

“WsDex 3.0”

Guía de ejemplo de Integración ECODEX



Powered by Deconti Systems



Contenido

WsDex – Webservice Versión 2.0.....	4
Soy nuevo en Web Services.....	4
Firma Digital:	4
Nuevas funciones mediante servicios REST:	5
Generar Token para Alta de Certificado:	5
Recuperar Comprobantes utilizando el Hash:	6
Obtener ultimos Documentos Timbrados:	7
Generar Token de Autenticación:	8
Servicio Seguridad:	9
Token de Servicio:.....	9
Token de transacción de usuario:.....	9
Token de alta de emisores.....	11
Servicio Clientes:.....	12
SolicitudRegistroCliente	13
AsignacionTimbres	14
Servicio Cancelación:	15
SolicitudCancelaMultiple	16
SolicitudCancelaOtros	17
RecuperaAcuseCancelacion.....	19
Servicio Repositorio:.....	21
Servicio Timbrar:.....	26
Seguridad:.....	28
Manejo de excepciones:.....	28
Kit de Pruebas.....	29
Direcciones (URLS) de Webservice de pruebas.....	30
Códigos de Error enviados por SAT	31

Complementos	31
Retenciones:	32
Validación de pruebas	32
LOCAL WSDL y otros Lenguajes	32
Herramientas WebService	32
Tabla de control de cambios	33

WsDex – WebService Versión 2.0

Desde su salida al mercado, ECODEX se ha centrado en la integración de empresas con un esquema de administración de facturación, que tiene la ventaja de ser modificado para ser conectado con nuestro servicio web (**WebService**) y obtener los sellos y timbres de comprobantes digitales (Comprobantes Fiscales Digitales – *CFDI's*) siendo el servicio prestado por ECODEX que está autorizado como **Proveedor Autorizado de Certificación** ante el **SAT**.

Soy nuevo en WebService

El modelo del cfdi ha representado un reto para la industria de desarrollo de software mexicana, pero ha sido una gran oportunidad de crecimiento. Los desarrolladores se han empapado de nuevo conocimiento y el entendimiento de web service les ha permitido ampliar su uso más allá del cfdi. La mayoría de los lenguajes más utilizados tienen bastante soporte para su manejo. En el caso de este kit de integración, se muestran ejemplos con algunos de los lenguajes más populares en nuestro país (C#, vb.net, java y php).

Para el caso de sistemas de los que no se tenga código fuente o cuya implementación de web service sea muy costosa, algunos de nuestros integradores que han desarrollado soluciones para estas necesidades.

Firma Digital:

Nosotros proveemos el servicio de sellado y timbrado de [comprobantes fiscales](#), lo que implica que sellamos el Certificado de Sello digital (CSD) utilizando el sello del cliente para después Timbrarlo con el Certificado de sello digital (CSD) que el SAT ha provisto, certificando todos los comprobantes fiscales que son enviados a nuestro servicio web, después de la validación técnica y legal correspondiente.

El Sello Digital es una cadena de texto con una serie de caracteres. Esta Cadena se forma como resultado de encriptar la información de la Cadena Original del Comprobante, lo que hace que el comprobante sea infalsificable debido a que cualquier cambio en los datos, genera un sello diferente.

Nuevas funciones mediante Servicios REST:

<https://pruebasapi.ecodex.com.mx>

1. Generar Token para Alta de Certificado.

/api/certificados/clave

Método que permite generar una clave (UUID) para carga de certificado del emisor.

Método: GET

Url: <https://pruebasapi.ecodex.com.mx/api/certificados/clave>

Headers:

Authorization: {token_type} {access_token}

X-Auth-Token: SHA1(ID Integrador | {service_token})

GET /api/certificados/clave		
Description		
Obtener una nueva clave para carga de certificado		
Responses		
Code	Description	Schema
200	Clave de certificado y rfc al cual pertenece	<pre>ClaveCertificadoResponse { clave: string Clave de carga de certificado generada rfc: string Rfc del emisor a quién pertenece la clave }</pre>
default	Este error se lanza cuando ocurre algún error	<pre>ErrorModel { error: string Tipo de error presentado durante la solicitud Enum: Array [3] 0: "FallaAplicacion" 1: "FallaValidacion" 2: "FallaSesion" error_description: string Descripción del error presentado error_code: integer Código correspondiente al error error_suggestion: string Posible solución para evitar el error }</pre>

2. Recuperar comprobantes utilizando el Hash.

/api/documentos/{hash}

Método que permite recuperar un documento utilizando como identificador el hash de la cadena original del mismo.

Método: GET

Url: <https://pruebasapi.ecodex.com.mx/api/documentos/{hash}>

Headers:

Authorization: {token_type} {access_token}

X-Auth-Token: SHA1(ID Integrador | {service_token})

GET /api/documentos/{hash}

Description

Obtener Xml del **Documento** solicitado.

Parameters

Name	Located in	Description	Required	Schema
hash	path	Hash de la cadena original del documento	Yes	≙ string

Responses

Code	Description	Schema
200	Xml del comprobante	≙ file (string)
default	Este error se lanza cuando ocurre algún error	≙ <pre>▼ ErrorModel { error: ▼ string Tipo de error presentado durante La solicitud Enum: ▼ Array [3] 0: "FallaAplicacion" 1: "FallaValidacion" 2: "FallaSesion" error_description: ▼ string Descripción del error presentado error_code: ▼ integer Código correspondiente al error error_suggestion: ▼ string Posible solución para evitar el error }</pre>

3. Obtener últimos documentos timbrados.

/api/documentos

Método que permite obtener los últimos 100 documentos timbrados por un emisor.

Método: GET

Url: <https://pruebasapi.ecodex.com.mx/api/documentos>

Headers:

Authorization: {token_type} {access_token}

X-Auth-Token: SHA1(ID Integrador | {service_token})

GET /api/documentos

Description

Obtener lista últimos **Documentos** emitidos.

Parameters

Name	Located in	Description	Required	Schema
limit	query	tamaño límite de la lista	No	≥ integer

Responses

Code	Description	Schema
200	Lista de documentos	<pre> { "documents": [{ "uuid": "string", "hash": "string", "tipoDocumento": "string", "fechaTimbrado": "string (date-time)" }] } </pre> <p><i>Fólio Fiscal del comprobante</i> <i>Hash de la cadena original del comprobante</i> <i>Tipo de comprobante</i> <i>Enum:</i> <i>Array [2]</i> <i>0: "Comprobante"</i> <i>1: "Retenciones"</i> <i>Fecha de timbrado del comprobante</i></p>
default	Este error se lanza cuando ocurre algún error	<pre> { "error": "string", "error_description": "string", "error_code": "integer", "error_suggestion": "string" } </pre> <p><i>Tipo de error presentado durante la solicitud</i> <i>Enum:</i> <i>Array [3]</i> <i>0: "FallaAplicacion"</i> <i>1: "FallaValidacion"</i> <i>2: "FallaSesion"</i> <i>Descripción del error presentado</i> <i>Código correspondiente al error</i> <i>Posible solución para evitar el error</i></p>

4. Generar Token de Autenticacion.

/token?version=2

Este método genera un token de autenticación del servicio válido durante 10 minutos.

Método: POST

Url: <http://pruebasapi.ecodex.com.mx/token?version=2>

Parámetros: rfc={rfc emisor}&grant_type=authorization_token

Content-Type: application/x-www-form-urlencoded

POST /token

Description

Obtener el token de autenticación del servicio

Parameters

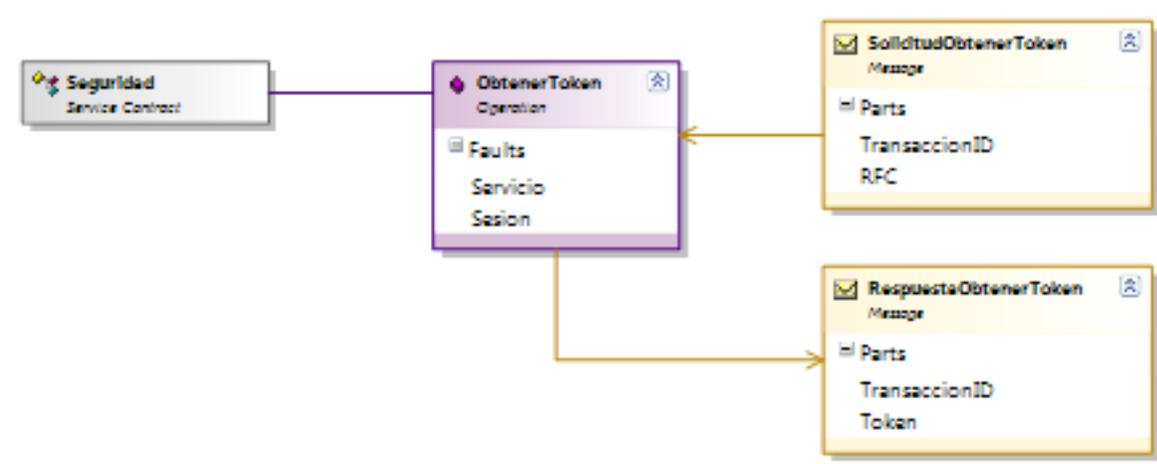
Name	Located in	Description	Required	Schema
version	query	Versión del token a solicitar, se espera el valor 2	Yes	integer
accessTokenRequest	body		Yes	<pre>▼ AccessTokenRequest { grant_type: ▶ string * rfc: ▶ string * }</pre>

Responses

Code	Description	Schema
200	Obtención de token satisfactoria	<pre>▼ AccessTokenResponse { access_token: ▼ string token_type: ▼ string expires_in: ▼ integer service_datetime: ▼ string (date-time) service_token: ▼ string }</pre>
default	Error en generación de token	<pre>▼ ErrorModel { error: ▼ string error_description: ▼ string error_code: ▼ integer error_suggestion: ▼ string }</pre>

Servicio Seguridad:

Este método brinda una forma confiable de controlar el acceso y duración de la sesión, por medio del método ObtenerToken. Esta llamada valida la existencia del emisor en nuestro servicio y regresa un Token de Servicio.



Token de Servicio:

Antes de comenzar el proceso de sellado y timbrado, la aplicación que se conecta con ECODEX debe hacer una petición a nuestro servicio web para solicitar un "token" (clave temporal) y pueda ser utilizado durante una sesión por un período de 10 minutos.

Token de transacción de usuario:

El token de transacción de usuario, es distinto al de Token de Servicio, es un función hash **SHA1** que se calcula utilizando el servicio de token obtenidos de los servicios de seguridad y el **ID integrador** (que es secreta y nunca debe viajar a través de la red), y sirve para comunicarse en las demás solicitudes.

El token es una función hash SHA1 de una cadena con el identificador del integrador, además del signo 'tubería' o pipe, y el Token de Servicio vigente.

Formula:

SHA1(ID Integrador | Token de Servicio)

Ejemplo:

**SHA1 (123 | bba1a8454180f89603def4757f7a4bc03d67397b) =
ab5935f20420a75fa03d981663ee2d94f1663211**

Este servicio incluye un Kit de pruebas de llamadas, así como una consulta del estatus de nuestras aplicaciones, permitiendo conocer la disponibilidad de la aplicación de servicio.

Este kit viene en distintos lenguajes de programación, por ejemplo para el kit de pruebas con **C#** en la **class Utilerias** se puede cambiar los datos de **RFC** y el **UUID** de Integrador, para fines de pruebas.

```
class Utilerias
{
    /// <summary>
    /// This id can come from a database or be hard coded
    /// by each service integrator or service reseller.
    /// </summary>
    public String INTEGRATOR_ID = "2b3a8764-d586-4543-9b7e-82834443f219";
    public String RFC = "SUL010720JN8";
    private IInterface _view;
    public Utilerias()

```

Así como un ejemplo de cómo se podría hacer la llamada a la solicitud de token de seguridad:

```
public string CreateToken(string RFC, Int64 transactionID, ref string token)
{
    string result = string.Empty;
    try
    {
        #region Build Token
        using (var securitySrv = new SrvSecurity.SeguridadClient())
        {
            var serviceToken = securitySrv.ObtenerToken(RFC, ref transactionID);
            var toHash = String.Format("{0}|{1}", INTEGRATOR_ID, serviceToken);
            token = Security.Hash(toHash);
        }
    }
    #endregion
}
```

Token de alta de emisores

El token de alta de emisores es un nuevo token diseñado específicamente para este método. Es un función hash **SHA1** que se calcula utilizando el servicio de token obtenidos de los servicios de seguridad, el **ID integrador** y el **ID de Alta de emisores**.

Al igual que token de transacción, también es una función hash SHA1 de una cadena con el identificador del integrador, además del signo 'tubería' o pipe, el Token de Servicio vigente y el ID de alta de emisores.

Formula:

SHA1(ID Integrador | ID alta de emisores | token de servicio)

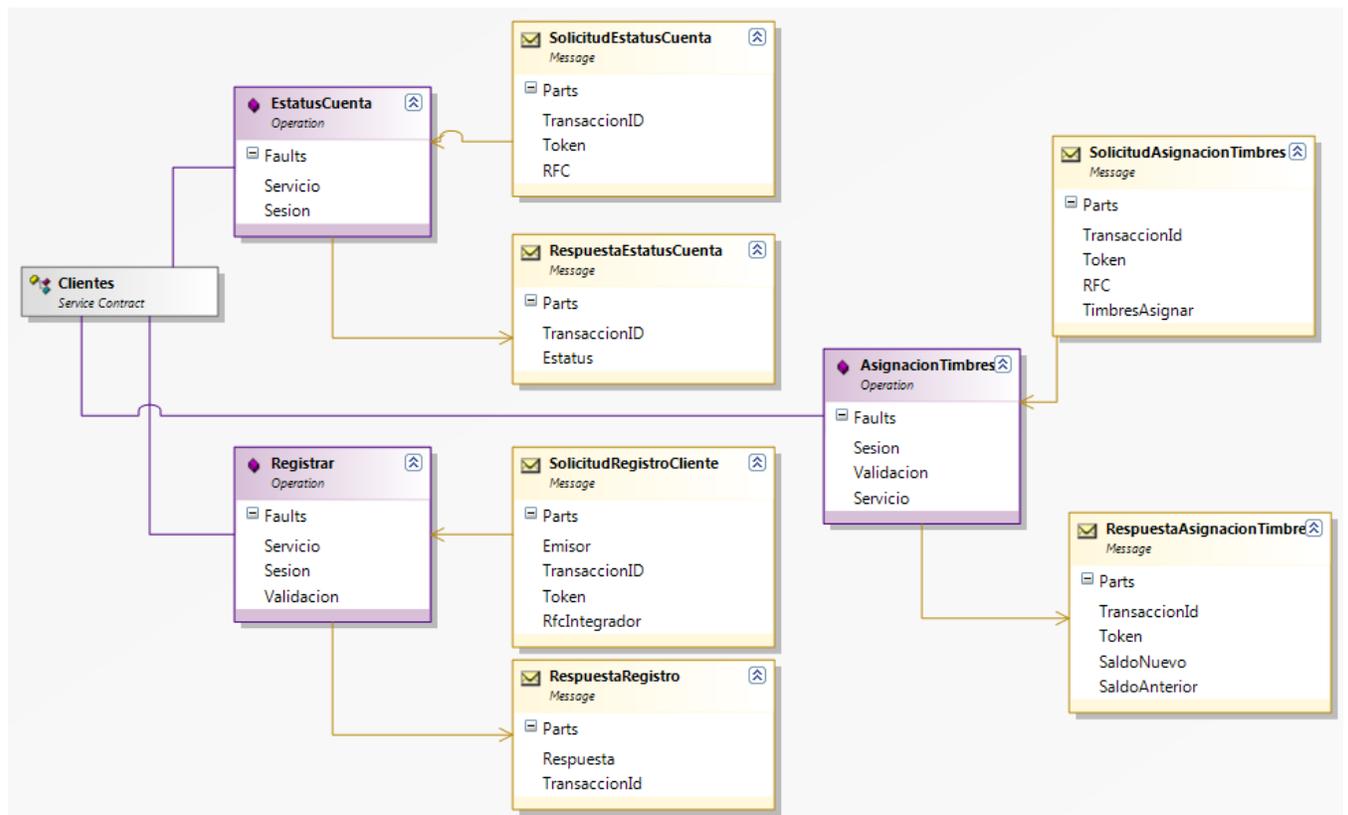
Ejemplo:

**SHA1 (123 | 123312 | bba1a8454180f89603def4757f7a4bc03d67397b) =
ab5935f20420a75fa03d981663ee2d94f1663211**

Servicio Clientes:

Este servicio tiene los métodos:

5. **EstatusCuenta:** Este método sirve para verificar el estatus de la cuenta de un cliente, regresando los datos de la cuenta, así como los timbres asignados y vigentes.
6. **Registrar:** Este método sirve para dar de alta a nuevos Emisores, regresando el RFC del Emisor que se dio de alta, así como su estatus y clave para subir los certificados del mismo.
7. **AsignacionTimbres:** Este método sirve para asignar timbres a un cliente, regresando el saldo nuevo y saldo anterior del cliente a que se le asignaron los timbres. Para desasignar timbres, basta con poner la cantidad en negativo y se hará el descuento en el saldo del emisor.



SolicitudRegistroCliente

Solicitud

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:cfdi="http://Ecodex.WS.Model/2011/CFDI">
  <soapenv:Header/>
  <soapenv:Body>
    <cfdi:SolicitudRegistroCliente>
      <cfdi:Emisor>
        <cfdi:RFC>HEAD010101HJ1</cfdi:RFC>
        <cfdi:RazonSocial>Razon de cliente prueba</cfdi:RazonSocial>
        <cfdi:CorreoElectronico>cliente@ecodex.com.mx</cfdi:CorreoElectronico>
      </cfdi:Emisor>
      <cfdi:RfcIntegrador>BBB010101001</cfdi:RfcIntegrador>
      <cfdi:Token>sdasdasdasdasdaswqeqwh</cfdi:Token>
      <cfdi:TransaccionID>1234</cfdi:TransaccionID>
    </cfdi:SolicitudRegistroCliente>
  </soapenv:Body>
</soapenv:Envelope>
```

RFC: El RFC del Emisor que será dado de Alta.

RazonSocial: Nombre o Razón social del emisor que será dado de Alta.

CorreoElectronico: Es el correo electrónico del emisor que será dado de Alta.

RfcIntegrador: Es el RFC del Integrador que está dando de Alta al emisor.

Token: Es el [token](#) de alta de emisores.

TransaccionID: Es el Identificador de la transacción.

Respuesta

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <RespuestaRegistro xmlns="http://Ecodex.WS.Model/2011/CFDI">
      <Respuesta xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <RFC>HEAD010101HJ1</RFC>
        <Estatus>Activo</Estatus>
        <ClaveCertificado>af6c9bc6-1d23-44fe-a2e4-39d2711c3947</ClaveCertificado>
      </Respuesta>
      <TransaccionId>1234</TransaccionId>
    </RespuestaRegistro>
  </s:Body>
</s:Envelope>
```

RFC: El RFC del Emisor que será dado de Alta.

Estatus: Nombre o Razón social del emisor que será dado de Alta.

ClaveCertificado: Es la clave de certificado para la generación de la URL para subir el certificado del emisor.

TransaccionID: Es el mismo Identificador de la transacción original.

AsignacionTimbres

Solicitud

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:cfdi="http://Ecodex.WS.Model/2011/CFDI">
  <soapenv:Header/>
  <soapenv:Body>
    <cfdi:SolicitudAsignacionTimbres>
      <cfdi:RFC>AAA010101AAA</cfdi:RFC>
      <cfdi:TimbresAsignar>150</cfdi:TimbresAsignar>
      <cfdi:Token>79249aefac5fbb73981412e8fa41a50cd5c75b75</cfdi:Token>
      <cfdi:TransaccionId>1232</cfdi:TransaccionId>
    </cfdi:SolicitudAsignacionTimbres>
  </soapenv:Body>
</soapenv:Envelope>
```

RFC: El RFC del Emisor al que se le harán la asignación / desasignación.

TimbresAsignar: Valor entero con el número de timbres a asignar. Para desasignar el valor debe ser negativo y se hará la transacción si el usuario cuenta con suficientes folios.

Token: Es el token de transacción del usuario.

TransaccionID: Es el Identificador de la transacción.

Respuesta

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <RespuestaAsignacionTimbres xmlns="http://Ecodex.WS.Model/2011/CFDI">
      <SaldoAnterior>51127</SaldoAnterior>
      <SaldoNuevo>51277</SaldoNuevo>
      <cfdi:Token>79249aefac5fbb73981412e8fa41a50cd5c75b75</cfdi:Token>
      <TransaccionId>1232</TransaccionId>
    </RespuestaAsignacionTimbres>
  </s:Body>
</s:Envelope>
```

SaldoAnterior: Saldo del emisor antes de la asignación.

SaldoNuevo: Saldo del emisor después de la asignación.

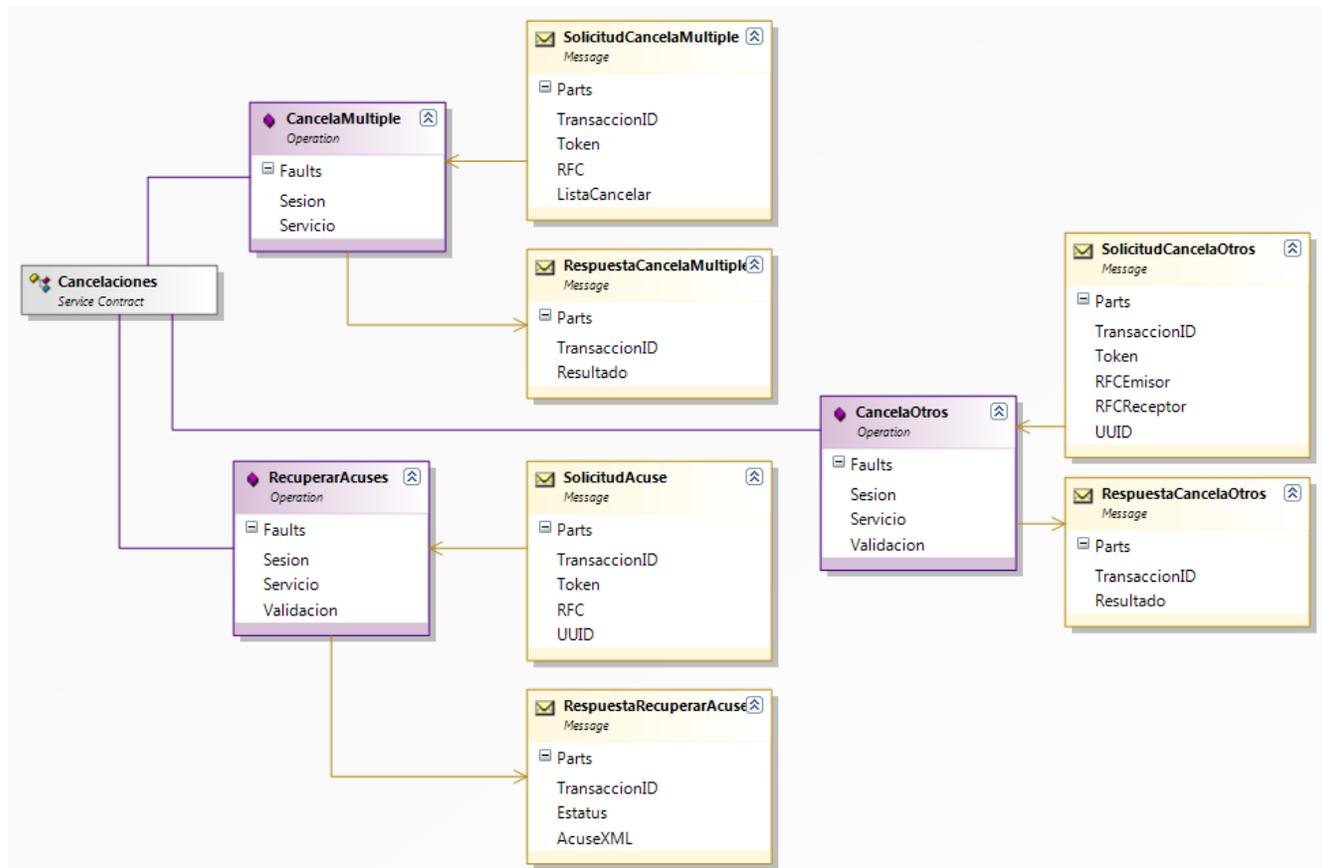
Token: Es el token de transacción del usuario.

TransaccionID: Es el Identificador de la transacción

Servicio Cancelación:

Este servicio cuenta con 3 métodos:

8. **CancelaMúltiples:** Método para cancelar uno o más comprobantes en una sola llamada. Sirve tanto para servicios de sella timbra como de solo timbrado. Gradualmente irá sustituyendo al resto de los métodos de cancelación.
9. **CancelaOtros:** Método para cancelar comprobantes de otros PAC's. No realiza verificación de la existencia de los comprobantes del emisor, así que es responsabilidad del emisor que existan.
10. **RecuperarAcuses:** Obtiene los acuses de cancelación del SAT de todos los métodos de cancelación existentes.



SolicitudCancelaMultiple

Solicitud:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:cfdi="http://Ecodex.WS.Model/2011/CFDI">
  <soapenv:Header/>
  <soapenv:Body>
    <cfdi:SolicitudCancelaMultiple>
      <cfdi:ListaCancelar>
        <cfdi:guid>F13CD4A0-CE57-4313-96E1-B616859D2C2E</cfdi:guid>
        <cfdi:guid>6E456701-8964-450F-AC09-6052E39618B7</cfdi:guid>
        <cfdi:guid>34A282F4-D631-4979-B754-BF551216001F</cfdi:guid>
      </cfdi:ListaCancelar>
      <cfdi:RFC>AAA010101AAA</cfdi:RFC>
      <cfdi:Token>7581eab9752726c6d8b2b8ef49d3a8e7dd6beb20</cfdi:Token>
      <cfdi:TransaccionID>1234</cfdi:TransaccionID>
    </cfdi:SolicitudCancelaMultiple>
  </soapenv:Body>
</soapenv:Envelope>
```

ListaCancelar: Permite agregar la lista de los UUID de los comprobantes que se manden a cancelar.

Guid: Identificador asignado por nuestro servicio durante el proceso de Sellado y Timbrado.

RFC: El RFC del Contribuyente.

Token: Es el token de transacción del usuario.

TransaccionID: Es el Identificador de la transacción.

Respuesta:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <RespuestaCancelaMultiple xmlns="http://Ecodex.WS.Model/2011/CFDI">
      <Resultado xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <ResultadoCancelacion>
          <UUID>f13cd4a0-ce57-4313-96e1-b616859d2c2e</UUID>
          <Estatus>Cancelado</Estatus>
        </ResultadoCancelacion>
        <ResultadoCancelacion>
          <UUID>6e456701-8964-450f-ac09-6052e39618b7</UUID>
          <Estatus>Cancelado</Estatus>
        </ResultadoCancelacion>
        <ResultadoCancelacion>
          <UUID>34a282f4-d631-4979-b754-bf551216001f</UUID>
          <Estatus>Cancelado</Estatus>
        </ResultadoCancelacion>
      </Resultado>
      <TransaccionID>1234</TransaccionID>
    </RespuestaCancelaMultiple>
  </s:Body>
</s:Envelope>
```

- **UUID:** Identificador asignado por nuestro servicio durante el proceso de Sellado y Timbrado.
- **Estatus:** Estado de la factura si realizo con éxito la cancelación tendrá en Estatus Cancelado.
- **TransaccionID:** El mismo identificador de transacción de la solicitud.

SolicitudCancelaOtros

Solicitud:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:cfdi="http://Ecodex.WS.Model/2011/CFDI">
  <soapenv:Header/>
  <soapenv:Body>
    <cfdi:SolicitudCancelaOtros>
      <cfdi:RFCEmisor>AAA010101AAA</cfdi:RFCEmisor>
      <cfdi:RFCReceptor>CROW100623S70</cfdi:RFCReceptor>
      <cfdi:Token>6f2ead943965904a5542d60cad023b6cf04d70d6</cfdi:Token>
      <cfdi:TransaccionID>1234</cfdi:TransaccionID>
      <cfdi:UUID>FC782F05-6005-447D-89D6-D0DD0E49AAEA</cfdi:UUID>
    </cfdi:SolicitudCancelaOtros>
  </soapenv:Body>
</soapenv:Envelope>
```

RFCEmisor: El RFC del Contribuyente.

RFCReceptor: El RFC del Contribuyente a quien se emitió el comprobante.

Token: Es el token de transacción del usuario.

TransaccionID: Es el Identificador de la transacción.

UUID: Identificador asignado por el servicio durante el proceso de Sellado y Timbrado.

Respuesta:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <RespuestaCancelaOtros xmlns="http://Ecodex.WS.Model/2011/CFDI">
      <Resultado xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <ResultadoCancelacion>
          <UUID>fc782f05-6005-447d-89d6-d0dd0e49aaaa</UUID>
          <Estatus>Cancelada</Estatus>
        </ResultadoCancelacion>
      </Resultado>
      <TransaccionID>1234</TransaccionID>
    </RespuestaCancelaOtros>
  </s:Body>
</s:Envelope>
```

Resultado:

- **UUID:** Identificador asignado por nuestro servicio durante el proceso de Sellado y Timbrado.
- **Estatus:** Estado de la factura si realizo con éxito la cancelación tendrá en Estatus Cancelado.
- **TransaccionID:** El mismo identificador de transacción de la solicitud.

RecuperaAcuseCancelacion

Solicitud:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope,
xmlns:cfdi="http://Ecodex.WS.Model/2011/CFDI">
  <soapenv:Header/>
  <soapenv:Body>
    <cfdi:SolicitudAcuse>
      <cfdi:RFC>AAA010101AAA</cfdi:RFC>
      <cfdi:Token>e2053ac187cc7f7c1dba94979612d18623208935</cfdi:Token>
      <cfdi:TransaccionID>1123</cfdi:TransaccionID>
      <cfdi:UUID>2C12DB6D-E392-4870-BD53-724DF9224BF4</cfdi:UUID>
    </cfdi:SolicitudAcuse>
  </soapenv:Body>
</soapenv:Envelope>
```

SolicitudAcuse

- **RFC:** El RFC del Contribuyente.
- **Token:** Es el token de transacción del usuario.
- **TransaccionID:** Es el Identificador de la transacción.
- **UUID:** Identificador del comprobante que se desea obtener el acuse.

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <RespuestaRecuperarAcuse xmlns="http://Ecodex.WS.Model/2011/CFDI">
      <AcuseXML><![CDATA[<?xml version="1.0"?>
<Acuse xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
RfcEmisor="AAA010101AAA" Fecha="2013-08-20T20:09:25.6851404">
  <Folios xmlns="http://cancelacfd.sat.gob.mx">
    <UUID>2C12DB6D-E392-4870-BD53-724DF9224BF4</UUID>
    <EstatusUUID>201</EstatusUUID>
  </Folios>
  <Signature .....
  .....</Signature>
</Acuse>]]></AcuseXML>
      <Estatus>Cancelada</Estatus>
      <TransaccionID>1123</TransaccionID>
    </RespuestaRecuperarAcuse>
  </s:Body>
</s:Envelope>
```

RespuestaAcuse

- **AcuseXML:** Acuse de cancelación del sat.
- **Estatus: Estatus del comprobante**
- **TransaccionID:** Es el Identificador de la transacción.

NOTA: En el servicio de **RecuperaAcuseCancelacion** se podrán obtener acuses en los casos en que los Estatus ante el SAT sean 201 ó 202. Si el tiempo transcurrido entre la emisión y la cancelación es mayor a 3 horas, el acuse podrá obtenerse de manera casi inmediata. Si el tiempo transcurrido entre la emisión y la cancelación es menor a dos horas, el acuse tarda en promedio un poco más de 2 horas en reflejarse en nuestros repositorios. Si el acuse es de un comprobante timbrado con otro pac, el tiempo puede incluso variar más. Favor de tomar esto en cuenta para programar la recuperación de los acuses después de un lapso razonable después de la emisión para evitar peticiones innecesarias y darles mejor servicio.

Importante: acerca del Hash del sello.

El sello del documento se genera a partir de los elementos definidos por el **xslt** dado por el SAT para la generación del sello Digital para el **Comprobante Fiscal Digital por internet (cfdi)**. Este sello es el resultado de un hash del tipo **Sha1** sobre los elementos de la cadena original de acuerdo al **anexo 20**.

Este sello sirve como identificador único del documento, sin embargo, en algunos casos puede llegar a duplicarse si todos los elementos de un documento son iguales (*emisor, receptor, conceptos, importe, total, complementos, hora de sellado, etc*). Cuando esto suceda el web Service enviara una excepción del tipo **"Duplicidad de Hash"** en el segundo documento.

En el caso del servicio de **Solo Timbrado** será necesario volver a sellar el documento con una hora diferente al menos para que pueda ser timbrado correctamente. Es importante tomar en cuenta que para la generación de este Hash **NO** se toma en cuenta el **folio** y la **serie**.

Servicio Repositorio:

Este servicio por compatibilidad permanece sin cambios, pero en caso de cambiar del servicio de sella timbra a solo timbrado, se recomienda utilizar las referencias a los métodos del servicio de timbrado:

11. **CancelaComprobante:** Permite la solicitud de la cancelación de un Comprobante fiscal. Este método será sustituido por el método de cancelación masiva.
12. **EstatusComprobante:** Permite saber el estado en que se encuentra un comprobante (Generada o Cancelada)
13. **ObtenerComprobante:** Permite obtener la cadena comprobante XML del comprobante fiscal, de 2 maneras, ya sea por medio del UUID o por el ID de transacción con que fue creada.
14. **ObtenerQR:** Permite obtener la cadena para la generación del QR de un comprobante fiscal.

Para buscar por medio del parámetro de **TransaccionOriginal** siendo esté el **Id de la transacción** con que se envió la solicitud de **SolicitudSellaTimbra**, el valor del atributo **UUID** deberá ser *null*

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:cfdi="http://Ecodex.WS.Model/2011/CFDI">
  <soapenv:Header/>
  <soapenv:Body>
    <cfdi:SolicitudEstatusComprobante>
      <cfdi:RFC>AAA010101AAA</cfdi:RFC>
      <cfdi:Token>cddb6c6454854c6af9cabe8dffeb7bd8d83d2d5d</cfdi:Token>
      <cfdi:TransaccionID>90</cfdi:TransaccionID>
      <cfdi:TransaccionOriginal>100</cfdi:TransaccionOriginal>
      <cfdi:UUID></cfdi:UUID>
    </cfdi:SolicitudEstatusComprobante>
  </soapenv:Body>
</soapenv:Envelope>
```

RFC: El RFC del Contribuyente.

Token: Es el token de transacción del usuario.

TransaccionID: Es el Identificador de la transacción.

TransaccionOriginal: Es el identificador de la transacción con que se envió la solicitud de Sellar y Timbrar (SolicitudSellaTimbraXML), esta opción puede ser usado 3 días después de que el comprobante fue sellado y timbrado.

UUID (Universally unique identifier): Identificador asignado por nuestro servicio durante el proceso de Sellado y Timbrado.

Respuesta:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <RespuestaEstatusComprobante xmlns="http://Ecodex.WS.Model/2011/CFDI">
      <Estatus xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <UUID>e6ed9ce8-4b3c-4458-a950-f5de9f3b4ef4</UUID>
        <Codigo>600</Codigo>
        <Descripcion>Generada</Descripcion>
        <UltimaActualizacion>0001-01-01T00:00:00</UltimaActualizacion>
      </Estatus>
      <TransaccionID>0</TransaccionID>
    </RespuestaEstatusComprobante>
  </s:Body>
</s:Envelope>
```

UUID: Identificador asignado por el servicio del SAT durante el proceso de Sellado y Timbrado.

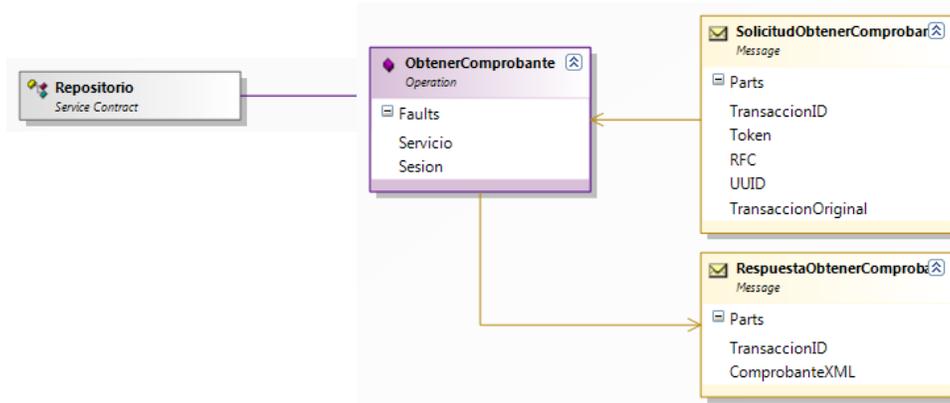
Codigo: Es el código del estado actual en que se encuentra el comprobante fiscal.

Descripcion: Es una breve descripción del estado actual.

UltimaActualizacion: La fecha de la última modificación realizada al comprobante.

TransaccionID: El mismo identificador de transacción de la solicitud.

15. Solicitud Obtener Comprobante



Solicitud:

- Para enviar la solicitud solo con el **UUID**, el valor de la **TransaccionOriginal** deberá ser 0.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cfdi="http://Ecodex.WS.Model/2011/CFDI">
  <soapenv:Header/>
  <soapenv:Body>
    <cfdi:SolicitudObtenerComprobante>
      <cfdi:RFC>AAA010101AAA</cfdi:RFC>
      <cfdi:Token>7bb0c09a107500fa1be86be4038d63afa008ed15</cfdi:Token>
      <cfdi:TransaccionID>117</cfdi:TransaccionID>
      <cfdi:TransaccionOriginal>0</cfdi:TransaccionOriginal>
      <cfdi:UUID>70DF00D9-85F3-4461-A522-BD96CBA1AA09</cfdi:UUID>
    </cfdi:SolicitudObtenerComprobante>
  </soapenv:Body>
</soapenv:Envelope>
  
```

- Para buscar por medio del parámetro de **TransaccionOriginal** siendo éste el **Id de la transacción** con que se envió la solicitud de **SolicitudSellaTimbra**, el valor del atributo UUID deberá ser *null*.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cfdi="http://Ecodex.WS.Model/2011/CFDI">
  <soapenv:Header/>
  <soapenv:Body>
    <cfdi:SolicitudObtenerComprobante>
      <cfdi:RFC>AAA010101AAA</cfdi:RFC>
      <cfdi:Token>7bb0c09a107500fa1be86be4038d63afa008ed15</cfdi:Token>
      <cfdi:TransaccionID>117</cfdi:TransaccionID>
      <cfdi:TransaccionOriginal>110</cfdi:TransaccionOriginal>
      <cfdi:UUID>70DF00D9-85F3-4461-A522-BD96CBA1AA09</cfdi:UUID>
    </cfdi:SolicitudObtenerComprobante>
  </soapenv:Body>
</soapenv:Envelope>
  
```

RFC: El RFC del Contribuyente.

Token: Es el token de transacción del usuario.

TransaccionID: Es el Identificador de la transacción.

TransaccionOriginal: Es el identificador de la transacción con que se envió la solicitud de Sellar y Timbrar (**SolicitudSellaTimbraXML**), esta opción puede ser usado 3 días después de que el comprobante fue sellado y timbrado.

UUID (*Universally unique identifier*): Identificador asignado por nuestro servicio durante el proceso de Sellado y Timbrado.

Respuesta:

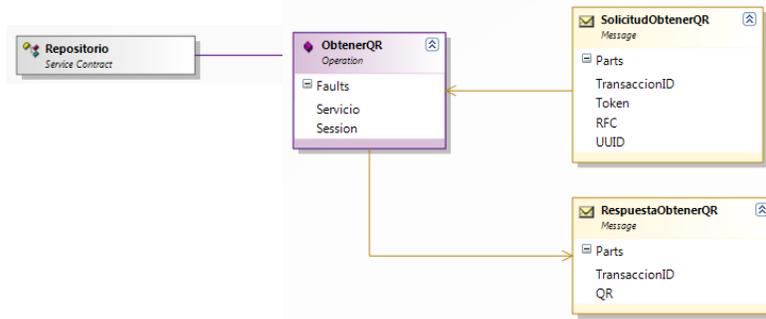
La respuesta a cualquiera de las 2 formas en que se envió (UUID o Transacción Original) será la cadena XML del comprobante encontrado.

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <RespuestaObtenerComprobante xmlns="http://Ecodex.WS.Model/2011/CFDI">
      <ComprobanteXML xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <DatosXML><![CDATA[<cfdi:Comprobante> </cfdi:Comprobante]]></DatosXML>
      </ComprobanteXML>
      <TransaccionID>117</TransaccionID>
    </RespuestaObtenerComprobante>
  </s:Body>
</s:Envelope>
```

CFDi: Es el comprobante en formato de cadena XML, procesado con el Sello y Timbre digital. Al ser enviado con **CDATA** (Character Data), se recibe en el mismo formato, "**<![CDATA [**" y termina con **"]>**".

TransaccionID: El mismo identificador de la transacción.

16. SolicitudObtenerQR (QuickResponse code)



Solicitud:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:cfdi="http://Ecodex.WS.Model/2011/CFDI">
  <soapenv:Header/>
  <soapenv:Body>
    <cfdi:SolicitudObtenerQR>
      <cfdi:RFC>AAA010101AAA</cfdi:RFC>
      <cfdi:Token>ccb9c9078840d58a1e9fed5449326bab2c913cb3</cfdi:Token>
      <cfdi:TransaccionID>0</cfdi:TransaccionID>
      <cfdi:UUID>70DF00D9-85F3-4461-A522-BD96CBA1AA09</cfdi:UUID>
    </cfdi:SolicitudObtenerQR>
  </soapenv:Body>
</soapenv:Envelope>
```

RFC: El RFC del Contribuyente.

Token: Es el token de transacción del usuario.

TransaccionID: Es el Identificador de la transacción.

UUID (Universally unique identifier): Identificador asignado por el servicio del SAT durante el proceso de Sellado y Timbrado.

Respuesta:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <RespuestaObtenerQR xmlns="http://Ecodex.WS.Model/2011/CFDI">
      <QR xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <Imagen>Qk3+AwAAAAAAD4AAAAoAAAAUAAAAFAAAAAABAAEAAAAAAAAAAADEdGAAx4AAAIAAAACAAAAAAAAA...
        |. ///////////////AAD/////////////////AAD/////////////////AAA=</Imagen>
      </QR>
      <TransaccionID>0</TransaccionID>
    </RespuestaObtenerQR>
  </s:Body>
</s:Envelope>
```

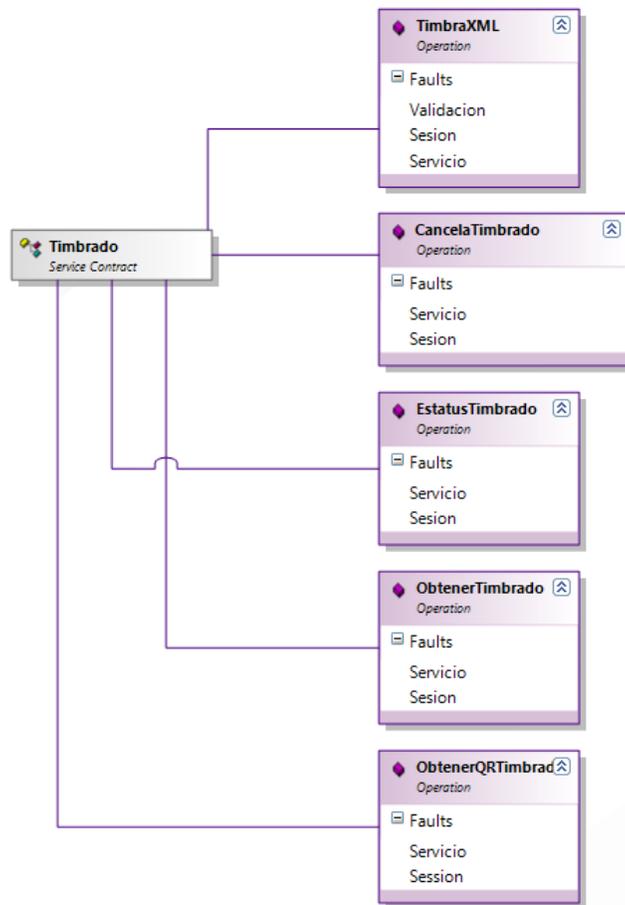
Imagen: Este es el arreglo de bits de la imagen generada para el QR.

TransaccionID: El mismo identificador de transacción de la solicitud.

Servicio Timbrar:

En caso de no requerir el servicio de Sellado se ofrece la opción de **Solo Timbrar**, el cual cuenta con sus propias solicitudes para realizar el **timbrado**, y para el manejo de los comprobantes que solo fueron timbrados por nosotros:

- **TimbraXML:** Permite realizar el timbrado a un CFD con un sello válido, siempre y cuando no haya sido sellado en las últimas 72 horas.
- **CancelaTimbrado:** Permite solicitar la cancelación de un timbre a un CFDi que haya sido timbrado por nuestro servicio.
- **EstatusTimbrado:** Permite consultar el estado actual del timbre digital (Generado y Cancelado)
- **ObtenerTimbrado:** Permite recuperar la cadena XML del nodo del timbre digital de un comprobante por medio de su UUID o número de transacción.
- **ObtenerQRTimbrado:** Permite obtener la cadena del QR generado para el timbrado de un comprobante.



➤ *SolicitudTimbraXML*

Solicitud:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:cfdi="http://Ecodex.WS.Model/2011/CFDI">
  <soapenv:Header/>
  <soapenv:Body>
    <cfdi:SolicitudTimbraXML>
      <cfdi:ComprobanteXML>
        <cfdi:DatosXML><![CDATA[ <cfdi:Comprobante:....>
          </cfdi:Comprobante> ]]></cfdi:DatosXML>
        </cfdi:ComprobanteXML>
        <cfdi:RFC>AAA010101AAA</cfdi:RFC>
        <cfdi:Token>1234</cfdi:Token>
        <cfdi:TransaccionID>0</cfdi:TransaccionID>
      </cfdi:SolicitudTimbraXML>
    </soapenv:Body>
  </soapenv:Envelope>
```

ComprobanteXML: El comprobante en una cadena XML, que contiene la información del Emisor, receptor y conceptos de venta. En algunos casos es necesario agregar el CDATA (Character Data) lo que está dentro de una sección CDATA es ignorado por el intérprete. Una sección CDATA empieza con "<![CDATA [" y termina con "]]>". Esto dentro de la sección DatosXML.

RFC: El RFC de contribuyente.

TransaccionID: El número identificador de la transacción.

Token: Es el token de Transacción de Usuario.

Respuesta:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <RespuestaTimbraXML xmlns="http://Ecodex.WS.Model/2011/CFDI">
      <Advertencias xmlns:i="http://www.w3.org/2001/XMLSchema-instance"/>
      <ComprobanteXML xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <DatosXML><![CDATA[<cfdi:Comprobante></cfdi:Comprobante>]]></DatosXML>
      </ComprobanteXML>
      <TransaccionID>0</TransaccionID>
    </RespuestaTimbraXML>
  </s:Body>
</s:Envelope>
```

CFDi: Es el comprobante en formato de cadena XML, procesado con el Sello y Timbre digital. Al ser enviado con CDATA (Character Data), se recibe en el mismo formato, "<![CDATA [" y termina con "]]>".

TransaccionID: El mismo número de transacción de la solicitud.

Seguridad:

La comunicación de los clientes con ECODEX es llevada a través de canales seguros de comunicación con servicios de encriptación de datos utilizando SSL, asegurando que la información que viaja entre los dos puntos no puede ser conocida por un tercero.

SSL proporciona autenticación y privacidad de la información entre extremos sobre Internet mediante el uso de criptografía. Habitualmente, sólo el servidor es autenticado (es decir, se garantiza su identidad) mientras que el cliente se mantiene sin autenticar.

SSL implica una serie de fases básicas:

- Negociar entre las partes el algoritmo que se usará en la comunicación.
- Intercambio de claves públicas y autenticación basada en certificados digitales.
- Cifrado del tráfico basado en cifrado simétrico.

Manejo de excepciones:

En la mayor parte de todos los métodos, nuestros servicios manejan 3 tipos de excepciones

- **FallaSesion:** Por lo general indica la expiración del token o el uso de uno malformado o que no corresponde al emisor del que se hace la operación.
- **FallaValidacion:** Esta indica algún error en uno o más datos enviados por el usuario. En el caso del timbrado se regresan los errores definidos por el SAT. En el caso del resto de los métodos,
- **FallaServicio:** Excepciones no controladas o que puedan ser causadas por algún problema en el servidor. Se recomienda implementar un método de reintentos en las llamadas.

Estos errores manejan mucha información que puede ayudar a encontrar errores en los métodos del desarrollador, ya que principalmente las excepciones del tipo FallaValidacion, son muy descriptivas sobre lo que pasa en cada llamada.

El manejo de otro tipo de excepciones, como timeouts queda del lado del usuario, implementando el manejo recomendado por su herramienta de programación.

Kit de Pruebas

El objetivo del kit de Integración Ecodex, es ofrecer una demostración de la forma en que el Webservice realiza el sellado y timbrado de los comprobantes.

Proveemos los siguientes ejemplos:

- [Ejemplo C#:](#) Contiene el ejemplo en código C# la solución fue desarrollado utilizando:
 - **Microsoft Visual Studio 2010.**

- [Ejemplo VB.net:](#) Contiene el ejemplo en código Visual Basic.net la solución fue desarrollado utilizando:
 - **Microsoft Visual Studio 2010.**

- [Ejemplo PHP:](#) Contiene el ejemplo en php.

- [Ejemplo PHP:](#) Contiene el ejemplo en Java desarrollado en netbeans.

- [Ejemplo Python:](#) Contiene el ejemplo en python cortesía de Universo Libre ©

Emisor para casos de prueba:

RFC	Número de certificado	Clave de Integrador
AAA010101AAA	20001000000100005867	2b3a8764-d586-4543-9b7e-82834443f219
SUL010720JN8	20001000000200000258	2b3a8764-d586-4543-9b7e-82834443f219

Integrador para alta de emisores:

RFC	Clave de Integrador
BBB010101001	DF627BC3-A872-4806-BF37-DBD040CBAC7C

Direcciones (URLS) de Webservice de pruebas

Las direcciones de nuestros Servicio web (*WebService*) para pruebas son:

Para utilizar:	
El servicio de seguridad <i>ServicioSeguridad.svc</i> :	http://pruebas.ecodex.com.mx:2044/ServicioSeguridad.svc?wsdl
El servicio de Repositorio <i>ServicioRepositorio.svc</i> :	http://pruebas.ecodex.com.mx:2044/ServicioRepositorio.svc?wsdl
El servicio de Solo Timbrado <i>ServicioTimbrado.svc</i> :	http://pruebas.ecodex.com.mx:2044/ServicioTimbrado.svc?wsdl
El servicio de Clientes <i>ServicioClientes.svc</i> :	http://pruebas.ecodex.com.mx:2044/ServicioClientes.svc?wsdl
El servicio de Cancelacion <i>ServicioCancelacion.svc</i> :	http://pruebas.ecodex.com.mx:2044/ServicioCancelacion.svc?wsdl
Para pruebas de los web services SSL:	
El servicio de seguridad <i>ServicioSeguridad.svc</i> :	https://pruebas.ecodex.com.mx:2045/ServicioSeguridad.svc?wsdl
El servicio de Repositorio <i>ServicioRepositorio.svc</i> :	https://pruebas.ecodex.com.mx:2045/ServicioRepositorio.svc?wsdl
El servicio de Solo Timbrado <i>ServicioTimbrado.svc</i> :	https://pruebas.ecodex.com.mx:2045/ServicioTimbrado.svc?wsdl
El servicio de Clientes <i>ServicioClientes.svc</i> :	https://pruebas.ecodex.com.mx:2045/ServicioClientes.svc?wsdl
El servicio de Clientes <i>ServicioCancelacion.svc</i> :	http://pruebas.ecodex.com.mx:2044/ServicioCancelacion.svc?wsdl
Para pruebas Servicio Rest:	http://pruebasapi.ecodex.com.mx

Códigos de Error enviados por SAT

La siguiente es una lista de errores que especifica el SAT no se deben presentar para lograr el sellado y timbrado exitoso de los comprobantes fiscales digitales, se pueden utilizar con los siguientes datos de *Emisor* a fin de probar tanto los certificados como los demás servicios web, **se recomienda utilizar un solo RFC de Receptor para las pruebas que realicen** para fines de poder brindar un mejor soporte.

Código de error del SAT	Mensaje Original de SAT
301	Que no cumpla con el estándar de XML (Conforme al W3C) y con la estructura XML (XSD y complementos aplicables).
302	Que el sello del Emisor no sea válido
303	Que el CSD del Emisor no corresponda al RFC que viene como Emisor en el Comprobante.
304	Que el CSD del Emisor haya sido revocado, utilizando la lista de LCO.
305	Que la fecha de emisión esté fuera de la vigencia del CSD del Emisor.
306	Que la llave utilizada para sellar no corresponda a un CSD (ej. de FIEL).
307	Que contenga un timbre previo.
308	Que el CSD del Emisor no haya sido firmado por uno de los Certificados de Autoridad de SAT.
401	Que el rango de la fecha de generación sea mayor a 72 horas para la emisión del timbre.
402	Que no exista el RFC del emisor conforme al régimen autorizado (Lista de validación de régimen) LCO.
403	Que la fecha de emisión sea anterior al 01 de Enero 2011.

Complementos

Se incluye también dentro del kit, un comprobante con complementos de cada tipo para la validación de elaboración Certificados que soportamos ante el SAT de los siguientes casos, los cuales están en la carpeta Complementos:

- **Detallista.**
- **Divisas.**
- **Donatoria.**
- **EstadoDeCuentaCombustible.**
- **ImpuestosLocales.**
- **Iedu**
- **LeyendasFiscales**
- **Terceros11**
- **Ventavehiculos**
- **Nominas**
- **Servicios parciales de construccion**

Para mayor referencia de cada una consultar en la siguiente página oficial:

http://www.sat.gob.mx/sitio_internet/e_sat/comprobantes_fiscales/15_9255.html

Retenciones:

El servicio de solo timbrado soporta también el timbrado de comprobantes de retenciones con todos sus complementos publicados hasta el momento. Las llamadas no difieren de las de cfdi's y el único cambio es el contenido del archivo enviado. Para más información sobre los esquemas de retenciones consultar:

http://www.sat.gob.mx/informacion_fiscal/factura_electronica/Paginas/cfdi_retenciones_pagos.aspx

Validación de pruebas

El sat proporciona un validador de forma y sintaxis en:

https://www.consulta.sat.gob.mx/sicofi_web/moduloECFD_plus/ValidadorCFDI/Validador%20cfdi.html

Se recomienda implementar validaciones previas al timbrado de los errores más comunes en las pruebas sobre los elementos que puedan ser más susceptibles a capturas erróneas de los usuarios (Rfc's, no de comprobantes, elementos opcionales, etc).

Importante: ya que se libere en ambiente productivo, evite hacer pruebas de facturas por 1 peso. Nuestro servicio garantiza la entrega de comprobantes si se timbro correctamente.

LOCAL WSDL y otros Lenguajes

Para clientes en lenguajes que utilicen el Soap Toolkit e incluso en lenguajes de script como php, python, etc. Se recomienda que bajen el wsdl en modo local, así se evita 1 vuelta al servidor cada vez que se hace una petición y el servicio se hace de un modo más rápido. Los WSDL de pruebas se anexan en el kit de Integración en la Carpeta WSDL. Los WSDL se enviarán a petición del integrador al momento de adquirir sus timbres.

Ligas con los problemas más comunes.

Conversión de bmp a .png en php

<http://www.phpro.org/examples/Convert-BMP-to-JPG.html>

Conversion de encode a utf – 8 en Fox pro

<http://www.west-wind.com/presentations/foxunicode/foxunicode.asp#WhatProblem>

Herramientas Webservice

Soap UI: Herramienta libre para pruebas de web services. Altamente recomendado.

CURL: Herramienta de línea de comando que soporta transferencia de datos con sintaxis url.

[Advanced Rest Client](#): Extensión para Chrome que ayuda en la generación de peticiones de manera sencilla. También disponible para Firefox.

Contacto

Contáctenos para comenzar el proceso de implementación y adquisición de Webservice **WsDex** para el servicio de Sellado y Timbrado para sus comprobantes fiscales digitales.

Desarrollo Corporativo de Negocios en Tecnología de la Información SA de CV

Av. López Mateos Sur 2077 Local Z-16, C.P. 44510

Guadalajara, Jal, México.

Teléfono: (33) 30 30 7258

Lada nacional sin costo: 01800 777 8484

Skype:

- Soporte_ecodex_2
- Soporte_ecodex_3
- Soporte_ecodex_5
- Soporte_ecodex_6

contacto@ecodex.com.mx

Tabla de control de cambios:

Version	Changes
V2.0	New Versión
V2.0.1	Correcciones en base a los comentarios de Andrés Moreno sobre los ejemplos de cancelación y alta de emisores.
V3.0	Agregado de nuevas funcionalidades via JSON
V3.0.1	Actualización de versión del token para el servicio REST