



DOCUMENTO DE INTEGRACION WEB SERVICES

SERVICIO IMPLEMENTACIÓN UBL 2.1

Abril – 2018

Historial de Revisiones

Fecha	Versión	Descripción	Autor
02/04/2018	1.0	Versión Inicial	Pedro Guevara Niño

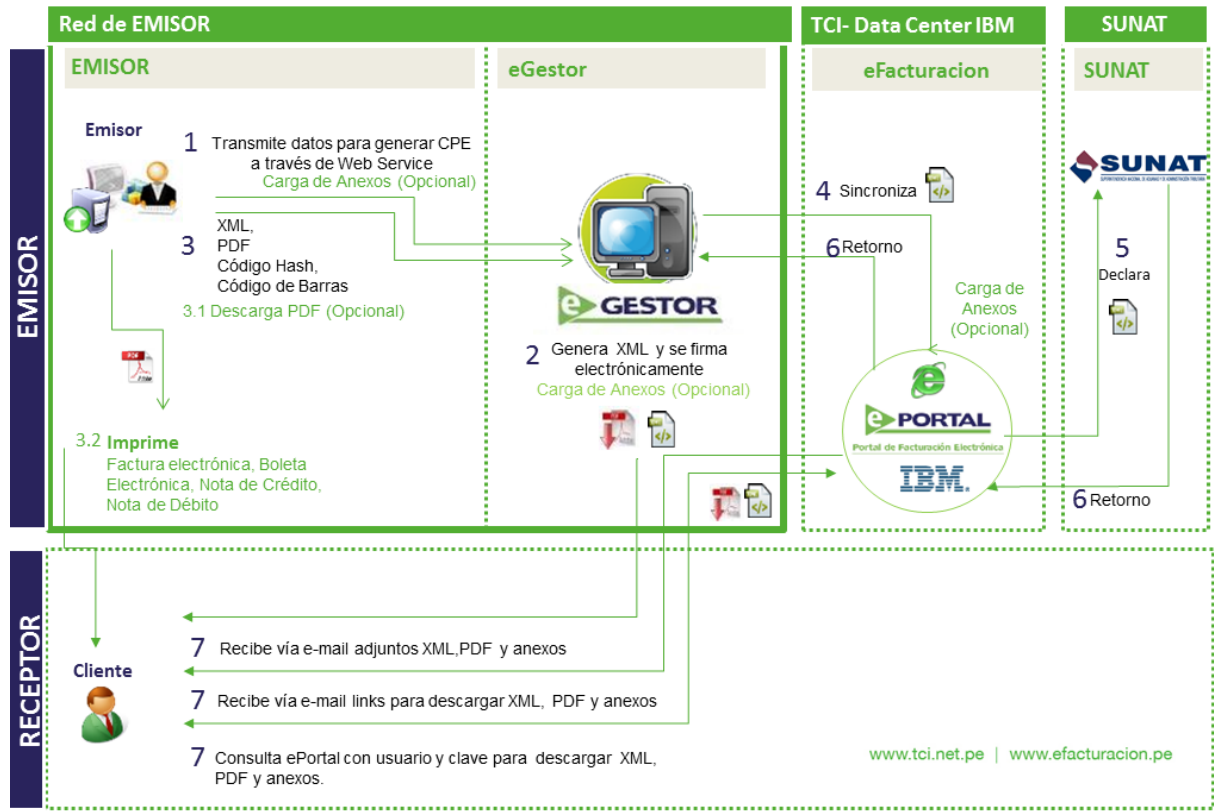
Contenido

1.	Arquitectura de la Solución	5
2.	Métodos de Web Services que cambian para el UBL 2.1	6
3.	Detalle de cada método de Webservices	7
4.1.	Método “Registrar”	7

ÍNDICE DE ANEXOS

Anexo 1. Tecnología webservice a utilizar, “WS” o “WCF”?	9
Anexo 2. URL’s de prueba para integración Mixta y Nube	10
Anexo 3. Uso de SOAP UI	11

1. Arquitectura de la Solución



2. Métodos de Web Services que cambian para el UBL 2.1

Método	Descripción
Registrar	Método por el cual se realiza la generación y firma de XML UBL 2.1 de las Facturas, Boletas, Notas de Crédito y Notas de Débito. Es el mismo método que se utilizó para la integración de UBL 2.0

3. Detalle de cada método de Webservices

Considerar la nomenclatura para los tipos de datos soportados:

a	caracter alfabético
n	caracter numérico
an	carácter alfanumérico
a3	3 caracteres alfabéticos de longitud fija
n3	3 caracteres numéricos de longitud fija
an3	3 caracteres alfa-numéricos de longitud fija
a..3	hasta 3 caracteres alfabéticos
n..3	hasta 3 caracteres numéricos
an..3	hasta 3 caracteres alfa-numéricos

De igual forma cuando se desee establecer la obligatoriedad o no de un determinado elemento de dato se usará la siguiente nomenclatura:

M: Mandatorio u obligatorio

C: Condicional

Repetición:

: Aparece una sola vez.

1...n: Aparece Una vez o más.

0...1: Puede no aparecer, o Aparece una vez

0...n: Puede no aparecer, o puede aparecer una o más veces

En lo referente a la identificación del formato de los elementos de datos se especifica en el cuadro cuando se da las siguientes situaciones:

n(12,2)	elemento numérico hasta 12 enteros+punto decimal+ hasta dos decimales
n(2,2)	elemento numérico hasta 2 enteros+punto decimal+ hasta dos decimales
F###	elemento inicia con la letra F seguida de 3 dígitos
YYYY-MM-DD	formato fecha yyyy=año, mm=mes, dd=día

Colores: Los colores dentro de cada estructura que se muestra a continuación indica la delimitación de los datos de cada propiedad.

4.1. Método “Registrar”

Método: Registrar			
Parámetro de Entrada	Descripción		
oGeneral	Se usa el tipo de dato: General	1..1	Este es el parámetro a variar para la implementación del UBL 2.1

oTipoComprobante	Se usa el tipo de dato: TipoDocumento Factura, Boleta, NotaCredito, NotaDebito Si en caso el campo exige numéricos 1 = Factura 2 = Boleta 3 = NotaCredito 4 = NotaDebito	1..1	
TipoCodigo	Se envia el indicador de tipo de Codigo a Obtener: 0; Codigo de Barras (Formato QR) 1; Codigo Hash	1..1	n1
IdComprobanteCliente	Se envia un valor numérico de 9 dígitos, este valor es retornado en todos los metodos de consulta, con la finalidad de mejorar la identificación de comprobantes generados.	1..1	n..9
Otorgar	Se envia los posibles valores: 0: El comprobante sera otorgado Manualmente. 1: El comprobante sera otorgado automaticamente, este indicador puede tener otros comportamientos dependiendo de las configuraciones realizadas en el Portal.	1..1	an1
Parámetro de Salida		Descripción	
Retorno	Verdadero; Proceso todo correctamente Falso; Proceso presento Inconvenientes, debe revisar el parámetro Cadena .	1..1	bool
Cadena	Parametro de Entrada/ Salida Si Retorno es Verdadero ; Se entrega vacio la variable. Si Retorno es Falso ; Se entrega textualmente el inconveniente por el cual fallo la generación	0..1	an
CodigoBarras	Parametro de Entrada/ Salida El valor lleno de este parametro se identifica en la variable de retorno TipoCodigo , Si se envia el valor 0.	0..1	[argb]
CodigoHash	Parametro de Entrada/ Salida El valor lleno de este parametro se identifica en la variable de retorno TipoCodigo , Si se envia el valor 1.	0..1	an
ListaError	Parametro de Entrada/ Salida Arreglo ENErrorComunicacion	0..n	
.TipoComprobante	Indica el tipo de comprobante por el cual presento inconsistencias	1..1	
.Serie	Indica la Serie del comprobante por el cual presento inconsistencias	1..1	
.Numero	Indica el numero del comprobante por el cual presento inconsistencias	1..1	
.Fecha	Indica la fecha del comprobante por el cual se presento inconsistencias	1..1	
.DescripcionError	Mensaje por el cual presento inconsistencias	1..1	

La estructura del objeto "oGeneral" se encuentra en el archivo "Estructura de datos WS - UBL 2.1.xlsx", tab "WS_MaestroCampos F, B, NC, ND".

Anexos

Anexo 1. Tecnología webservice a utilizar, “WS” o “WCF”?

- Se debe mantener la tecnología que se utilizó para implementar los cambios de la UBL 2.1

Anexo 2. URL's de prueba para integración Mixta y Nube

Ambiente			Portal
UBL21	Web Services	http://egestor.ubl21.efacturacion.pe/WS_TCI/Service.asmx?wsdl	https://cloud.ubl21.efacturacion.pe/appefacturacion/
	WCF	http://egestor.ubl21.efacturacion.pe/WCF_TCI/Service.svc?wsdl	

- Tener en consideración que, si su ERP es SAP y por algún motivo es necesario cambiar la url, debe referenciar nuevamente url.
- Para ambiente productivo utilizarán las mismas urls que vienen utilizando, previa actualización de su servidor eGestor a su última versión, la cual debe contener los cambios de UBL 2.1.
- Durante las pruebas, no se tendrá acceso al eGestor Aplicativo ya que el servicio es remoto. Para que validen sus pruebas deberán ingresar al ePortal.

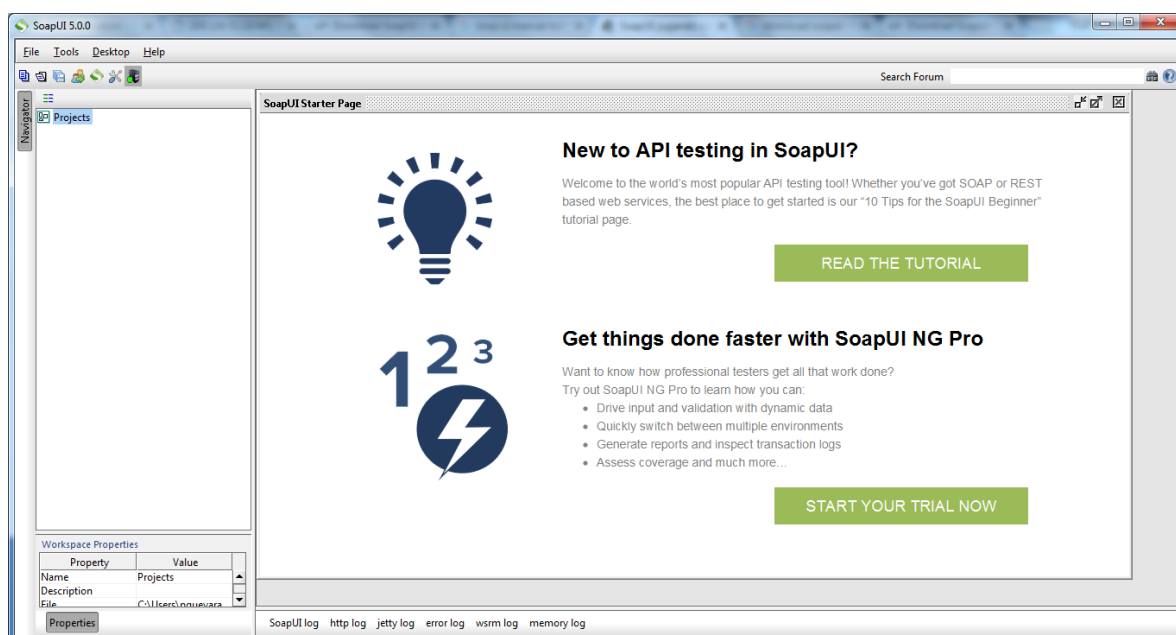
Anexo 3. Uso de SOAP UI

SoapUI es una aplicación muy fácil de usar que permite probar, simular ser un cliente web services y consumir servicios web de forma ágil, partiendo del WSDL.

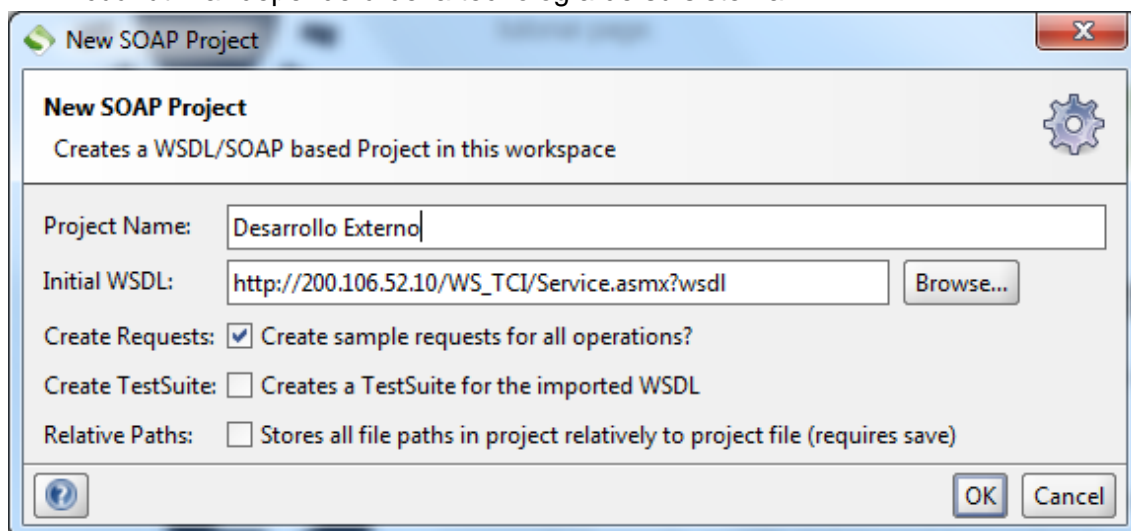
Esta aplicación nos permitirá probar los servicios web del eGestor en su ambiente UBL21.

Este software tiene una versión freeware que se puede descargar de la siguiente página <https://www.soapui.org/downloads/soapui.html>

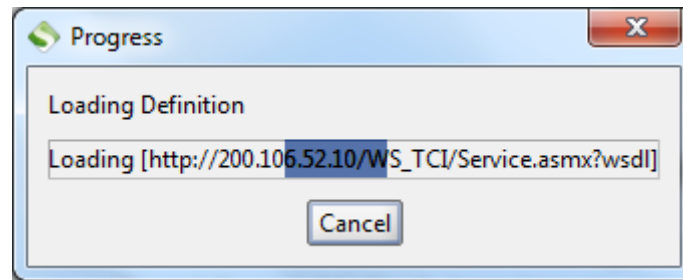
Una vez instalado procedemos a referencia el webservice del eGestor



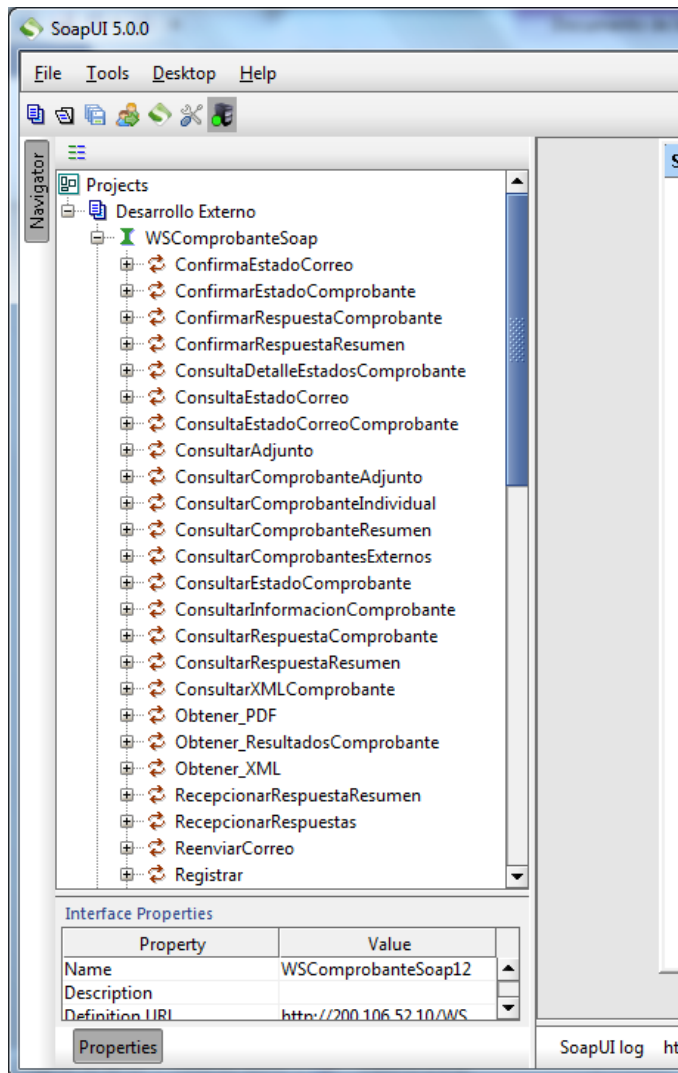
Referenciamos al webservice del eGestor. Para este caso utilizaremos la tecnología WEB SERVICES, también se dispone de la tecnología WCF. Decidir cuál utilizar dependerá de la tecnología de su sistema ERP.



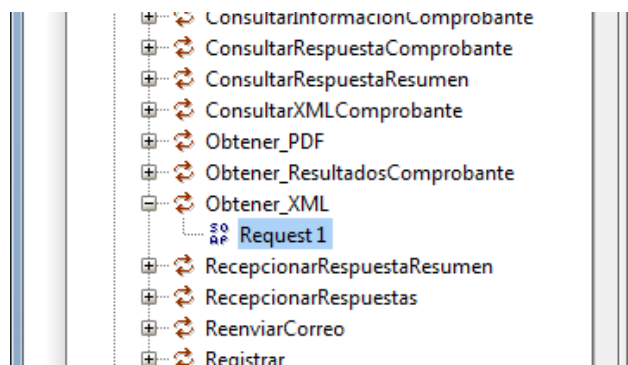
Esperamos un momento



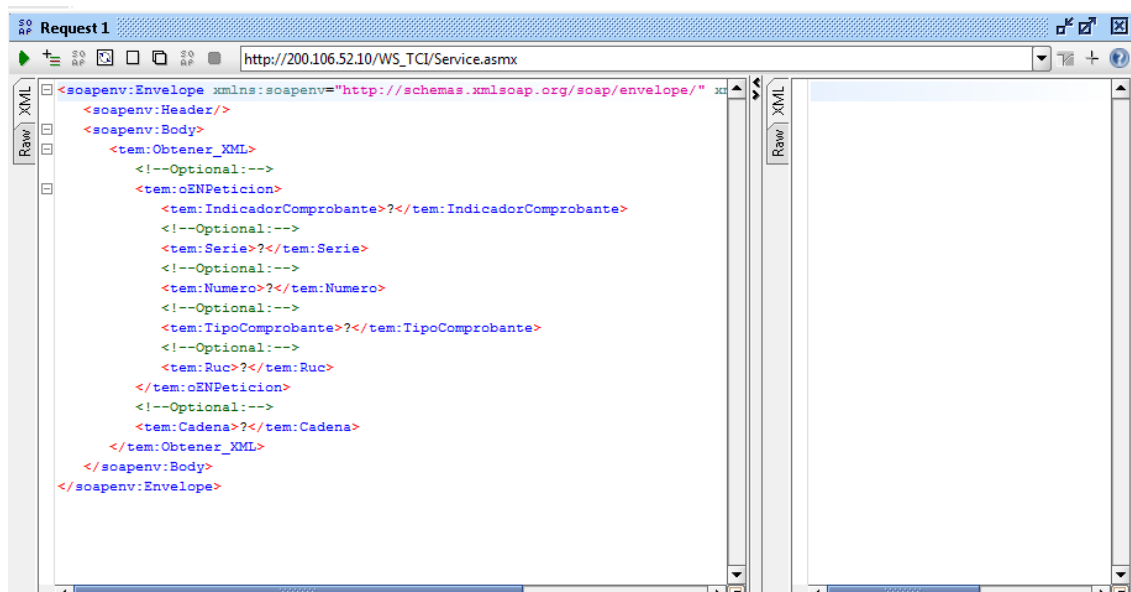
Vemos todos los métodos disponibles en el panel izquierdo



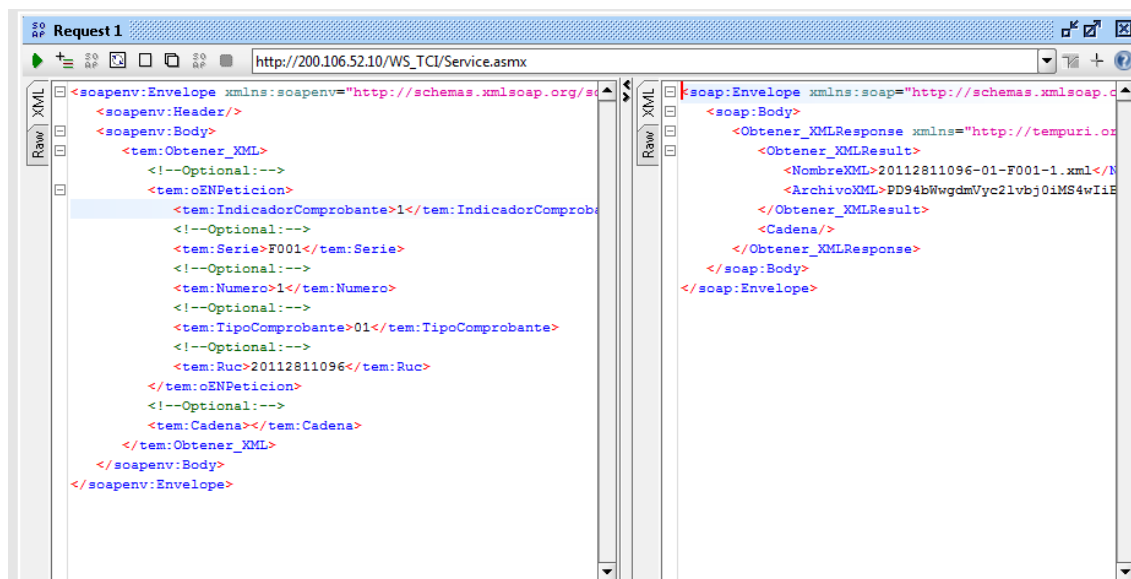
Seleccionamos uno de los métodos más simples de consumir como es el "Obtener_XML"



Damos doble click en la opción “Request 1” y nos aparece la trama SOAP

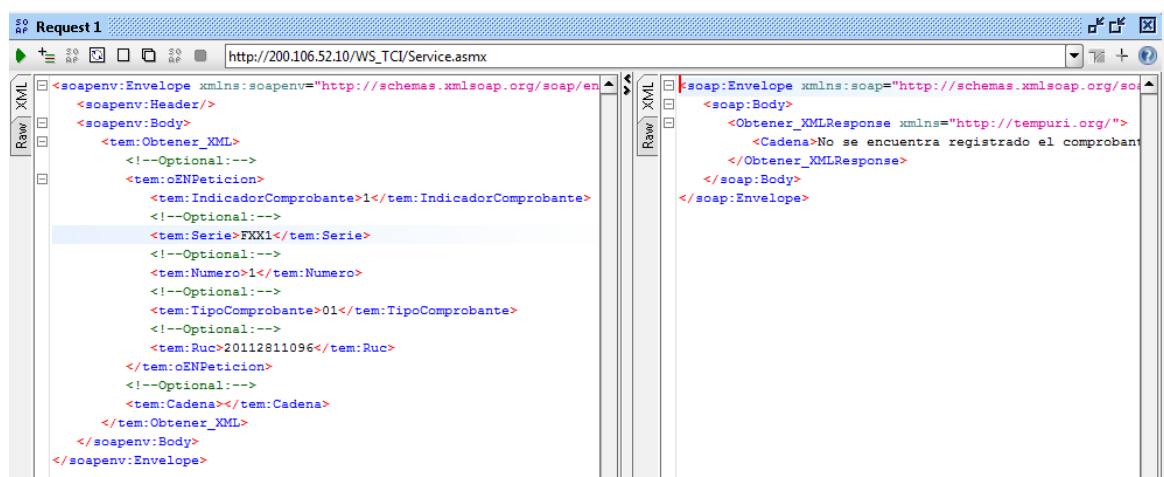


Sustituimos los valores por datos válidos y damos click en el ícono 



Al lado derecho se observa la respuesta del webservice, con esto nos aseguramos que el webservice de TCI está activo y responde correctamente.

Hay que aclarar que este método webservice solo responderá exitosamente si el comprobante ha sido emitido de lo contrario mostrará la siguiente respuesta:



```

Request 1
http://200.106.52.10/WS_TCI/Service.asmx

XML
<?xml version='1.0' encoding='utf-8'>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <tem:Obtener_XML>
      <!--Optional:-->
      <tem:oENPeticion>
        <tem:IndicadorComprobante>1</tem:IndicadorComprobante>
        <!--Optional:-->
        <tem:Serie>FXK1</tem:Serie>
        <!--Optional:-->
        <tem:Numero>1</tem:Numero>
        <!--Optional:-->
        <tem:TipoComprobante>01</tem:TipoComprobante>
        <!--Optional:-->
        <tem:Ruc>20112811096</tem:Ruc>
      </tem:oENPeticion>
      <!--Optional:-->
      <tem:Cadena></tem:Cadena>
    </tem:Obtener_XML>
  </soapenv:Body>
</soapenv:Envelope>

XML
<?xml version='1.0' encoding='utf-8'>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Obtener_XMLResponse xmlns="http://tempuri.org/">
      <Cadena>No se encuentra registrado el comprobante</Cadena>
    </Obtener_XMLResponse>
  </soap:Body>
</soap:Envelope>
  
```