



Manual de timbrado CFDI v 4.0 y Retenciones v 2.0

Contenido

Objetivo	3
Alcance	3
Definiciones	3
Especificaciones del Web Service de Timbrado	4
URL de timbrado para producción (donde el cliente genera su XML sellado):.....	4
URL de timbrado pruebas (donde el cliente genera su XML sellado):.....	4
Códigos de respuesta del timbrado de CFDI 4.0 y Retenciones 2.0.....	6
Ejemplos de consumo SOAP del timbrado de CFDI 4.0	7
Integración	11
Cliente de ejemplo con Axis2 Java	11
Retenciones 2.0	12
URL producción:	12
URL pruebas:	12
Ejemplo de consumo SOAP Retenciones 2.0.	12
Web Service RESTFULL	14
URL ambiente de pruebas:.....	14
Ejemplo de consumo Web Service RESTFULL	14

Objetivo

Proporcionar la información necesaria para que cualquier cliente sea capaz de integrar y consumir nuestro servicio de Timbrado dentro de su sistema.

Alcance

En el presente documento se especifica como consumir el servicio de timbrado de manera directa, puesto que, se encuentra desplegada bajo estándares que garantizan su integración con la mayoría de los lenguajes de programación o tecnologías.

Definiciones

CFDI (*Comprobante Fiscal Digital por Internet*): es como normalmente se le conoce a la factura electrónica. El CFDI funciona como un comprobante que describe un determinado bien o servicio adquirido, la fecha de la transacción, su costo y desglosa los impuestos correspondientes al pago de dicha transacción.

CSD (*Certificado de Sello Digital*): es un documento electrónico mediante el cual el SAT garantiza la vinculación de información entre un contribuyente (persona física o moral) y el SAT, y sirve para firmar sus facturas electrónicas con cualquier Proveedor de Certificación Autorizado (PAC).

SAT (*Servicio de Administración Tributaria*): es un órgano desconcentrado de la Secretaría de Hacienda y Crédito Público, que tiene la responsabilidad de aplicar la legislación fiscal y aduanera, con el fin de que las personas físicas y morales contribuyan proporcional y equitativamente al gasto público.

XML: *Lenguaje Etiquetado Extensible*, un XML en la Factura Electrónica tiene que cumplir con el formato del anexo 20 de la RMF.

WSDL, las siglas de Web Services Description Language, WSDL es una notación XML para describir un servicio web. Una definición WSDL indica a un cliente cómo componer una solicitud de servicio web y describe la interfaz que proporciona el proveedor del servicio web.

Nota: Las pruebas para el timbrado de los XML que se visualizarán a continuación se realizaron en la aplicación Postman for Web Version 9.6.1-211221-0530 Edge 96 / Windows 10

Especificaciones del Web Service de Timbrado

La principal mejora sobre nuestro servicio de timbrado se encuentra en la capacidad de procesar en un mismo endpoint tanto CFDI 4.0 y Retenciones 2.0 sin necesidad de crear clientes separados, así mismo de contar con modalidades tanto SOAP como RESTFULL, es decir, las adecuaciones que se realizaron han sido necesarias para brindar una mejor y más rápida experiencia de integración.

Así mismo se incluyen nuevos parámetros en el Response que permiten brindar trazabilidad en cada petición para facilitar la resolución de incidencias o errores de cada CFDI, también permitirán que usted conozca avisos importantes acerca del servicio que le prestamos.

Nota: Para el timbrado en ambiente de Pre-Producción/Pruebas, se deberá sellar el XML con alguno de los CSD's de pruebas del SAT que corresponda al RFC Emisor, así mismo se deberán utilizar RFC's Emisor y Receptor de Pruebas proporcionados por el SAT (Solicite con nuestro departamento de Atención a Clientes se le proporcionen los CSD's y la RFC de Pruebas). La especificación de nuestro nuevo servicio web en sus diferentes ambientes y modalidades es como se describe a continuación:

URL de timbrado para producción (donde el cliente genera su XML sellado):

<https://appliance.expidetufactura.com.mx:8585/CoreTimbrado.produccion/TimbradoWSSoapSingle?wsdl>

URL de timbrado pruebas (donde el cliente genera su XML sellado):

<https://appliance-qa.expidetufactura.com.mx:8585/CoreTimbrado.test/TimbradoWSSoapSingle?wsdl>

Nota: Debido a las últimas tecnologías implementadas recientemente, solo se aceptan las versiones de TLS1.2 en adelante, dejando las versiones TLS1.0 y TLS1.1 inválidas.

Técnicamente, se debe generar un cliente de nuestro servicio web, en el cual se enviarán los siguientes datos:

Parámetro	Tipo de Dato SOAP	Descripción SOAP	Tipo de Dato REST	Descripción REST	Tipo de Parámetro
Metodo	HttpMethod	Indica qué método acepta la acción pertinente y para estos casos es POST.	String	Indica qué método acepta la acción pertinente y para estos casos es POST.	Entrada
Contentype	Header Content Type	Se debe de indicar el tipo de contenido de la petición y debe de ser "text/xml"	String	Se debe de indicar el tipo de contenido de la petición y debe de ser "application/json"	Entrada
usuario	String	El usuario correspondiente a su cuenta de timbrado, para el caso del ambiente de pre - producción se utiliza el usuario "testUser".	String	El usuario correspondiente a su cuenta de timbrado, para el caso del ambiente de pre - producción se utiliza el usuario "testUser".	Entrada
contrasena	String	La contraseña asignada a la cuenta de timbrado, para el caso del ambiente de pre - producción se hace uso de la contraseña "1234".	String	La contraseña asignada a la cuenta de timbrado, para el caso del ambiente de pre - producción se hace uso de la contraseña "1234".	Entrada
cfdi	Byte	Deberá contener el CFDI a timbrar codificado en base64	String	Deberá contener el CFDI a timbrar codificado en base64	Entrada

Tabla 1. Parámetros de entrada.

La respuesta del servicio web de timbrado contendrá los siguientes datos:

Parámetro	Tipo de Dato	Descripción	Tipo de Parámetro
codigo	String	El valor numérico del código de respuesta generado por la petición enviada.	Salida
mensaje	String	El mensaje asociado al código de respuesta obtenido por la petición enviada.	Salida
timbre	String	El contenido del archivo XML enviado, ya con el timbre fiscal asignado, (para los casos que no sean satisfactorios este campo contendrá la lista de errores que contiene el CFDI enviado mostrando tanto código de error de CFDI v4.0, Retenciones 2.0 y sus complementos).	Salida
uuid	String	El Folio fiscal asignado al comprobante enviado a timbrar.	Salida
hash	String	Cadena original del CFDI cifrada en algoritmo SHA-256.	Salida
transacción	String	Folio del registro de la petición realizada.	Salida
notificacion	String	Mensaje informativo del servicio.	Salida

Tabla 2. Parámetros de salida.

Códigos de respuesta del timbrado de CFDI 4.0 y Retenciones 2.0

A continuación, se describen los códigos de respuesta y mensajes que el servicio de timbrado devuelve:

Nota: La siguiente tabla muestra validaciones previas a la matriz de validación propia del CFDI 4.0 o Complementos, las cuales se deben cumplir antes de pasar por el Core de Timbrado.

Código	Mensaje
200	Proceso satisfactorio
301	XML mal formado
302	Sello mal formado o inválido
303	Sello no corresponde a emisor
304	Certificado revocado o caduco
305	La fecha de emisión no está dentro de la vigencia del CSD del emisor
306	El certificado no es de tipo CSD
307	El CFDI contiene un timbre previo
308	Certificado no expedido por el SAT
401	Fecha y hora de generación fuera de rango
402	RFC del emisor no se encuentra en el régimen de contribuyentes
403	La fecha de emisión no es posterior al 01 de enero de 2012
501	Usuario y/o contraseñas inválidas
502	Usuario no autorizado
503	No hay timbres disponibles
504	Timbrado previamente
500	Intente de nuevo más tarde
505	La transacción HTTP con el servicio de validación no fue satisfactoria (error interno xpd)
CFDI40101	El campo fecha debe corresponder con la hora local donde se expide el comprobante
CFDI40102	El resultado de la digestión debe ser igual al resultado de la descripción del sello
CFDI40106	El certificado no cumple con alguno de los valores permitidos

Tabla 3. Código de Respuesta de timbrado de CFDI

Ejemplos de consumo SOAP del timbrado de CFDI 4.0

En la *Imagen 1* se visualiza el Método de tipo POST y el Content-Type “text/xml”.

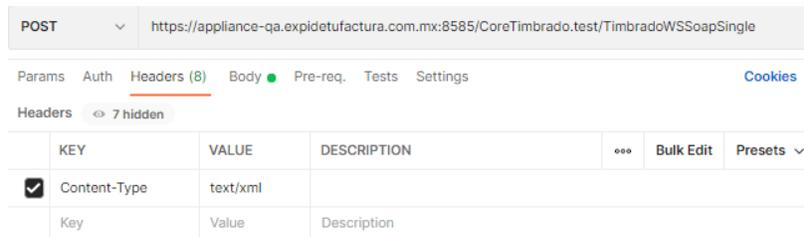


Imagen 1. Headers.

En la *Imagen 2* se visualiza el **código 504 – Ya ha sido timbrado previamente** se presenta cuando un CFDI ha sido enviado con anterioridad, por lo que el archivo no será procesado ni timbrado, pero de igual forma se regresará un timbre que es correspondiente al archivo timbrado por primera vez.

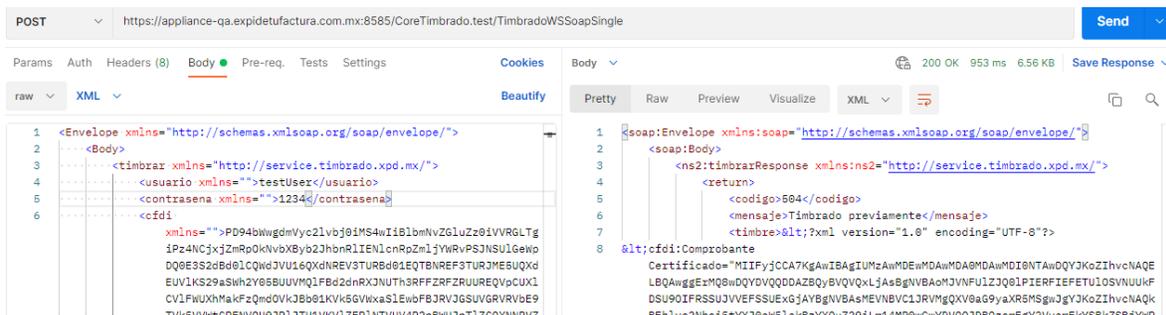


Imagen 2. Prueba de código de error 504 - Timbrado Previamente.

Request ejemplo Prueba de código de error 504. Petición hacia el servicio con los parámetros ya establecidos anteriormente.

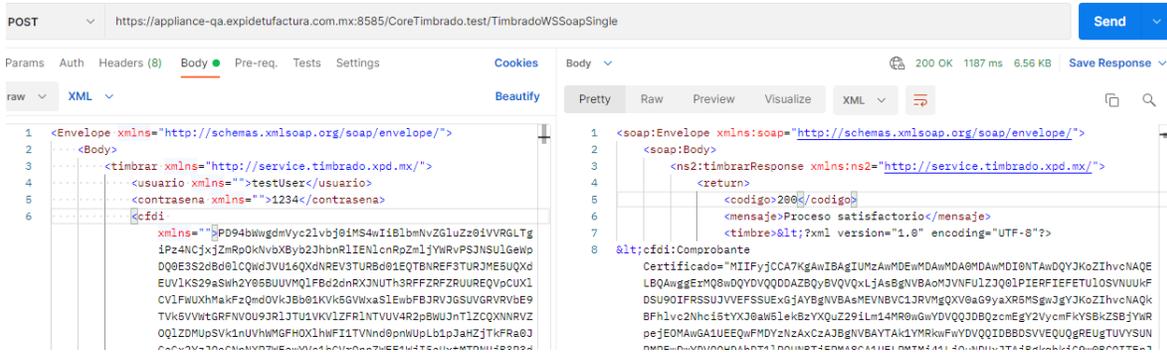
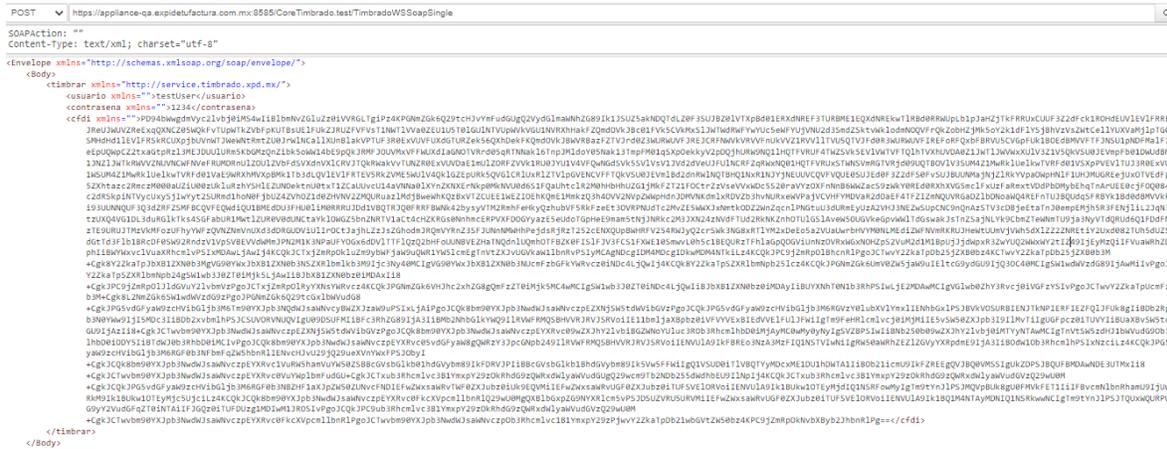
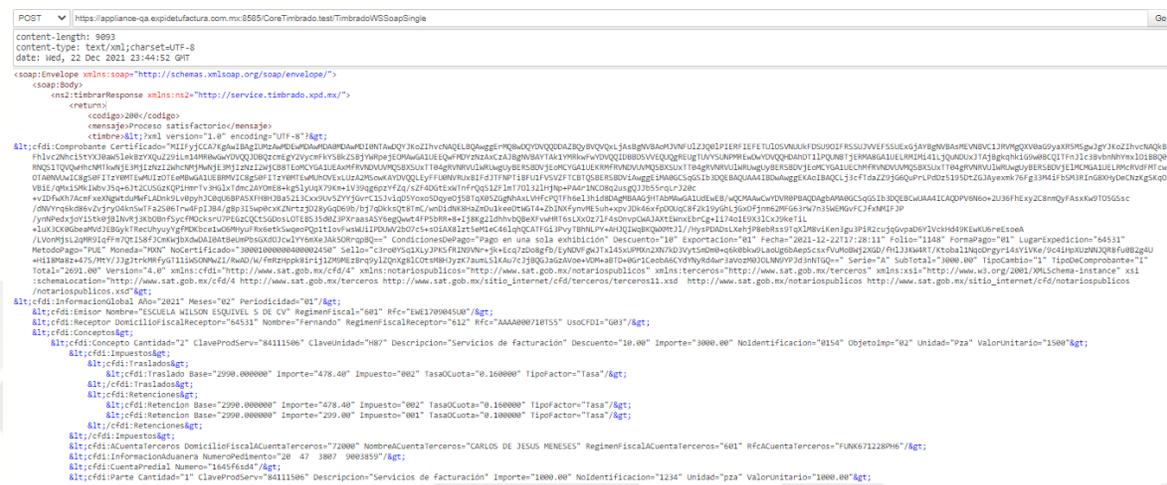


Imagen 3. Prueba de código 200 – Proceso Satisfactorio.

Request ejemplo Prueba de código 200. Petición hacia el servicio con los parámetros ya establecidos anteriormente.



Response ejemplo Prueba de código 200. Respuesta de la solicitud realizada.



Integración

Cliente de ejemplo con Axis 1.4 Java

Para poder realizar el llamado de nuestro servicio web, es necesario contar con una librería para el consumo de servicios SOAP. A continuación, se agrega un ejemplo de la implementación en Java utilizando la librería Axis 1.4.

```
TimbradoWSSoapSingleProxy timbradoProxy = new TimbradoWSSoapSingleProxy();
timbradoProxy.setEndpoint("https://appliance-ga.expiderufactura.com.mx:8585/CoreTimbrado.test/TimbradoWSSoapSingle?wsdl");//pruebas
//timbradoProxy.setEndpoint("https://appliance.expiderufactura.com.mx:8585/CoreTimbrado.produccion/TimbradoWSSoapSingle?wsdl");//produccion

respuesta = timbradoProxy.timbrar("testUser", "1234", cadenaXml.getBytes()); //pruebas
```

Ilustración 1. Ejemplo en Java para consumir servicio web de timbrado.

```
if (respuesta != null) {
    if (!StringUtil.isNullOrEmpty(respuesta.getCodigo()) && !StringUtil.isNullOrEmpty(respuesta.getMensaje())) {
        cadenaXml = null;
        cadenaXml = respuesta.getTimbre();
        log.info(cadenaXml);
        System.out.println(respuesta.getCodigo());
        System.out.println(respuesta.getMensaje());
        System.out.println(respuesta.getUuid());
    }
}
```

Ilustración 2. Ejemplo impresión de respuesta.

En el ejemplo de timbrado con Java (*Ilustración 1*), se muestra cómo se genera una instancia de la clase **TimbradoWSSoapSingleProxy**, posteriormente, se llama al método **timbrar** y la respuesta de este método se asigna a una instancia de la clase **timbradoProxy.timbrar**

Es altamente recomendable tener los valores parametrizados, a fin de cada uno de los valores que se utilicen dentro de su implementación, no se encuentren en “código duro”. El valor para Endpoint deberá ser la URL de nuestro servicio web de timbrado, ya sea el de **pre-producción** o **producción**, el usuario y el password son los valores que se asignan cuando adquiere timbres según el caso y el parámetro xml contiene el CFDI codificado en Base64.

Como buena práctica, es recomendable “capturar” la excepción, a fin de que en caso de que algún error se presente, ya sea en la configuración del cliente o de conexión, este se maneje de forma adecuada.

Retenciones 2.0

El timbrado de Retenciones 2.0 forma parte de nuestro servicio, por lo cual, el URL a utilizar es el presentado anteriormente.

El proceso de integración es igual que en el timbrado, es decir, que los parámetros son los mismos, presentados anteriormente.

URL producción:

<https://appliance.expidetufactura.com.mx:8585/CoreTimbrado.produccion/TimbradoWSSoapSingle?wsdl>

URL pruebas:

<https://appliance-qa.expidetufactura.com.mx:8585/CoreTimbrado.test/TimbradoWSSoapSingle?wsdl>

Ejemplo de consumo SOAP Retenciones 2.0.

En la *Imagen 5* se visualiza el Método de tipo POST y el Content-Type “text/xml”.

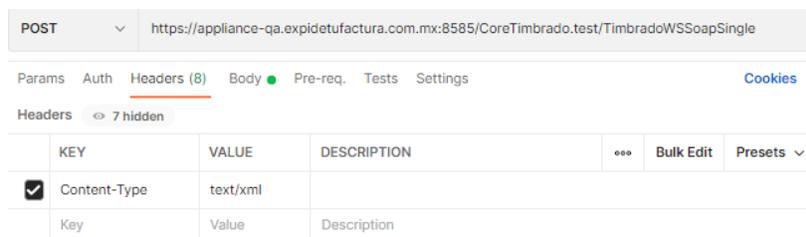


Imagen 5. Retenciones, prueba de código 200 – Proceso satisfactorio.

En la *Imagen 6* se visualiza el **código 200 – Proceso Satisfactorio**.



Imagen 6. Retenciones, prueba de código 200 – Proceso satisfactorio.

Web Service RESTFULL

También se dispone de la modalidad para el Timbrado de CFDI 4.0 y Retenciones 2.0 con sus respectivos complementos mediante un Web Service de tipo RESTFULL, utilizando un Content-Type “application/json” como se ejemplifica a continuación.

URL ambiente de pruebas:

<https://appliance-qa.expidetufactura.com.mx:8585/CoreTimbrado.test/TimbradoWSRest/timbrarCfdiSingle>

Ejemplo de consumo Web Service RESTFULL

En la *Imagen 7* se visualiza el **código 200 – Proceso satisfactorio**.

Nota: Los parámetros resaltados en la siguiente imagen aún no contienen un valor, se espera que para próximas actualizaciones ya se cuente con el uso de estos.

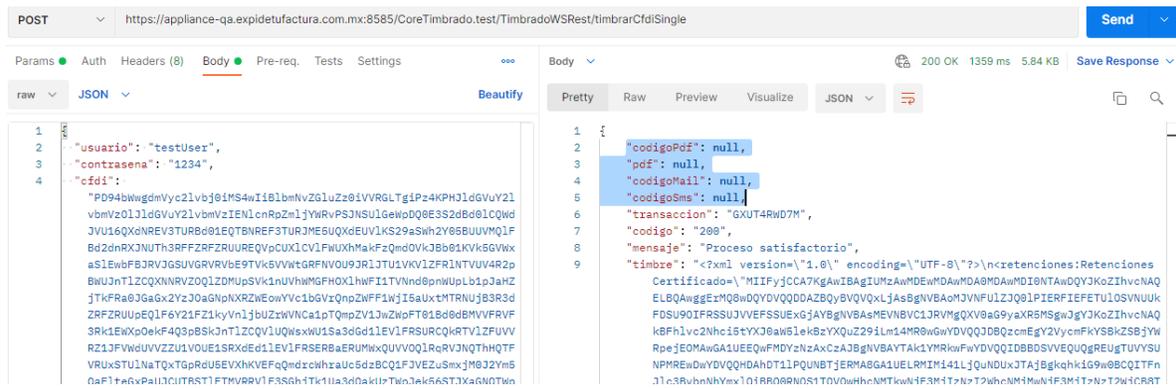
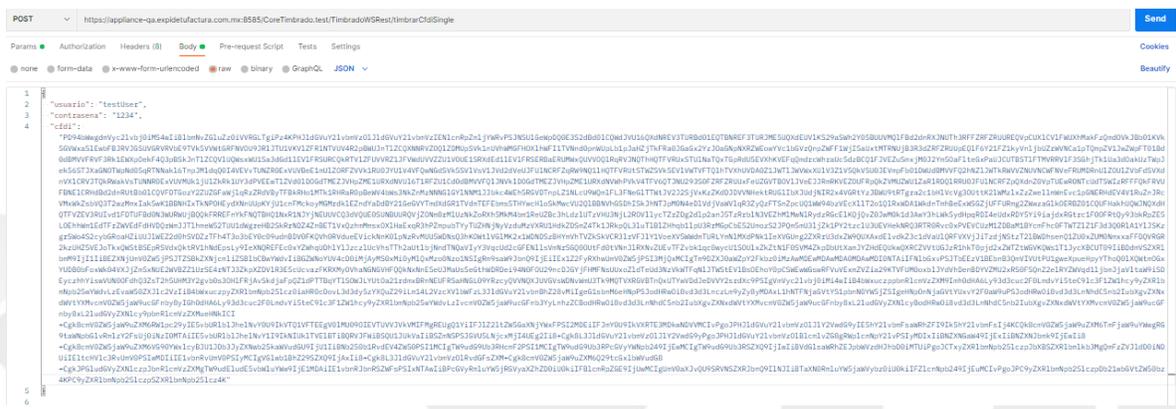


Imagen 7. RESTFULL, prueba de código 200 – Proceso satisfactorio

Request ejemplo RESTFULL, prueba de código 200. Petición realizada con los parámetros mencionados anteriormente.





**Centro Mayor, Torre Alfa, Piso 11.
Oficina 1105. Calzada Zavaleta
1108. Santa Cruz Buenavista
Puebla, Puebla. C.P. 72150.**



Ventas:

01 (222) 2 .48 .74 .85



Soporte Técnico:

01 (222) 2 .26 .77 .24



w w w . x p d . m x