadPesoMerc, decimal pPesoBruto, decimal pPesoNeto, decimal pPesoTara, int? pNumPiezas)	Crea y define el nodo DetalleMercancia como hijo del ultimo nodo Mercancia . Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:Mercancia/cartaporte30:DetalleMercancia Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, el elemento Mercancias o al menos un elemento Mercancia. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
public void CP30_SET_CartaPorte_Mercancias_ TransporteAereo(string PermSCT, string NumPermisoSCT, string MatriculaAeronave, string NombreAseg, string NumPolizaSeguro, string NumeroGuia, string LugarContrato, string CodigoTransportista, string RFCEmbarcador, string	Crea y define el nodo TransporteAereo como hijo del nodo Mercancias . Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:TransporteAereo Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, o el elemento Mercancias.

Métodos complemento	Descripción
NumRegIdTribEmbarc, string ResidenciaFiscalEmbarc, string NombreEmbarcador)	Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
public void CP30_SET_CartaPorte_Mercancias_TransporteFerroviario(string TipoDeServicio, string TipoDeTrafico, string NombreAseg, string NumPolizaSeguro)	Crea y define el nodo TransporteFerroviario como hijo del nodo Mercancias . Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:TransporteFerroviario Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, o el elemento Mercancias. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
public void CP30_SET_CartaPorte_Mercancias_TransporteMaritimo(string PermSCT, string NumPermisoSCT, string NombreAseg, string NumPolizaSeguro, string TipoEmbarcacion, string Matricula, string NumeroOMI, int? AnioEmbarcacion, string NombreEmbarc, string NacionalidadEmbarc, decimal UnidadesDeArqBruto, string TipoCarga, decimal? Eslora, decimal? Manga, decimal? Calado, decimal? Puntal, string LineaNaviera, string NombreAgenteNaviero, string NumAutorizacionNaviero, string NumViaje, string NumConocEmbarc, string PermisoTempNavegacion)	Crea y define el nodo TransporteMaritimo como hijo del nodo Mercancias . Xpath: /cartaporte30:CartaPorte/cartaporte30:Mercancias/cartaporte30:TransporteMaritimo Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, o el elemento Mercancias. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
public void CP30_SET_CartaPorte_Ubicacion_Domicilio(int indexUbicacion, CartaPorte10_Domicilio domic)	Crea y define el nodo Domicilio como hijo del nodo Ubicacion . Xpath: /cartaporte30:CartaPorte/cartaporte30:Ubicaciones/cartaporte30:Ubicacion/cartaporte30:Domicilio Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.0, o no existe el nodo Ubicacion en el índice especificado por @indexUbicacion. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.
public ReturnXmlComplemento CartaPorte31_generaComplemento()	Genera el complemento Carta Porte 3.1 que es devuelto en el objeto ReturnXmlComplemento , si existe un error no fatal es incluido en el mismo objeto. Ver propiedades: <ul style="list-style-type: none"> ReturnXmlComplemento.IsXmlComplementoCreated ReturnXmlComplemento.ErrorDetalle

Métodos complemento	Descripción
	<p>Excepciones: Exception Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP31_ADD_CartaPorte_Mercancias_ Autotransporte_Remolques(string SubTipoRem, string Placa)</pre>	<p>Crea y define el nodo Remolques como hijo de Autotransporte.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:Autotransporte/cartaporte31:Remolques</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancias o el elemento Autotransporte. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public int CP31_ADD_CartaPorte_Mercancias_ Mercancia(string BienesTransp, string ClaveSTCC, string Descripcion, decimal Cantidad, string ClaveUnidad, string Unidad, string Dimensiones, string MaterialPeligroso, string CveMaterialPeligroso, string Embalaje, string DescripEmbalaje, string SectorCOFEPRIS, string NombreIngredienteActivo, string NomQuimico, string DenominacionGenericaProd, string DenominacionDistintivaProd, string Fabricante, DateTime? FechaCaducidad, string LoteMedicamento, string FormaFarmaceutica, string CondicionesEspTransp, string RegistroSanitarioFolioAutorizac ion, string PermisoImportacion, string FolioImpoVUCEM, string NumCAS, string RazonSocialEmpImp, string NumRegSanPlagCOFEPRIS, string DatosFabricante, string DatosFormulador, string DatosMaquilador, string UsoAutorizado, decimal PesoEnKg, decimal? ValorMercancia, string Moneda, string FraccionArancelaria, string UUIDComercioExt, string TipoMateria, string DescripcionMateria)</pre>	<p>Crea y define el nodo Mercancia como hijo de Mercancias.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:Mercancia</p> <p>Returns: Índice del nodo Mercancia que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancias. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>



Métodos complemento	Descripción
<pre>public void CP31_ADD_CartaPorte_Mercancias_ Mercancia_CantidadTransporta_Ul timaMercancia(decimal pCantidad, string pIDOrigen, string pIDDestino, string pCvesTransporte)</pre>	<p>Crea y define el nodo CantidadTransporta como hijo del último nodo Mercancia.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:Mercancia/cartaporte31:CantidadTransporta</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancias o no existe ningún elemento Mercancia. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP31_ADD_CartaPorte_Mercancias_ mercancia_DocumentacionAduanera_ UltimaMercancia(string TipoDocumento, string NumPedimento, string IdentDocAduanero, string RFCImpo)</pre>	<p>Crea y define el nodo DocumentacionAduanera como hijo del último nodo Mercancia.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:Mercancia/cartaporte31: DocumentacionAduanera</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancias o no existe ningún elemento Mercancia. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP31_ADD_CartaPorte_Mercancias_ Mercancia_GuiasIdentificacion_U ltimaMercancia(string pNumeroGuiaIdentificacion, string pDescripGuiaIdentificacion, decimal pPesoGuiaIdentificacion)</pre>	<p>Crea y define el nodo GuiasIdentificacion como hijo del último nodo Mercancia.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:Mercancia/cartaporte31:GuiasIdentificacion</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancias o no existe ningún elemento Mercancia. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public int CP31_ADD_CartaPorte_Mercancias_ TransporteFerroviario_Carro(str ing TipoCarro, string MatriculaCarro, string GuiaCarro, decimal ToneladasNetasCarro)</pre>	<p>Crea y define el nodo Carro como hijo de TransporteFerroviario.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:TransporteFerroviario/cartaporte31:Carro</p> <p>Returns: Índice del nodo Carro que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancias o el elemento TransporteFerroviario. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>

Métodos complemento	Descripción
<pre>public void CP31_ADD_CartaPorte_Mercancias_ TransporteFerroviario_Carro_Con tenedor_UltimoCarro(string TipoContenedor, decimal PesoContenedorVacio, decimal PesoNetoMercancia)</pre>	<p>Crea y define el nodo Contenedor como hijo del último nodo Carro.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:TransporteFerroviario/cartaporte31:Carro/cartaporte31:Contenedor</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancias, elemento TransporteFerroviario, o no existe ningún elemento Carro. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP31_ADD_CartaPorte_Mercancias_ TransporteFerroviario_DerechosD ePaso(string TipoDerechoDePaso, decimal KilometrajePagado)</pre>	<p>Crea y define el nodo DerechosDePaso como hijo del nodo TransporteFerroviario.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:TransporteFerroviario/cartaporte31:DerechosDePaso</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancias, o el elemento TransporteFerroviario. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP31_ADD_CartaPorte_Mercancias_ TransporteMaritimo_Contenedor(s tring TipoContenedor, string MatriculaContenedor, string NumPrecinto, string IdCCPRelacionado, string PlacaVMCCP, DateTime? FechaCertificacionCCP)</pre>	<p>Crea y define el nodo Contenedor como hijo del nodo TransporteMaritimo.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:TransporteMaritimo/cartaporte31:Contenedor</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancias, o el elemento TransporteMaritimo. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP31_ADD_CartaPorte_Mercancias_ TransporteMaritimo_Contenedor_R emolqueCCP_UltimoContenedor (string SubTipoRemCCP, string PlacaCCP)</pre>	<p>Crea y define el nodo RemolquesCCP como hijo del nodo TransporteMaritimo.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:TransporteMaritimo /cartaporte31:Contenedor/cartaporte31:RemolquesCCP</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancias, o el elemento TransporteMaritimo. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>

Métodos complemento	Descripción
<pre>public void CP31_ADD_CartaPorte_RegimenAdua neroCCP (string RegimenAduanero)</pre>	<p>Crea y define el nodo RegimenAduaneroCCP como hijo del nodo RegimenesAduaneros.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:RegimenesAduaneros/ca rtaporte31:RegimenAduaneroCCP</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1 o el parámetro RegimenAduanero es nulo o vacío. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP31_ADD_CartaPorte_TiposFigura (string TipoFigura, string RFCFigura, string NumLicencia, string NombreFigura, string NumRegIdTribFigura, string ResidenciaFiscalFigura, CartaPorte10_Domicilio domicilio)</pre>	<p>Crea y define el nodo TiposFigura como hijo del nodo FiguraTransporte. Si se incluye el parámetro @domicilio, se crea el nodo Domicilio como hijo del nodo TiposFigura.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:FiguraTransporte/cart aporte31:TiposFigura</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP31_ADD_CartaPorte_TiposFigura _PartesTransporte_UltimoTipoFig ura(string partesTransp)</pre>	<p>Crea y define el nodo PartesTransporte como hijo del nodo TiposFigura.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:FiguraTransporte/cart aporte31:TiposFigura/cartaporte31:PartesTransporte</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1 o el elemento FiguraTransporte/TiposFigura. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public int CP31_ADD_CartaPorte_Ubicacion(s tring TipoUbicacion, string IDUbicacion, string RFCRemitenteDestinatario, string NombreRemitenteDestinatario, string NumRegIdTrib, string ResidenciaFiscal, string NumEstacion, string NombreEstacion, string NavegacionTrafico, DateTime FechaHoraSalidaLlegada, string TipoEstacion, decimal? DistanciaRecorrida)</pre>	<p>Crea y define el nodo Ubicacion como hijo del nodo Ubicaciones.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Ubicaciones/cartaport e31: Ubicacion</p> <p>Returns: Índice del nodo Ubicacion que se agregó</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>

Métodos complemento	Descripción
<pre>public void CP31_REMOVEALL_CartaPorte_Merca ncias_Autotransporte_Remolques()</pre>	<p>Remueve todos los elementos Remolques del nodo Autotransporte.</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancías o el elemento Autotransporte.</p>
<pre>public void CP31_SET_CartaPorte(string IdCCP, string TranspInternac, string EntradaSalidaMerc, string PaisOrigenDest, string ViaEntradaSalida, decimal? TotalDistRec, string RegistroISTMO, string UbicacionPoloOrigen, string UbicacionPoloDestino)</pre>	<p>Crea y define el nodo CartaPorte.</p> <p>Xpath: cfdi:Complemento/cartaporte31:CartaPorte</p>
<pre>public void CP31_SET_CartaPorte_Mercancias(decimal PesoBrutoTotal, string UnidadPeso, decimal? PesoNetoTotal, int NumTotalMercancias, decimal? CargoPorTasacion, string LogisticaInversaRecoleccionDevo lucion)</pre>	<p>Crea y define el nodo Mercancias como hijo del nodo CartaPorte.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP31_SET_CartaPorte_Mercancias_ Autotransporte(triing PermSCT, string NumPermisoSCT, string ConfigVehicular, decimal PesoBrutoVehicular, string PlacaVM, int AnioModeloVM)</pre>	<p>Crea y define el nodo Autotransporte como hijo del nodo Mercancias. y a su vez el nodo IdentificacionVehicular como hijo del nodo Autotransporte.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:Autotransporte</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1 o el elemento Mercancías. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP31_SET_CartaPorte_Mercancias_ Autotransporte_Seguros(string AseguraRespCivil, string PolizaRespCivil, string AseguraMedAmbiente, string PolizaMedAmbiente, string AseguraCarga, string PolizaCarga, decimal? PrimaSeguro)</pre>	<p>Crea y define el nodo Seguros como hijo del nodo Autotransporte.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:Autotransporte/cartaporte31:Seguros</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancías o el elemento Autotransporte. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<pre>public void CP31_SET_CartaPorte_Mercancias_ Mercancia_DetalleMercancia_Ulti maMercancia(string</pre>	<p>Crea y define el nodo DetalleMercancia como hijo del ultimo nodo Mercancia.</p> <p>Xpath:</p>

Métodos complemento	Descripción
<p>pUnidadPesoMerc, decimal pPesoBruto, decimal pPesoNeto, decimal pPesoTara, int? pNumPiezas)</p>	<p>/cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:Mercancia/cartaporte31:DetalleMercancia</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, el elemento Mercancias o al menos un elemento Mercancia. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<p>public void CP31_SET_CartaPorte_Mercancias_TransporteAereo(string PermSCT, string NumPermisoSCT, string MatriculaAeronave, string NombreAseg, string NumPolizaSeguro, string NumeroGuia, string LugarContrato, string CodigoTransportista, string RFCEmbarcador, string NumRegIdTribEmbarc, string ResidenciaFiscalEmbarc, string NombreEmbarcador)</p>	<p>Crea y define el nodo TransporteAereo como hijo del nodo Mercancias.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:TransporteAereo</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, o el elemento Mercancias. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<p>public void CP31_SET_CartaPorte_Mercancias_TransporteFerroviario(string TipoDeServicio, string TipoDeTrafico, string NombreAseg, string NumPolizaSeguro)</p>	<p>Crea y define el nodo TransporteFerroviario como hijo del nodo Mercancias.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:TransporteFerroviario</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, o el elemento Mercancias. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<p>public void CP31_SET_CartaPorte_Mercancias_TransporteMaritimo(string PermSCT, string NumPermisoSCT, string NombreAseg, string NumPolizaSeguro, string TipoEmbarcacion, string Matricula, string NumeroOMI, int? AnioEmbarcacion, string NombreEmbarc, string NacionalidadEmbarc, decimal UnidadesDeArqBruto, string TipoCarga, decimal? Eslora, decimal? Manga, decimal? Calado, decimal? Puntal, string LineaNaviera, string NombreAgenteNaviero, string NumAutorizacionNaviero, string NumViaje, string</p>	<p>Crea y define el nodo TransporteMaritimo como hijo del nodo Mercancias.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Mercancias/cartaporte31:TransporteMaritimo</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, o el elemento Mercancias. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>

Métodos complemento	Descripción
NumConocEmbarc, <code>string</code> PermisoTempNavegacion)	
<code>public void</code> CP31_SET_CartaPorte_Ubicacion_D omicilio(<code>int</code> indexUbicacion, <code>CartaPorte10_Domicilio</code> domic)	<p>Crea y define el nodo Domicilio como hijo del nodo Ubicacion.</p> <p>Xpath: /cartaporte31:CartaPorte/cartaporte31:Ubicaciones/cartaport e31:Ubicacion/cartaporte31:Domicilio</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Carta Porte 3.1, o no existe el nodo Ubicacion en el índice especificado por @indexUbicacion. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<code>public ReturnXmlComplemento</code> Donat11_generaComplemento(<code>string</code> noAutorizacion, <code>DateTime</code> fechaAutorizacion, <code>string</code> leyenda)	<p>Genera el complemento de Donatarias 1.1. El complemento viene la propiedad XMLComplemento de ReturnXmlComplemento.</p> <p>Excepciones: Exception Esta excepción ocurre si los parámetros no cumplen la especificación del SAT en formato o restricción.</p>
<code>public void</code> PAGOS20_ADD_Pagos_Pago(<code>DateTime</code> pFechaPago, <code>string</code> pFormaDePago, <code>string</code> pMoneda, <code>decimal?</code> pTipoCambio, <code>decimal</code> pMonto, <code>string</code> pNumOperacion, <code>string</code> pRfcEmisorCtaOrd, <code>string</code> pNomBancoOrdExt, <code>string</code> pCtaOrdenante, <code>string</code> pRfcEmisorCtaBen, <code>string</code> pCtaBeneficiario, <code>string</code> pTipoCadPago, <code>byte[]</code> pCertPago, <code>string</code> pCadPago, <code>byte[]</code> pSelloPago)	<p>Crea y define el nodo Pago como hijo del nodo Pagos.</p> <p>Xpath: /pago20:Pagos/pago20:Pago</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Pagos 2.0. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<code>public void</code> PAGOS20_ADD_Pagos_Pago_DoctoRel acionado_RetencionDR_UltimoDoct oRelacionado(<code>decimal</code> pBaseDR, <code>string</code> pImpuestoDR, <code>string</code> pTipoFactorDR, <code>decimal</code> pTasaOCuotaDR, <code>decimal</code> pImporteDR)	<p>Crea y define el nodo RetencionDR como hijo del nodo RetencionesDR en el último nodo DoctoRelacionado.</p> <p>Xpath: /pago20:Pagos/pago20:Pago/pago20:DoctoRelacionado/pago20:Im puestosDR/pago20:RetencionesDR/pago20:RetencionDR</p> <p>Excepciones: Exception Esta excepción ocurre si no se ha definido el complemento Pagos 2.0, el elemento Pago o no existe ningún elemento DoctoRelacionado. Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
<code>public void</code> PAGOS20_ADD_Pagos_Pago_DoctoRel acionado_TransladoDR_UltimoDocto Relacionado(<code>decimal</code> pBaseDR, <code>string</code> pImpuestoDR, <code>string</code> pTipoFactorDR, <code>decimal?</code>	<p>Crea y define el nodo TrasladoDR como hijo del nodo TrasladosDR en el último nodo DoctoRelacionado.</p> <p>Xpath: /pago20:Pagos/pago20:Pago/pago20:DoctoRelacionado/pago20:Im puestosDR/pago20:TrasladosDR/pago20:TrasladoDR</p>

Métodos complemento	Descripción
pTasaOCuotaDR, decimal? pImporteDR)	<p>Excepciones:</p> <p>Exception Esta excepción ocurre si no se ha definido el complemento Pagos 2.0, el elemento Pago o no existe ningún elemento DoctoRelacionado.</p> <p>Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
public void PAGOS20_ADD_Pagos_Pago_DoctoRelacionado_UltimoPago(string pIdDocumento, string pSerie, string pFolio, string pMonedaDR, decimal? pEquivalenciaDR, int pNumParcialidad, decimal pImpSaldoAnt, decimal pImpPagado, decimal pImpSaldoInsoluto, string pObjetoImpDR)	<p>Crea y define el nodo DoctoRelacionado como hijo del último nodo Pago.</p> <p>Xpath: /pago20:Pagos/pago20:Pago/pago20:DoctoRelacionado</p> <p>Excepciones:</p> <p>Exception Esta excepción ocurre si no se ha definido el complemento Pagos 2.0, o el elemento Pago.</p> <p>Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
public void PAGOS20_ADD_Pagos_Pago_RetencionP_UltimoPago(string pImpuestoP, decimal ImporteP)	<p>Crea y define el nodo RetencionP como hijo del nodo RetencionesP del último nodo Pago.</p> <p>Xpath: /pago20:Pagos/pago20:Pago/pago20:ImpuestosP/pago20:RetencionesP/pago20:RetencionP</p> <p>Excepciones:</p> <p>Exception Esta excepción ocurre si no se ha definido el complemento Pagos 2.0, o el elemento Pago.</p> <p>Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
public void PAGOS20_ADD_Pagos_Pago_TransladoP_UltimoPago(decimal pBaseP, string pImpuestoP, string pTipoFactorP, decimal? pTasaOCuotaP, decimal? pImporteP)	<p>Crea y define el nodo TrasladoP como hijo del nodo TrasladosP del último nodo Pago.</p> <p>Xpath: /pago20:Pagos/pago20:Pago/pago20:ImpuestosP/pago20:TrasladosP/pago20:TrasladoP</p> <p>Excepciones:</p> <p>Exception Esta excepción ocurre si no se ha definido el complemento Pagos 2.0, o el elemento Pago.</p> <p>Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>
public ReturnXmlComplemento Pagos20_generaComplemento()	<p>Genera el complemento Pagos 2.0 que es devuelto en el objeto ReturnXmlComplemento, si existe un error no fatal es incluido en el mismo objeto. Ver propiedades:</p> <ul style="list-style-type: none"> ReturnXmlComplemento.IsXmlComplementoCreated ReturnXmlComplemento.ErrorDetalle <p>Excepciones:</p> <p>Exception</p> <p>Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</p>

Métodos complemento	Descripción
<pre>public void PAGOS20_SET_Pagos(decimal? pTotalRetencionesIVA, decimal? pTotalRetencionesISR, decimal? pTotalRetencionesIEPS, decimal? pTotalTrasladosBaseIVA16, decimal? pTotalTrasladosImpuestoIVA16, decimal? pTotalTrasladosBaseIVA8, decimal? pTotalTrasladosImpuestoIVA8, decimal? pTotalTrasladosBaseIVA0, decimal? pTotalTrasladosImpuestoIVA0, decimal? pTotalTrasladosBaseIVAExento, decimal pMontoTotalPagos)</pre>	<p>Crea y define el nodo Pagos y el nodo Totales como hijo del nodo Pagos.</p> <p>Xpath: /pago20:Pagos/pago20:Totales</p> <p>Excepciones: Exception Si los parámetros no cumplen con las especificaciones definidas por el SAT en formato o combinación.</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 " Matriz de Errores " del SAT.
<code>public XmlDocument XML { get; set; }</code>	Contendrá el comprobante en XML generado o un complemento. Esto varía dependiendo del método que lo retorne.
<code>public bool EstaSellado { get; set; }</code>	<i>Booleano</i> que indicará si el comprobante está sellado.
<code>public string CadenaOriginal { get; set; }</code>	<i>String</i> que alberga la cadena original del comprobante.
<code>public string Sello { get; set; }</code>	Sello del comprobante.
<code>public string NoCert { get; set; }</code>	Número de certificado.

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 <code>XmlElement</code> que contiene Timbre en XML.
<code>public byte[] Cbb { get; set; }</code>	Arreglo de bytes del archivo imagen jpg del código de barras bidimensional.
<code>public TimbreBE Timbre { get; set; }</code>	Datos del timbre fiscal versión 1.1.
<code>public XmlNode NodoTimbre { get; set; }</code>	Nodo del complemento timbre.
<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 globla, 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 globa.

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

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>	String con el detalle del error.

Constructor	Descripción
<code>public List<string> 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 <i>ConsultaRelacionados</i> 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

Constructor	Descripción
<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>	String con el detalle del error.
<code>public string CodEstatus { get; set; }</code>	Código del WS
<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>	String con el detalle del error.
<code>public string CodEstatus { get; set; }</code>	Código de estatus donde, 1000 = Se recibió la respuesta de la petición de forma exitosa
<code>public string CodEstatusDescripcion { get; set; }</code>	Descripción del código de estatus

Constructor	Descripción
<code>public string RfcReceptor { get; set; }</code>	RFC receptor que envia 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<DetalleFolios> Folios { get; set; }</code>	Colección de objetos DetalleFolios

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 específico para la petición Cancelada
<code>public string EstatusUUIDDscripcion { get; }</code>	Descripción del estatus UUID.
<code>public TipoRespuesta Respuesta { get; set; }</code>	Respuesta correspondiente al folio que se envia.

9.6.9.3 Enumeración 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
<code>detallistaAllowanceChargeAllowanceChargeType</code>
<code>detallistaAllowanceChargeMonetaryAmountOrPercentageRateBase</code>
<code>detallistaAllowanceChargeSettlementType</code>
<code>detallistaAllowanceChargeSpecialServicesType</code>
<code>detallistaCurrencyCurrencyFunction</code>
<code>detallistaCurrencyCurrencyISOCODE</code>
<code>detallistaDocumentStatus</code>
<code>detallistaInvoiceCreatorAlternatePartyIdentificationType</code>

Enumeraciones de Detallista

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>	String que contiene el XML del acuse de cancelación.
<code>public DateTime Fecha { get; set; }</code>	Fecha en que se generó el acuse.

Constructor	Descripción
<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 <code>status</code> .
<code>public List<AcuseDetalle> AcuseDetalle { get; set; }</code>	Lista de detalles de acuse (indicando en UUID y estatus) ver clase AcuseDetalle .

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

9.9.4 Clase Cancelacion 2018

9.9.4.1 Constructores

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

9.9.4.2 Métodos

Metodos de Cancelacion2018	Descripción
<pre>public ReturnCancelacion2018 cancelaCfdi(ref string msg, string[] listaUUIIDS, 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 @listaUUIIDS, siguiendo el nuevo esquema de cancelación, es decir, se sigue el nuevo mecanismo implementado para la cancelación.</p> <p>Excepciones: Exception Si no existe el certificado indicado por el path @rutaCert. Si el certificado es inválido. Si hubo un error al crear el PFX.</p> <p>ArgumentException Ninguna.</p> <p>Returns: Objeto de tipo <code>ReturnCancelacion</code> que indica el resultado de la operación cancelar (ver clase <code>ReturnCancelacion</code>).</p>
<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>Returns: 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 ReturnQrAcuse consultaEstatusCfdi(string uuid, string rfcEmisor, string rfcReceptor, decimal total, string selloCfd8, Definiciones.TipoTimbrado tipoServicio);</pre>	<p>Consulta el estado de un CFDI.</p> <p>@selloCfd8 Ultimos 8 caracteres del sello</p> <p>Returns: 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 ReturnPeticonesPendientes getPeticonesCancelacionPendientes(string rfcUsuarioSifei, string password,</pre>	<p>Consulta las peticiones de cancelación pendiente que requieren autorización para ser canceladas por el @rfcReceptor</p>

Metodos de Cancelacion2018	Descripción
<pre>string rfcReceptor, Definiciones.TipoTimbrado pTipoTimbrado); public ReturnAceptacionRechazoResult procesaRespuestaPeticonesCancelacion(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>Returns: Objeto de tipo <code>ReturnPeticonesPendientes</code>,</p> <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>Returns: <code>ReturnAceptacionRechazoResult</code></p>

9.10 Referencia del Namespace SifeiSignature3_3.Complementos.Cce11

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
<code>public ComercioExterior();</code>	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>

Propiedades

```
public bool CertificadoOrigenSpecified { get; set; }
public decimal TotalUSD { get; set; }
public int CertificadoOrigen { get; set; }
public string ClaveDePedimento { get; set; }
public string TipoOperacion { get; set; }
public bool MotivoTrasladoSpecified { get; set; }
public string MotivoTraslado { get; set; }
public string Version { get; set; }
public ComercioExteriorMercancia[] Mercancias { get; set; }
public ComercioExteriorDestinatario[] Destinatario { get; set; }
public ComercioExteriorReceptor Receptor { get; set; }
public ComercioExteriorPropietario[] Propietario { get; set; }
public ComercioExteriorEmisor Emisor { get; set; }
public bool ClaveDePedimentoSpecified { get; set; }
public bool TotalUSDSpecified { get; set; }
```

9.10.3 Clase ComercioExteriorDestinatario

9.10.3.1 Constructores

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

9.10.3.2 Propiedades

Propiedades

```
public ComercioExteriorDestinatarioDomicilio[] Domicilio { get; set; }
public string NumRegIdTrib { get; set; }
public string Nombre { get; set; }
```

9.10.4 Clase ComercioExteriorDestinatarioDomicilio

9.10.4.1 Constructores

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

9.10.4.2 Propiedades

Propiedades

```
public string Calle { get; set; }
public string NumeroExterior { get; set; }
public string NumeroInterior { get; set; }
public string Colonia { get; set; }
public string Localidad { get; set; }
public string Referencia { get; set; }
public string Municipio { get; set; }
public string Estado { get; set; }
public string Pais { get; set; }
public string CodigoPostal { get; set; }
```

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</code> UnidadAduanaSpecified { <code>get</code> ; <code>set</code> ; }
<code>public decimal</code> ValorUnitarioAduana { <code>get</code> ; <code>set</code> ; }
<code>public bool</code> ValorUnitarioAduanaSpecified { <code>get</code> ; <code>set</code> ; }
<code>public decimal</code> ValorDolares { <code>get</code> ; <code>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</code> Marca { <code>get</code> ; <code>set</code> ; }
<code>public string</code> Modelo { <code>get</code> ; <code>set</code> ; }
<code>public string</code> SubModelo { <code>get</code> ; <code>set</code> ; }
<code>public string</code> NumeroSerie { <code>get</code> ; <code>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</code> NumRegIdTrib { <code>get</code> ; <code>set</code> ; }
<code>public string</code> ResidenciaFiscal { <code>get</code> ; <code>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
<code>public ComercioExteriorReceptorDomicilio</code> Domicilio { <code>get</code> ; <code>set</code> ; }
<code>public string</code> NumRegIdTrib { <code>get</code> ; <code>set</code> ; }

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

Clases

- class [Nomina](#)
- class [NominaDeducciones](#)
- class [NominaDeduccionesDeducccion](#)
- class [NominaEmisor](#)
- class [NominaEmisorEntidadSNCF](#)
- class [NominaIncapacidad](#)
- class [NominaOtroPago](#)
- class [NominaOtroPagoCompensacionSaludosAFavor](#)
- 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
NominaReceptorSindicalizado

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>


```
public NominaReceptor Receptor { get; set; }
public NominaEmisor Emisor { get; set; }
public string TipoNomina { get; set; }
public bool TotalOtrosPagosSpecified { get; set; }
```

9.12.4 Clase NominaDeducciones

9.12.4.1 Propiedades

Propiedades

```
public NominaDeduccionesDeduccion[] Deduccion { get; set; }
public decimal TotalOtrasDeducciones { get; set; }
public bool TotalOtrasDeduccionesSpecified { get; set; }
public decimal TotalImpuestosRetenidos { get; set; }
public bool TotalImpuestosRetenidosSpecified { get; set; }
```

9.12.5 Clase NominaDeduccionesDeduccion

9.12.5.1 Propiedades

Propiedades

```
public string TipoDeduccion { get; set; }
public string Clave { get; set; }
public string Concepto { get; set; }
public decimal Importe { get; set; }
```

9.12.6 Clase NominaEmisor

9.12.6.1 Propiedades

Propiedades

```
public string TipoDeduccion { get; set; }
public string Clave { get; set; }
public string Concepto { get; set; }
public decimal Importe { get; set; }
```

9.12.7 Clase NominaEmisorEntidadSNCF

9.12.8 Propiedades

Propiedades

```
public string OrigenRecurso { get; set; }
public decimal MontoRecursoPropio { get; set; }
public bool MontoRecursoPropioSpecified { get; set; }
```

9.12.9 Clase NominaIncapacidad

9.12.9.1 Propiedades

Propiedades

```
public int DiasIncapacidad { get; set; }
public string TipoIncapacidad { get; set; }
public decimal ImporteMonetario { get; set; }
public bool ImporteMonetarioSpecified { get; set; }
```

9.12.10 Clase NominaOtroPago

9.12.10.1 Propiedades

Propiedades
<code>public NominaOtroPagoSubsidioAlEmpleado SubsidioAlEmpleado { 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 NominaOtroPagoSubsidioAlEmpleado

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>

Propiedades

```
public decimal IngresoAcumulable { get; set; }
public decimal IngresoNoAcumulable { get; set; }
```

9.12.15 Clase NominaPercepcionesPercepcion

9.12.15.1 Propiedades

Propiedades

```
public NominaPercepcionesPercepcionAccionesOTitulos AccionesOTitulos { get; set; }
public NominaPercepcionesPercepcionHorasExtra[] HorasExtra { get; set; }
public string TipoPercepcion { get; set; }
public string Clave { get; set; }
public string Concepto { get; set; }
public decimal ImporteGravado { get; set; }
public decimal ImporteExento { get; set; }
```

9.12.16 Clase NominaPercepcionesPercepcionAccionesOTitulos

9.12.16.1 Propiedades

Propiedades

```
public decimal ValorMercado { get; set; }
public decimal PrecioAlOtorgarse { get; set; }
```

9.12.17 Clase NominaPercepcionesPercepcionHorasExtra

9.12.17.1 Propiedades

Propiedades

```
public int Dias { get; set; }
public string TipoHoras { get; set; }
public int HorasExtra { get; set; }
public decimal ImportePagado { get; set; }
```

9.12.18 Clase NominaPercepcionesSeparacionIndemnizacion

9.12.18.1 Propiedades

Propiedades

```
public decimal TotalPagado { get; set; }
public int NumAñosServicio { get; set; }
public decimal UltimoSueloMensOrd { get; set; }
public decimal IngresoAcumulable { get; set; }
public decimal IngresoNoAcumulable { get; set; }
```

9.12.19 Clase NominaReceptor

9.12.19.1 Propiedades

Propiedades

```
public decimal SalarioDiarioIntegrado { get; set; }
public bool SalarioBaseCotAporSpecified { get; set; }
public decimal SalarioBaseCotApor { get; set; }
public string CuentaBancaria { get; set; }
```

Propiedades

```
public bool BancoSpecified { get; set; }
public string Banco { get; set; }
public string PeriodicidadPago { get; set; }
public bool RiesgoPuestoSpecified { get; set; }
public string RiesgoPuesto { get; set; }
public string Puesto { get; set; }
public string Departamento { get; set; }
public bool SalarioDiarioIntegradoSpecified { get; set; }
public string NumEmpleado { get; set; }
public bool TipoJornadaSpecified { get; set; }
public string TipoJornada { get; set; }
public bool SindicalizadoSpecified { get; set; }
public NominaReceptorSindicalizado Sindicalizado { get; set; }
public string TipoContrato { get; set; }
public string Antigüedad { get; set; }
public bool FechaInicioRelLaboralSpecified { get; set; }
public DateTime FechaInicioRelLaboral { get; set; }
public string NumSeguridadSocial { get; set; }
public string Curp { get; set; }
public NominaReceptorSubContratacion[] SubContratacion { get; set; }
public string TipoRegimen { get; set; }
public string ClaveEntFed { get; set; }
```

9.12.20 Clase NominaReceptorSubContratacion

9.12.20.1 Propiedades

Propiedades

```
public string RfcLabora { get; set; }
public decimal PorcentajeTiempo { get; set; }
```

9.13 Referencia del Namespace SifeiSignature3_3.Complementos.Pagos10

Clases

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

9.13.2 Clase Pagos

9.13.2.1 Propiedades

Propiedades

```
public PagosPago[] Pago { get; set; }
public string Version { get; set; }
```



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 .Complementos.cartaporte20

9.14.1 Clases

- `class CartaPorte`
- `class CartaPorteUbicacion`
- `class CartaPorteUbicacionDomicilio`
- `class CartaPorteMercancias`
- `class CartaPorteMercanciasMercancia`
- `class CartaPorteMercanciasMercanciaPedimentos`
- `class CartaPorteMercanciasMercanciaGuiasIdentificacion`
- `class CartaPorteMercanciasMercanciaCantidadTransporta`
- `class CartaPorteMercanciasMercanciaDetalleMercancia`
- `class CartaPorteMercanciasAutotransporte`
- `class CartaPorteMercanciasAutotransporteIdentificacionVehicular`
- `class CartaPorteMercanciasAutotransporteSeguros`
- `class CartaPorteMercanciasAutotransporteRemolque`
- `class CartaPorteMercanciasTransporteMaritimo`
- `class CartaPorteMercanciasTransporteMaritimoContenedor`
- `class CartaPorteMercanciasTransporteAereo`
- `class CartaPorteMercanciasTransporteFerroviario`
- `class CartaPorteMercanciasTransporteFerroviarioDerechosDePaso`
- `class CartaPorteMercanciasTransporteFerroviarioCarro`
- `class CartaPorteMercanciasTransporteFerroviarioCarroContenedor`
- `class CartaPorteTiposFigura`
- `class CartaPorteTiposFiguraPartesTransporte`
- `class CartaPorteTiposFiguraDomicilio`

9.14.2 Clase CartaPorte

9.14.2.1 Propiedades

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

Propiedades
public bool EntradaSalidaMercSpecified { set; get; }
public CartaPorteTiposFigura[] FiguraTransporte { set; get; }
public CartaPorteMercancias Mercancias { set; get; }
public string PaisOrigenDestino { set; get; }
public bool PaisOrigenDestinoSpecified { set; get; }
public decimal TotalDistRec { set; get; }
public bool TotalDistRecSpecified { set; get; }
public string TranspInternac { set; get; }
public CartaPorteUbicacion[] Ubicaciones { set; get; }
public string Version { set; get; }
public string ViaEntradaSalida { set; get; }
public bool ViaEntradaSalidaSpecified { set; get; }

9.14.3 Clase CartaPorteMercancias

9.14.3.1 Propiedades

Propiedades
public CartaPorteMercanciasAutotransporte Autotransporte { set; get; }
public decimal CargoPorTasacion { set; get; }
public bool CargoPorTasacionSpecified { set; get; }
public CartaPorteMercanciasMercancia[] Mercancia { set; get; }
public int NumTotalMercancias { set; get; }
public decimal PesoBrutoTotal { set; get; }
public decimal PesoNetoTotal { set; get; }
public bool PesoNetoTotalSpecified { set; get; }
public CartaPorteMercanciasTransporteAereo TransporteAereo { set; get; }
public CartaPorteMercanciasTransporteFerroviario TransporteFerroviario { set; get; }
public CartaPorteMercanciasTransporteMaritimo TransporteMaritimo { set; get; }
public string UnidadPeso { set; get; }

9.14.4 Clase CartaPorteMercanciasAutotransporte

9.14.4.1 Propiedades

Propiedades
public CartaPorteMercanciasAutotransporteIdentificacionVehicular IdentificacionVehicular { set; get; }
public string NumPermisoSCT { set; get; }
public string PermSCT { set; get; }
public CartaPorteMercanciasAutotransporteRemolque[] Remolques { set; get; }
public CartaPorteMercanciasAutotransporteSeguros Seguros { set; get; }

9.14.5 Clase CartaPorteMercanciasAutotransporteIdentificacionVehicular

9.14.5.1 Propiedades

Propiedades
public int AnioModeloVM { set; get; }
public string ConfigVehicular { set; get; }
public string PlacaVM { set; get; }

9.14.6 Clase CartaPorteMercanciasAutotransporteRemolque

9.14.6.1 Propiedades

Propiedades
public string Placa { set; get; }
public string SubTipoRem { set; get; }

9.14.7 Clase CartaPorteMercanciasAutotransporteSeguros

9.14.7.1 Propiedades

Propiedades
public string AseguraCarga { set; get; }
public string AseguraMedAmbiente { set; get; }
public string AseguraRespCivil { set; get; }
public string PolizaCarga { set; get; }
public string PolizaMedAmbiente { set; get; }
public string PolizaRespCivil { set; get; }
public decimal PrimaSeguro { set; get; }
public bool PrimaSeguroSpecified { set; get; }

9.14.8 Clase CartaPorteMercanciasMercancia

9.14.8.1 Propiedades

Propiedades
public string BienesTransp { set; get; }
public decimal Cantidad { set; get; }
public CartaPorteMercanciasMercanciaCantidadTransporta[] CantidadTransporta { set; get; }
public string ClaveSTCC { set; get; }
public string ClaveUnidad { set; get; }
public string CveMaterialPeligroso { set; get; }
public bool CveMaterialPeligrosoSpecified { set; get; }
public string Descripcion { set; get; }
public string DescripEmbalaje { set; get; }
public CartaPorteMercanciasMercanciaDetalleMercancia DetalleMercancia { set; get; }
public string Dimensiones { set; get; }
public string Embalaje { set; get; }
public bool EmbalajeSpecified { set; get; }
public string FraccionArancelaria { set; get; }
public bool FraccionArancelariaSpecified { set; get; }
public CartaPorteMercanciasMercanciaGuiasIdentificacion[] GuiasIdentificacion { set; get; }
public string MaterialPeligroso { set; get; }
public bool MaterialPeligrosoSpecified { set; get; }
public string Moneda { set; get; }
public bool MonedaSpecified { set; get; }
public CartaPorteMercanciasMercanciaPedimentos[] Pedimentos { set; get; }
public decimal PesoEnKg { set; get; }
public string Unidad { set; get; }
public string UUIDComercioExt { set; get; }
public decimal ValorMercancia { set; get; }
public bool ValorMercanciaSpecified { set; get; }

9.15 Referencia del *Namespace* SifeiSignature3_3.srvSifeiCancelacion2018

Clases

- `class folios`

9.15.2 Enumeraciones

- `Respuesta`

Respuesta
Aceptacion = 0,
Rechazo = 1

9.15.3 Clase folios

9.15.3.1 Propiedades

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

9.16 Códigos de Respuesta de Cancelación

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

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



9.17 Códigos de Error de la Matriz de validaciones para CFDI versión 4.0

Para CFDI 4.0 se están siguiendo las reglas de validación que el SAT especificó en la Matriz de códigos de error para el Comprobante Fiscal Digital por Internet, version 4.0, disponible en

http://omawww.sat.gob.mx/tramitesyservicios/Paginas/anexo_20.htm

9.18 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

9.19 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

9.20 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

9.21 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

HORARIO DE ATENCIÓN

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

Lunes a viernes

De 09:00 a 19:00 hrs.

PÁGINAS OFICIALES DE SIFEI

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

UBICACIÓN DE OFICINA MATRIZ

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

<Fin del Documento>

