

## Librería javaWSAfip - Webservice wsfeV1

### Obtener Sign y Token y Obtener CAE

En este tutoria, se describen todos los metodos y clases referente a esta librería.

Para comenzar es importante tener en cuenta lo siguiente: La AFIP permite utilizar sus webservices de dos modos:

**Modo Homologación o Test:** en este modo, ud. Podrá probar la librería y realizar su proyecto utilizando los servicios de prueba de la AFIP. Las facturas generadas como el CAE otorgado no tienen validez fiscal.

**Modo Producción:** Una vez que realizo su aplicación y probó que todo funciona correctamente, puede activar este modo. Aquí Sí las facturas generadas como el CAE otorgado tienen validez fiscal.

Hay que tener en cuenta que en modo Homologación, se utiliza un certificado creado desde afip exclusivamente para este modo. Ese certificado no está autorizado para usarlo en modo Producción.

***Ahora vamos a nuestra librería:***

**javaWSAfip consta de lo siguiente:**

- **Directorios obligatorios que deben existir en la raiz de su proyecto y no se deben modificar:**

/certs/

/credentials/

En el directorio “certs” se tendrá que colocar el certificado con extensión .pfx que utilizará la librería.

En el directorio “credentials” estarán los archivos xml del token y sign, como así también las respuestas correspondientes a la autorizacion del ticket de acceso.

- **Archivo de configuración:**

En el directorio raiz de su proyecto debe estar este archivo: **configuracion.cfg**

Este archivo es muy importante ya que de él depende el funcionamiento de la librería, en él se deben configurar el nombre del certificado .pfx generado por Uds., la clave que utilizaron cuando generaron dicho certificado.

### Contenido del archivo *configuracion.cfg*

# -----Datos para usar en modo de produccion

# ---La clave del producto sera remitida por nosotros cuando compre nuestra librería  
*Estos campos no se deben modificar salvo que quiera utilizar la librería en modo producción.*

produccion=no

mail=

password=

# -----Servicio para el cual se requiere el ticket de acceso y padron

*Estos campos no se deben modificar.*

service=wsfe

serviceP=ws\_sr\_padron\_a5

# -----Destino del Servicio-----

*Si utiliza el modo Producción se deben quitar los # y ponerlos en dstdn y dstdnP en el apartado "Para Test".*

# --Para produccion

#dstdn=cn=wsaa,o=afip,c=ar,serialNumber=CUIT 33693450239

#dstdnP=cn=wsaa,o=afip,c=ar,serialNumber=CUIT 33693450239

# --Para Test

dstdn=cn=wsaahomo,o=afip,c=ar,serialNumber=CUIT 33693450239

dstdnP=cn=wsaahomo,o=afip,c=ar,serialNumber=CUIT 33693450239

# -----Keystore en formato PKCS#12 que posee el certificado del computador que solicita el acceso

*Aquí debe estar el nombre del certificado generado por Ud, debe estar en el directorio /certs/.*

keystore=certs/22433523.pfx

# -----Signer del certificado en el keystore-----

*No se debe modificar.*

keystore-signer=1

# -----Clave de acceso al keystore-----

*Aquí debe colocar la clave que ingresó cuando generó el certificado .pfx*

keystore-password=12345

# -----Tiempo de vida del ticket requerido

*No se debe modificar.*

TicketTime=36000

## Métodos y Funciones de nuestra Librería

Antes que nada, se deben instalar en su proyecto las librerías que se encuentran en el directorio “lib” del código que Ud. bajó.

Luego importar las clases en su proyecto:

```
import javaWSAfip.wsaa;  
import javaWSAfip.wsaa_padron;  
import javaWSAfip.wsfe;
```

El código de ejemplo que Ud. bajó tiene en su directorio “lib” las librerías requeridas y en el directorio “certs” un certificado \*.pfx de prueba brindado por nosotros. El programa ya viene configurado para usar ese certificado. Si quiere usar el suyo de prueba, debe modificar el archivo configuracion.cfg como se indicó en la página anterior. Recuerde que los certificados .pfx debe ser incluidos en el directorio “certs”.

### **Obtener Sign y Token:**

Lo más importante para generar una factura electrónica es obtener el CAE y su fecha de Vencimiento. Para lograr esto, la AFIP requiere primero un ticket de acceso que debemos enviar nosotros, si todo está correctamente, ellos nos devuelven un Sign y un Token. Estos son esenciales para poder generar un CAE y realizar diferentes consultas a sus webservices. Para lograr esto, primero se debe invocar al servicio de autorización “wsaa” que es el que valida nuestro requerimiento y nos envía un Sign y un Token. Teniendo en cuenta esto, lo primero que se debe invocar entonces antes que nada es al método:

```
wsaa.wsaa_autorizacion();
```

Este método almacena en **wsaa.autorizacion\_xml** un XML con los resultados de la autorización. Este XML se puede consultar en caso que haya errores y querramos saber cuales fueron.

Para consultar ese XML de respuesta, Ud. lo puede hacer de la siguiente manera:

```
System.out.println("Respuesta de Autorizacion: " + wsaa.autorizacion_xml);
```

No es necesario consultarlo, pero sirve para saber en caso de errores, cuáles fueron.

Si todo fue Ok, en el directorio /credentials/ se habrán guardado el Sign y el Token devuelto por AFIP. No nos tenemos que preocupar por estos archivos ya que la librería se encarga de leerlos automáticamente cuando los requiera.

## Obtención del CAE

Para obtener el CAE necesitamos realizar en forma ORDENADA una serie de pasos.

### • Paso 1: Completar las líneas del comprobante

Hay dos maneras de completar las líneas del comprobante:

**Para Responsable Inscripto, si Ud. emite Comprobantes A y B:**

Campos importantes:

Se debe especificar la alícuota del IVA del producto "sin el signo %"

- `wsfe.Lineas_Alic_IVA="21";`

Se debe especificar el precio del producto SIN el IVA

- `wsfe.Lineas_Subtotal=100;`

Se debe especificar el precio del producto CON el IVA INCLUIDO

- `wsfe.Lineas_Total=121;`

Cada vez que se completa una línea se ejecuta este método, el (1) es la línea que agregamos, si agregamos otra línea, agregamos este método con (2), y así sucesivamente dependiendo de las líneas que agreguemos.

- `wsfe.lineasCargar_I(1);`

#### Ejemplo para comprobantes "A" y "B":

// Primer línea del comprobante

```
wsfe.Lineas_Codigo="1";           // el código del producto si lo quiere ingresar
wsfe.Lineas_Desc="Producto 1";    // la descripción del producto
wsfe.Lineas_Cant=1;               // la cantidad de productos
wsfe.Lineas_PU=10;               // el Precio Unitario del producto
wsfe.Lineas_Alic_IVA="21";        // La alícuota del IVA de ese producto, en este caso el 21%
wsfe.Lineas_Subtotal=100;         // El subtotal de ese producto "precio sin iva"
wsfe.Lineas_Total=121;           // El Total de ese producto "precio con iva incluido"
wsfe.lineasCargar_I(1);          // Agrega la línea "1"
```

// Segunda línea del comprobante:

```
wsfe.Lineas_Codigo="2";           // el código del producto si lo quiere ingresar
wsfe.Lineas_Desc="Producto 2";    // la descripción del producto
wsfe.Lineas_Cant=10;             // la cantidad de productos
wsfe.Lineas_PU=100;              // el Precio Unitario del producto
wsfe.Lineas_Alic_IVA="10.5";      // La alícuota del IVA de ese producto, en este caso el 10.5%
wsfe.Lineas_Subtotal=100;         // El subtotal de ese producto "precio sin iva"
wsfe.Lineas_Total=110.5;         // El Total de ese producto "precio con iva incluido"
wsfe.lineasCargar_I(2);          // Agrega la línea "2"
```

## **Para Monotributistas o Exentos, si Ud. emite Comprobantes "C":**

Campos importantes:

Al ser comprobantes tipo "C", en la alícuota del IVA se debe colocar un "0" cero

- wsfe.Lineas\_Alic\_IVA="0";

Se debe especificar el precio del producto.

- wsfe.Lineas\_Subtotal=100;

Se debe especificar el precio del producto. El mismo que el Subtotal

- wsfe.Lineas\_Total=100;

### **Ejemplo para comprobantes "C":**

// Primer línea de la factura

```
wsfe.Lineas_Codigo="1";           // el código del producto si lo quiere ingresar
wsfe.Lineas_Desc="Producto 1";   // la descripción del producto
wsfe.Lineas_Cant=1;              // la cantidad de productos
wsfe.Lineas_PU=10;               // el Precio Unitario del producto
wsfe.Lineas_Alic_IVA="0";        // El IVA de ese producto, debe ser "0" cero
wsfe.Lineas_Subtotal=100;        // El precio del producto
wsfe.Lineas_Total=100;           // El precio del producto (el mismo que el Subtotal)
wsfe.lineasCargar_I(1);          // Agrega la línea "1"
```

// Segunda línea de la factura

```
wsfe.Lineas_Codigo="2";           // el código del producto si lo quiere ingresar
wsfe.Lineas_Desc="Producto 2";   // la descripción del producto
wsfe.Lineas_Cant=10;              // la cantidad de productos
wsfe.Lineas_PU=100;               // el Precio Unitario del producto
wsfe.Lineas_Alic_IVA="0";        // El IVA de ese producto, debe ser "0" cero
wsfe.Lineas_Subtotal=250;        // El precio del producto
wsfe.Lineas_Total=250;           // El precio del producto (el mismo que el Subtotal)
wsfe.lineasCargar_I(2);          // Agrega la línea "2"
```

### **Agregado de Tributos o Impuestos del comprobante si es necesario:**

Los campos que se deben modificar son los siguientes:

- wsfe.Tributo\_codigo="1"; Se debe especificar el código del tributo o impuesto.
- wsfe.Tributo\_Desc="Impuestos Nacionales"; Se especifica la descripción del tributo
- wsfe.Tributo\_alicuota=5.2; Se especifica la Alícuota del tributo "sin el signo %"
- wsfe.tributosCargar(1); Agrega la línea del tributo

### **Ejemplo de Tributos:**

```
wsfe.Tributo_codigo="1";  
wsfe.Tributo_Desc="Impuestos Nacionales";  
wsfe.Tributo_baseImp=wsfe.totlizar_neto(); //la Base imponible es el total de los importes netos de la factura  
wsfe.Tributo_alicuota=5.2;  
wsfe.Tributo_Importe=wsfe.fd((wsfe.totlizar_neto()*(wsfe.Tributo_alicuota/100),2);  
wsfe.tributosCargar(1);
```

Nota: Se pueden agregar más tributos de la misma manera, en este caso si se agregara otro, seria wsfe.tributosCargar(2), el "2" por tratarse del segundo tributo, y así sucesivamente.

***Los códigos de tipos de tributos podrán consultarlos en la correspondiente tabla al final de este documento.***

### **Emisión de Notas de Crédito o Débito**

Cuando se quiera emitir Notas de Crédito o Débito, **la AFIP obliga incluir un comprobante asociado**, es decir, si se emite una Nota de Crédito, el comprobante asociado debe ser a una Factura ya emitida.

Cuando hayamos completado las línea de la Nota de Crédito o Débito, de la misma manera que se realiza en las facturas, se debe agregar el comprobante asociado.

La manera de hacerlo es la siguiente.

#### **Agregar comprobante asociado:**

```
wsfe.compAsoc_tipo="1"; //En este caso estamos asociando a una Factura "A"  
wsfe.compAsoc_ptovta="2"; //El punto de venta de esa factura  
wsfe.compAsoc_nro="60"; //El número de esa factura a la cual se esta asociando  
wsfe.compAsoc_cuit="27077704257"; //El CUIT del cliente de esa factura  
wsfe.compAsoc_fecha="20210722"; //La fecha de esa factura  
wsfe.compAsocCargar(1); //Se agrega el comprobante asociado
```

## • Paso 2: Completar datos principales del comprobante

En este paso, una vez completado en el paso N° 1 las líneas del comprobante, se deben completar los datos principales del comprobante que se va a enviar para su autorización.

**Antes que nada, es importante obtener el número del último comprobante emitido para que el que se va a emitir continúe con la numeración correlativa.**

**wsfe.wsfe\_UltCompr(CUIT del emisor, Punto de Venta, Código del Comprobante);**

Esta función devuelve un entero con el último número del comprobante anteriormente emitido.

**Código de ejemplo:**

//Obtengo el último número del comprobante anteriormente emitido

```
int resp_ultcompr = wsfe.wsfe_UltCompr("20224335238",2,1);
```

```
wsfe.CantReg = 1; //no se debe cambiar
```

```
wsfe.CbteTipo = 1; //Codigo de Factura A – Debe ser el código correspondiente al comprobante
```

```
wsfe.PtoVta = 2; //punto de venta del comprobante a emitir
```

```
wsfe.Concepto = 1; //1: Productos - 2: Servicios - 3: Productos y Servicios
```

```
wsfe.DocTipo = 80; //Tipo de documento CUIT – Código del CUIT
```

```
wsfe.DocNro = "20077704257"; //CUIT del emisor del comprobante
```

```
wsfe.CbteDesde = resp_ultcompr+1; //numero correlativo del comprobante a generar
```

```
wsfe.CbteHasta = resp_ultcompr+1;
```

```
wsfe.CbteFch = "20210918"; //fecha del comprobante formato AAAAMMDD
```

```
wsfe.FchServDesde = ""; //se debe especificar en el caso que Conceptos sean 2 o 3, Servicios o Productos y Servicios
```

```
wsfe.FchServHasta = ""; //se debe especificar en el caso que Conceptos sean 2 o 3, Servicios o Productos y Servicios
```

```
wsfe.FchServVto = ""; //Fecha de Vencimiento. Se debe especificar en el caso que Conceptos sean 2 o 3, Servicios o Productos y Servicios
```

```
wsfe.ImpOpEx = 0;
```

```
wsfe.ImpTotConc = 0;
```

```
wsfe.ImpTotal = wsfe.totlizar_total()+wsfe.ImpOpEx+wsfe.ImpTotConc; //no se debe cambiar
```

```
wsfe.ImpNeto = wsfe.totlizar_neto(); //no se debe cambiar
```

```
wsfe.ImpTrib = wsfe.totlizar_tributos(); //no se debe cambiar
```

```
wsfe.ImpIVA = wsfe.totlizar_ivas(); //no se debe cambiar
```

```
wsfe.MonId = "PES"; //no se debe cambiar
```

```
wsfe.MonCotiz = 1; //no se debe cambiar
```

**Los Códigos de: Tipos de Comprobantes para “wsfe.CbteTipo”, Conceptos para “wsfe.Concepto”, Tipo de Documento para “wsfe.DocTipo”, Tipo de Moneda para “wsfe.MonId” podrán consultarlos en las correspondientes tablas al final de este documento.**

## • Paso 3: Obtención del CAE

Finalmente luego de **completar los Pasos 1 y 2**, estamos en condiciones de obtener el CAE.

La función que se encarga de esto es:

**wsfe.wsfe\_FECAESolicitar(CUIT del Emisor);**

Esta función devuelve un XML con la respuesta completa que envía la AFIP, el cual tiene el CAE, el vencimiento del CAE y la aprobación si fue Aprobada o Rechazada.

Este XML no es necesario consultarlo, depende de Ud. si quiere tener una información del comprobante enviado y sus errores si los hay.

Luego de esta llamada, Ud. tendrá los siguientes datos en formato String:

**wsfe.CAE** //Se encuentra el CAE otorgado

**wsfe.CAE\_Vencimiento** // Se encuentra la fecha de vencimiento del CAE en formato AAAAMMDD

**wsfe.Resultado** // Se encuentra la aprobación del requerimiento. Si es "A" está aprobado, si es "R" fue rechazado. Si fue rechazado, el CAE como su fecha de vencimiento es NULO.

Los errores, en caso de haberlos, se almacenan en el Vector:

**wsfe.WSFE\_Errores <String>**

Podrá recorrerlo para capturar los errores que hubieron y saber por qué fue rechazado el comprobante. Se debe hacer de esta manera:

//Compruebo si hay errores

```
if (wsfe.WSFE_Cantidad_Errores!=0){  
    for (int i = 0; i < wsfe.WSFE_Cantidad_Errores; i++){  
        System.out.println("Errores: "+wsfe.WSFE_Errores.get(i));  
    }  
}
```

Con este código, listaremos el o los errores que pudieran haber habido.



## Librería javaWSAfp - Webservice wsfeV1

### Otras funciones importantes de nuestra librería

#### **Consultar un Comprobante Previamente Generado por Ud.:**

**wsfe\_ConsultarComprobanteEmitido(CUIT, TipoComprobante, NumComprobante, PtoVenta)**

**CUIT:** Cuit del emisor del comprobante (String)

**TipoComprobante:** Código del tipo de comprobante (Int)

**NumComprobante:** El número del comprobante (Int)

**PtoVenta:** Punto de venta del comprobante (Int)

Esta función devuelve un XML en formato String con los datos del comprobante consultado.

#### **Ejemplo de código:**

```
String resp_compro_xml = "";  
resp_compro_xml = wsfe.wsfe_ConsultarComprobanteEmitido("20224335238", 1, 224, 2);  
System.out.println("-----XML CONSULTA DE UN COMPROBANTE AUTORIZADO-----");  
System.out.println(resp_compro_xml);
```

---

#### **Consultar Tipos de Comprobantes Permitidos para el CUIT del emisor:**

**wsfe.wsfe\_ConsultarTiposComprobantes(CUIT)**

**CUIT:** Cuit del emisor del comprobante (String)

Esta función devuelve un XML en formato String con todos los comprobantes que Ud. está autorizado a emitir.

#### **Ejemplo de código:**

```
String resp_tipos_compro_xml = "";  
resp_tipos_compro_xml = wsfe.wsfe_ConsultarTiposComprobantes("20224335238");  
System.out.println("-----XML TIPOS DE COMPROBANTES PERMITIDOS-----");  
System.out.println(resp_tipos_compro_xml);
```

---

### **Consultar Tipos de Documentos con su Código correspondiente:**

**wsfe.wsfe\_consultarTiposDocumento("20224335238")**

**CUIT:** Cuit del emisor del comprobante (String)

Esta función devuelve un XML en formato String con todos los tipos de documentos con su correspondiente código. Por ejemplo, el código correspondiente al CUIT es 80, el código correspondiente al DNI es 96, etc.

#### **Ejemplo de código:**

```
String resp_tipos_documentos_xml = "";  
System.out.println("-----XML TIPOS DE DOCUMENTOS-----");  
resp_tipos_documentos_xml = wsfe.wsfe_consultarTiposDocumento("20224335238");  
System.out.println(resp_tipos_documentos_xml);
```

---

### **Consultar Alícuotas de IVA:**

**wsfe.wsfe\_consultarAlicuotasIVA(CUIT)**

**CUIT:** Cuit del emisor del comprobante (String)

Esta función devuelve un XML en formato String con todas la alícuotas de IVA.

#### **Ejemplo de código:**

```
String resp_alicuotas_IVA_xml = "";  
System.out.println("-----XML ALICUOTAS IVA-----");  
resp_alicuotas_IVA_xml = wsfe.wsfe_consultarAlicuotasIVA("20224335238");  
System.out.println(resp_alicuotas_IVA_xml);
```

---

### **Consultar Códigos de Tipos de Monedas:**

**wsfe.wsfe\_consultarMonedas("20224335238")**

**CUIT:** Cuit del emisor del comprobante (String)

Esta función devuelve un XML en formato String con todos los tipos de monedas con su correspondiente código. Por ejemplo para Pesos "PES", Dólar "DOL", etc.

#### **Ejemplo de código:**

```
String resp_monedas_xml = "";  
System.out.println("-----XML MONEDAS-----");  
resp_monedas_xml=wsfe.wsfe_consultarMonedas("20224335238");  
System.out.println(resp_monedas_xml);
```

---

### **Consultar Cotización Actualizada de las Monedas:**

**wsfe.wsfe\_consultarCotizacionMoneda(CUIT, Código de Moneda)**

**CUIT:** Cuit del emisor del comprobante (String)

**Código de Moneda:** Es el código de moneda a consultar, ej. para el Dólar "DOL" (String)

Esta función devuelve un XML en formato String todas las cotizaciones actualizadas de cada moneda.

#### **Ejemplo de código:**

```
String resp_cotizacion_monedas_xml = "";  
System.out.println("-----XML COTIZACION MONEDAS-----");  
resp_cotizacion_monedas_xml = wsfe.wsfe_consultarCotizacionMoneda("20224335238", "DOL");  
System.out.println(resp_cotizacion_monedas_xml);
```

---

### **Consultar Puntos de Venta Habilitados para la facturación electrónica:**

**wsfe.wsfe\_PuntosdeVentaHabilitados(CUIT)**

**CUIT:** Cuit del emisor del comprobante (String)

Esta función devuelve un XML en formato String todos los puntos de ventas habilitados para la facturación electrónica. **Este método solo funciona para Producción. En modo Test no funciona.**

#### **Ejemplo de código:**

```
String resp_puntos_de_venta_autoriz_xml = "";  
System.out.println("-----XML PUNTOS DE VENTA AUTORIZADOS-----");  
resp_puntos_de_venta_autoriz_xml = wsfe.wsfe_PuntosdeVentaHabilitados("20224335238");  
System.out.println(resp_puntos_de_venta_autoriz_xml);
```

---

### **Consultar Tipos de Tributos con sus correspondientes Códigos:**

**wsfe.wsfe\_TiposdeTributos(CUIT)**

**CUIT:** Cuit del emisor del comprobante (String)

Esta función devuelve un XML en formato String todos los tributos con sus correspondientes códigos.

#### **Ejemplo de código:**

```
String resp_tipos_tributos_xml = "";  
System.out.println("-----XML TIPOS DE TRIBUTOS-----");  
resp_tipos_tributos_xml = wsfe.wsfe_TiposdeTributos("20224335238");  
System.out.println(resp_tipos_tributos_xml);
```

---

### **Consultar Tipos de Conceptos:**

**wsfe.wsfe\_TiposdeConceptos(CUIT)**

**CUIT:** Cuit del emisor del comprobante (String)

Esta función devuelve un XML en formato String todos los Conceptos con sus correspondientes códigos. Ej.: Productos (código 1), Servicios (código 2), Productos y Servicios (código 3).

#### **Ejemplo de código:**

```
String resp_tipos_de_concepto_xml = "";  
System.out.println("-----XML TIPOS DE CONCEPTOS-----");  
resp_tipos_de_concepto_xml = wsfe.wsfe_TiposdeConceptos("20224335238");  
System.out.println(resp_tipos_de_concepto_xml);
```

## Librería javaWSAfp - Webservice wsfeV1

### TABLA de Códigos de Tipos de Comprobantes

Código	Descripción
1	FACTURAS A
2	NOTAS DE DEBITO A
3	NOTAS DE CREDITO A
4	RECIBOS A
5	NOTAS DE VENTA AL CONTADO A
6	FACTURAS B
7	NOTAS DE DEBITO B
8	NOTAS DE CREDITO B
9	RECIBOS B
11	FACTURAS C
12	NOTAS DE DEBITO C
13	NOTAS DE CREDITO C
15	RECIBOS C
51	FACTURAS M
52	NOTAS DE DEBITO M
53	NOTAS DE CREDITO M
54	RECIBOS M

## Librería javaWSAfp - Webservice wsfeV1

### TABLA de Códigos de Tipos de Documentos

Código	Descripción
80	CUIT
86	CUIL
87	CDI
89	LE
90	LC
91	CI Extranjera
96	DNI
94	PASAPORTE
99	OTROS Documentos

## Librería javaWSAfp - Webservice wsfeV1

### TABLA de Códigos de Tipos de Conceptos

Código	Descripción
1	PRODUCTOS
2	SERVICIOS
3	PRODUCTOS Y SERVICIOS

## Librería javaWSAfp - Webservice wsfeV1

### TABLA de Códigos de Tipos de Tributos

Código	Descripción
1	IMPUESTOS NACIONALES
2	IMPUESTOS PROVINCIALES
3	IMPUESTOS MUNICIPALES
4	IMPUESTOS INTERNOS
99	OTROS
6	PERCEPCION DEL IVA
7	PERCEPCION DE IIBB
8	PERCEPCION POR IMPUESTOS MUNICIPALES
9	OTRAS PERCEPCIONES
13	PERCEPCION DE IVA A NO CATEGORIZADO