

## 3. DESCRIPCION GENERAL

---

El servicio web de pagos de Amerika, consiste en una interfaz de funcionalidades expuesta para la gestión de pagos de externos, permitiendo al cliente del servicio registrar, anular y consultar pagos en el sistema, así como también consultar la información y el saldo de un inmueble. Algunas de las características del servicio de pagos son las siguientes:

- Para cada operación expuesta por el servicio, el cliente debe enviar en los parámetros de entrada, las credenciales de autenticación (usuario y contraseña).
- En cada respuesta generada por el servicio, se devuelven dentro de los parámetros de salida, una estructura de mensaje indicando el resultado de la operación, la cual consiste en código y mensaje de error.
- Tener credenciales válidas para el servicio web no garantiza que el cliente tenga acceso a todas las operaciones del servicio, esto es, que por políticas internas de Amerika un cliente web no podría usar una operación dada así tenga las credenciales válidas.
- Antes de poder hacer uso del servicio, el cliente debe pasar por un proceso de registro y aceptación de la IP desde donde hará las peticiones al servicio.

## 4. DESCRIPCION DETALLADA DE LAS OPERACIONES

---

El servicio web de pagos de Amerika consta de las siguientes operaciones:

- **Registro de pagos**
- **Anulación de pagos**
- **Arqueo de pagos**
- **Consulta de pagos**
- **Consulta de saldo**
- **Consulta de información básica del inmueble**

## 4.1. REGISTRO DE PAGOS

Esta operación permite registrar un pago en el sistema Amerika, ofreciendo la posibilidad de indicar el documento a pagar (factura, cuenta de cobro, etc.). Tiene las siguientes características:

Parámetros Entrada: Se recibe la estructura de datos compleja ***pagoInputType***, la cual posee los siguientes campos:

- **inmCodigo** (Requerido): código del inmueble al cual se le aplicará el pago.
- **documento** (Opcional): indica el documento comercial a pagar, sea una factura, cuenta de cobro, etc. en caso no enviar ninguno, el sistema aplicará el pago a la deuda del inmueble.
- **valor** (Requerido): valor del pago, debe ser mayor que cero (0).
- **codTrans** (Opcional): corresponde al código de transacción del banco.

Respuesta: la respuesta de esta operación consiste en la estructura de datos ***pagoRespuesta***, la cual tiene la siguiente información:

- **cabecera**: contiene información del resultado de la información
  - **codigoRespuesta**: indica si hubo o no error en la operación.
  - **mensajeRespuesta**: mensaje de error en la operación.
- **pagos**: contiene una lista de la estructura de datos ***pagoConsulta***, con la siguiente información:
  - **codigoPago**: código del pago generado por el sistema amerika.
  - **documento**: documento al que se le aplicó el pago.
  - **inmCodigo**: código del inmueble al que se le aplica el pago.
  - **fecha**: fecha del registro del pago.
  - **valor**: valor del pago.

## 4.2. ANULACIÓN DEL PAGO

Esta operación permite anular un pago que haya sido registrado por medio de este servicio web, tiene las siguientes características: Parámetros entrada:

- **codigoPago** (Requerido): este código del pago se obtiene después de haber registrado uno, viene en la información de respuesta de la operación de registrar pago.
- **motivo** (Requerido): es un motivo de anulacion existente y valido en el sistema amerika.
- **codTrans** (Opcional): corresponde al código de transacción del banco.

Respuesta:

- **cabecera**: contiene información del resultado de la información
  - **codigoRespuesta**: indica si hubo o no error en la operación.
  - **mensajeRespuesta**: mensaje de error en la operación.

## 4.3. CONSULTA DE SALDO

Esta operación permita consultar el saldo de las facturas fuera de convenio y las cuotas de convenio facturadas de un inmueble específico, tiene las siguientes características:

Parámetros entrada:

- **inmCodigo** (Requerido): código del inmueble al cual se le desea consultar el saldo.

Respuesta: la respuesta de esta operación consiste en la estructura de datos compleja **saldoRespuesta**, la cual tiene la siguiente información:

- **saldoTotal**: totalizado del saldo de las facturas fuera de convenio y cuotas de convenio facturadas.
- **baseGravableTotal**: valor del saldo total de la deuda que fue gravada, solo incluye facturas fuera de convenio.
- **baseNoGravableTotal**: valor del saldo total de la deuda que no fue gravada, solo incluye facturas fuera de convenio.
- **ivaTotal**: iva total de la deuda, solo incluye facturas fuera de convenio.

- **cuotasConvenio:** listado de cuotas de convenio facturadas representado en la estructura de datos **cuotasConsulta**, tiene la siguiente información:
  - **periodo:** periodo al que pertenece la cuota.
  - **total:** valor total de la cuota de convenio.
  - **interés:** valor de los intereses de la cuota.
  - **valorAplicado:** valor pagado a la cuota.
  - **valorAplicadoInteres:** valor pagado a intereses en la cuota.
  - **saldo:** valor pendiente por pagar de la cuota.
- **facturas:** listado de facturas adeudadas fuera de convenio, representado en la estructura de datos **facturaConsulta**, tiene la siguiente información:
  - **periodo:** periodo al que pertenece la factura.
  - **documento:** número de la factura.
  - **fechaVencimiento:** fecha de vencimiento de la factura.
  - **total:** valor total facturado.
  - **iva:** iva facturado.
  - **pagado:** valor pagado
  - **baseGravable:** valor del saldo de la factura que fue gravado.
  - **baseNoGravable:** valor del saldo de la factura que no fue gravado.
  - **saldo:** saldo de la factura.

Cabe anotar que los valores Saldo total, Base gravable total, base no gravable total e Iva total, son totalizados de los listados de cuotas y facturas.

### 4.4. ARQUEO DE PAGOS

Esta operación permite realizar un arqueo de pagos de los pagos recibidos en una rango de fecha .Tiene las siguientes características:

Parámetros entrada: se recibe un parámetro del tipo *arqueoInputType*, el cual tiene la siguiente estructura

- **fechaInicial:** Fecha inicial para realizar el arqueo o consulta.
- **fechaFinal:** Fecha final para realizar el arqueo o consulta.

**Nota:** las fechas deben tener el formato DD-MM-YYYY, por ejemplo 21-12-2015.

Respuesta: se retorna la estructura de datos *arqueoRespuesta*, con la siguiente información:

- **arqueoCabeceraRes:** contiene información del resultado de la información
  - **codigoRespuesta:** indica si hubo o no error en la operación.
  - **mensajeRespuesta:** mensaje de error en la operación.
- **valorAplicado:** valor aplicado en pagos para una fecha dada.
- **valorAnulado:** valor anulado en pagos para una fecha dada

### 4.5. CONSULTA DE PAGOS

Esta operación permite consultar los pagos realizados en una fecha dada. Tiene las siguientes características:

Parámetros Entrada: se recibe un parámetro del tipo *arqueoInputType*, el cual tiene la siguiente estructura

- **fechaInicial:** Fecha inicial para realizar la consulta.
- **fechaFinal:** Fecha final para realizar la consulta.

**Nota:** las fechas deben tener el formato DD-MM-YYYY, por ejemplo 21-12-2015

Respuesta: la respuesta de esta operación consiste en la estructura de datos ***pagoRespuesta***, la cual tiene la siguiente información:

- ***cabecera***: contiene información del resultado de la información
  - **codigoRespuesta**: indica si hubo o no error en la operación.
  - **mensajeRespuesta**: mensaje de error en la operación.
- ***pagos***: contiene una lista de la estructura de datos ***pagoConsulta***, con la siguiente información:
  - **codigoPago**: código del pago generado por el sistema amerika.
  - **documento**: documento al que se le aplicó el pago.
  - **inmCodigo**: código del inmueble al que se le aplica el pago.
  - **fecha**: fecha del registro del pago.
  - **valor**: valor del pago.

#### 4.6. CONSULTA DE INFORMACIÓN BÁSICA DEL INMUEBLE

Esta operación permite consultar la información básica del inmueble, tiene las siguientes características:

Parámetros Entrada:

- ***inmCodigo***: código del inmueble a consultar.

Respuesta: el resultado de esta operación es la estructura de datos ***inmuebleRespuesta***, que contiene la siguiente información:

- ***cabecera***: contiene información del resultado de la información
  - **codigoRespuesta**: indica si hubo o no error en la operación.
  - **mensajeRespuesta**: mensaje de error en la operación.
- ***codCliente***: código del cliente asociado al inmueble.
- ***inmCodigo***: código del inmueble.
- ***nombreCliente***: nombre del cliente.
- ***tipoidentificacion***: tipo de identificación del cliente.
- ***docidentificacion***: número de documento de identificación del cliente.
- ***telefono***: teléfono del cliente.
- ***correo***: correo del cliente.
- ***direccion***: dirección del cliente.

## 5. CÓDIGOS Y MENSAJES DE RESPUESTA DEL SERVICIO WEB

Los códigos y mensajes de respuesta que maneja el servicio son los siguientes:

Código	Mensaje	Observación
0	OK	Operación exitosa
De 100 a 200	Mensaje del servicio	Problemas y validaciones de negocio, el servicio responderá según sea el caso
De 201 a 999	Problemas Técnicos	Problemas técnicos del servicios

## 6. WSDL DEL SERVICIO WEB

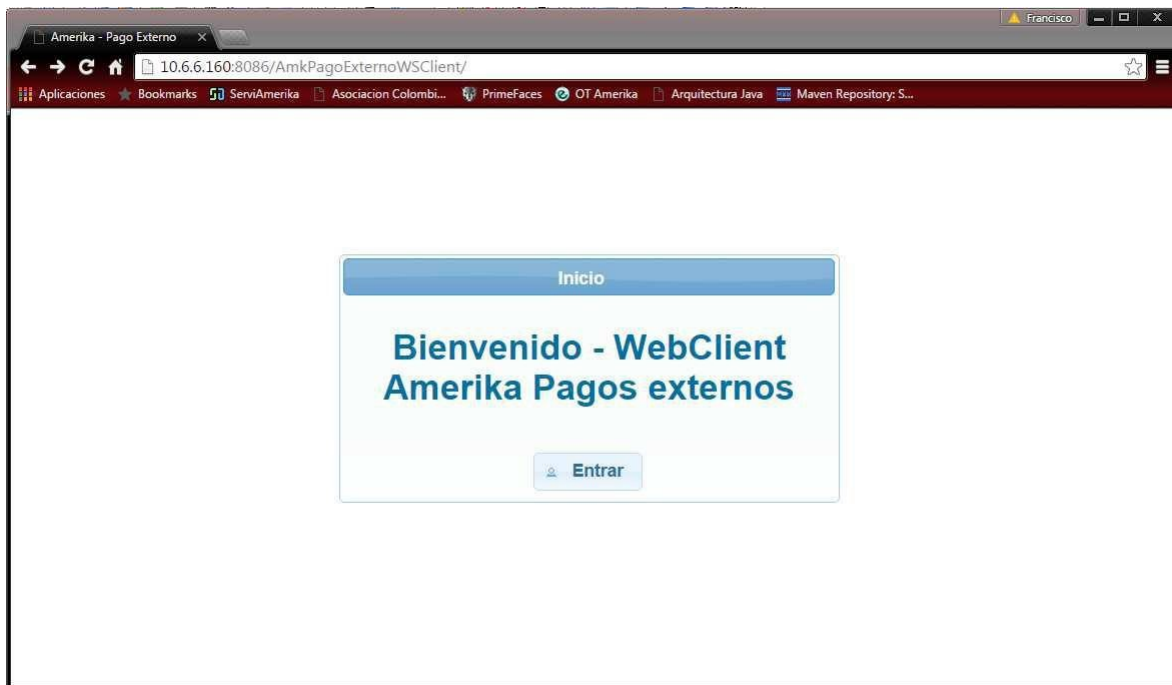
La dirección del WSDL del servicio es:

<https://wsproxy.amerika-e.com:12815/AmkWSPPagoExterno/AmkPagoExternoWSService?WSDL>

## 7. EJEMPLO DE CONSUMO DE SERVICIO WEB

Este ejemplo como se menciona al inicio de este documento, fue desarrollado en Java, cabe anotar que para la creación de un cliente web de servicio, debe descubrirse el servicio web haciendo uso del WSDL otorgado por el proveedor de servicio web, no está dentro del alcance de este ejemplo el descubrir el servicio web, por lo tanto en el ejemplo partimos del punto donde ya se ha descubierto el servicio web.

Pantalla de bienvenida del cliente web de pruebas.





## 7.1. Registrar Pago.

Consultamos el saldo antes de realizar el pago

The top screenshot displays the 'Facturas' section of the Amerika web application. It shows a table of invoices with columns for Periodo, Fec. lect., Lectura, Cons.Cons., Obs., Factura, Fecha fact., Fec. vcto., Total, and Importe pag. The table lists several invoices from 201501 to 201510. Below the table, there are sections for 'Estado de riesgo', 'Estados de cuentas', and 'Est. conceptos'. A summary table at the bottom shows 'Prom. mens de cons.' with values for 'No. per.', 'Promedio', 'Pendiente ant.', 'Saldo a favor pend.', 'Saldo convenios', 'Saldo dif. facturable', 'Total', and 'Saldo facturas'.

The bottom screenshot shows the 'Registrar Pago' form. It has a 'Login' section with 'Usuario: \*' (bbva) and 'Password: \*' (Amerika2015\*). Below this, there is a 'Registrar Pago' section with fields for 'Poliza: \*' (11646), 'Documento' (empty), and 'Valor: \*' (22.55). A 'Registrar Pago' button is at the bottom.

En la imagen anterior podemos observar como indicamos las credenciales, Código del inmueble (Póliza), Documento y el valor, luego de presionar el registrar pago, obtenemos la siguiente respuesta:

# Manual de integración con servicio web de pagos de Amerika

The screenshot shows a web browser window with the URL `10.6.6.160:8086/AmkPagoExternoWSCClient/faces/PagoExterno.xhtml`. The page has a sidebar with 'Operaciones' (Consultar Saldo, Registrar Pago, Anular Pago) and a main area with a 'Login' form and a 'Registrar Pago' section. The 'Registrar Pago' section shows a 'Mensaje Servidor: OK' and 'Codigo pago: 27'.

Obteniendo una respuesta satisfactoria (OK) por parte del servicio.

The screenshot shows a detailed view of a payment record. The table below represents the 'Facturas' section of the interface.

Periodo	Fec. lect.	Lectura	Cons.Cons.	Obs.	Factura	Fecha fact.	Fec. vcto.	Total	Importe pag.	Fecha pago	Días
201501	30-12-2014	15310	586 5172 D		3001748	07-01-2015	19-01-2015	22.55	.00		
201502	30-01-2015	21312	646 6002 BL		3037635	06-02-2015	14-02-2015	22.55	.00		
201503	03-03-2015	26895	699 5583 D		3073942	07-03-2015	17-03-2015	22.55	.00		
201504	02-04-2015	32049	720 5154 D		3110661	08-04-2015	16-04-2015	22.55	.00		
201505	04-05-2015	37666	986 5526 D		3147652	08-05-2015	18-05-2015	22.55	.00		
201506	03-06-2015	42911	843 5245 D		3184950	08-06-2015	16-06-2015	22.55	.00		
201507	03-07-2015	49043	625 5447 D		3222596	08-07-2015	16-07-2015	22.55	.00		
201508	03-08-2015	53450	603 5407 D		3260587	07-08-2015	17-08-2015	22.55	.00		
201509	02-09-2015	58773	603 5323 D		3298955	07-09-2015	15-09-2015	22.55	.00		
201510	02-10-2015	64216	581 5443 D		3337704	07-10-2015	15-10-2015	22.55	22.55	22-12-2015	76

Below the table, there are sections for 'Estado de riesgo', 'Estados de cuentas', 'Est. conceptos', 'Prom. mens. de cons', 'Detalle', and 'Pagos'.

Examinando el código fuente podemos observar los siguientes pasos:

```

public class ControllerAmkPagoExterno {

    ResourceBundle myResources = ResourceBundle.getBundle("config", new Locale("es", "CO"));
    AmkPagoExternoWSService service;
    AmkPagoExternoWS port;

    public void iniciar() throws MalformedURLException {
        service = new AmkPagoExternoWSService(new URL(myResources.getString("wsdl")));
        port = service.getAmkPagoExternoWSPort();
    }
}

```

Se declaran 2 objetos *AmkPagoExternoWSService* ( variable *service* ) y *AmkPagoExternoWS* (variable *port*), los cuales son necesarios para establecer la comunicación con el servicio web, luego se le indica a la variable *service* la url del servicio(en amarillo), esta parte es importante ya que si colocamos mal la url, no podremos conectarnos con el servicio web, por último se obtiene el objeto *port* de la variable *service*.

```

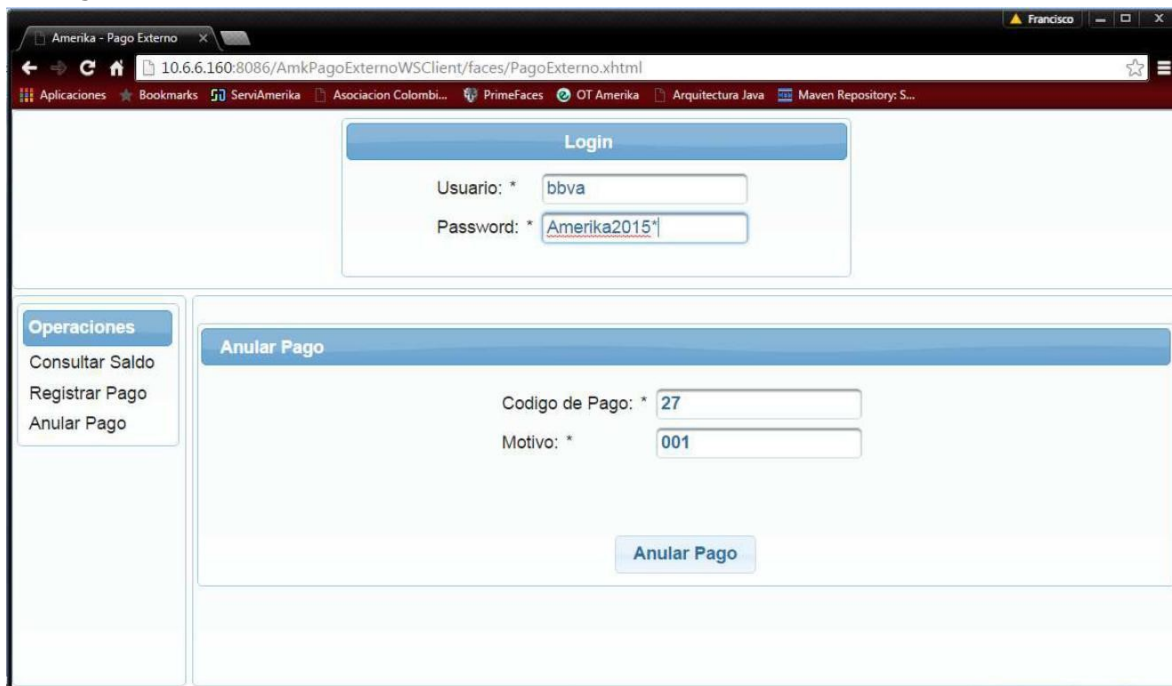
PagoInputType entradaPago = new PagoInputType();
entradaPago.setDocumento(documento);
entradaPago.setInmCodigo(inmCodigo);
entradaPago.setValor(valor);
PagoRespuesta response = port.registrarPago(usuario, password, entradaPago);
if (response.getPagos().isEmpty()) {
    respuesta.setMsgpago(response.getCabecera().getMensajeRespuesta());
} else {
    respuesta.setMsgpago(response.getCabecera().getMensajeRespuesta());
    respuesta.setLbelpagado("'" + response.getPagos().get(0).getCodigoPago());
}

```

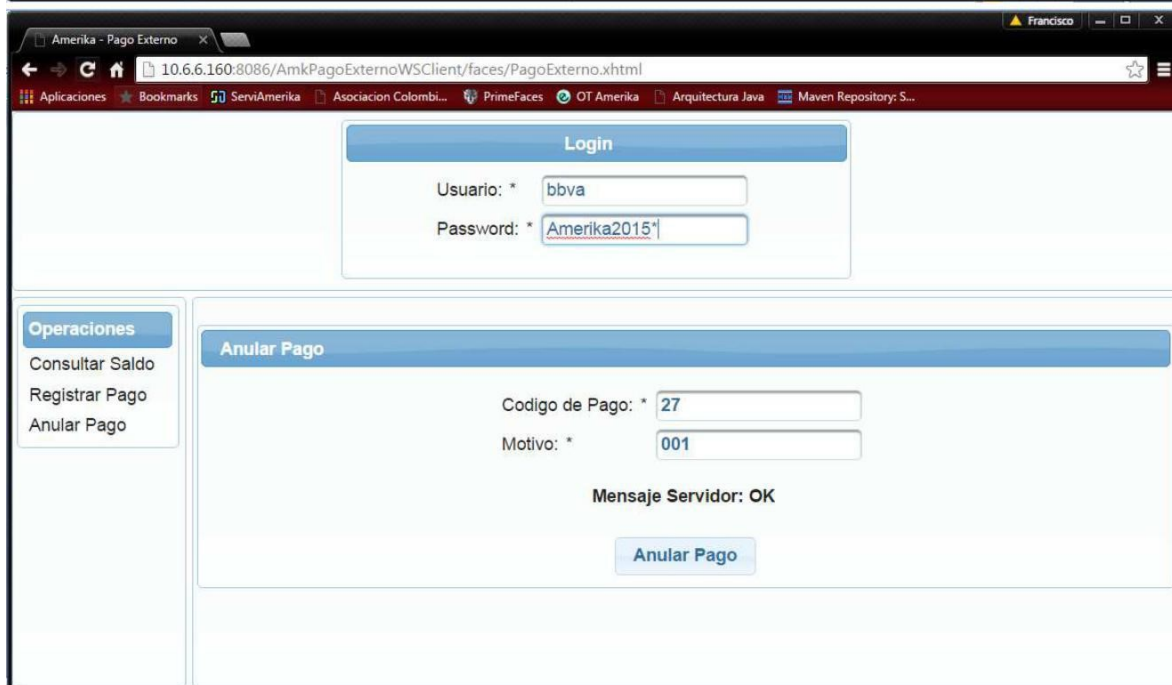
En esta imagen se hace uso de las estructuras de datos especificadas en el servicio web de pagos para la operación del registro de pagos, observamos como tenemos un objeto de tipo *PagoInputType*(variable *entradaPago*) y luego de asignarle a cada atributo del objeto los valores definidos, se envía como parámetro de entrada al llamar a la operación *registrarPago* del objeto *port*, la respuesta de la operación es almacenada en la variable *response*, que posteriormente usaremos para examinar si en la respuesta vienen los datos del pago que se registró, en caso contrario se toman los datos de la cabecera de la respuesta para informar al usuario sobre el estado de la operación.

## 7.2. Anulación del pago

Procederemos a anular el pago 27, que fue ingresado por la operación de registro de pagos.



The screenshot shows a web browser window with the URL `10.6.6.160:8086/AmkPagoExternoWSCClient/faces/PagoExterno.xhtml`. The page has a sidebar menu on the left with the following items: **Operaciones**, Consultar Saldo, Registrar Pago, and Anular Pago. The main content area is titled **Anular Pago** and contains two input fields: **Codigo de Pago: \*** with the value `27` and **Motivo: \*** with the value `001`. Below these fields is a button labeled **Anular Pago**. At the top of the page, there is a **Login** section with fields for **Usuario: \*** (value `bbva`) and **Password: \*** (value `Amerika2015*`).



This screenshot shows the same web form as above, but with an additional message: **Mensaje Servidor: OK** displayed below the input fields. The **Anular Pago** button remains visible at the bottom of the form.

Obteniendo una respuesta satisfactoria (OK) por parte del servicio.

Examinando el código fuente podemos observar los siguientes pasos:

```
PagoCabeceraRes response = port.anularPago(usuario, password, codPago, motivo);
respuesta.setLblanulado("Mensaje Servidor: " + response.getMensajeRespuesta());
```

Obtenemos la respuesta del servicio en la variable response, de la cual obtenemos código y mensaje de respuesta, para este ejemplo el resultado es satisfactorio (OK).

## 7.3. Consulta de Saldo

The screenshot displays the 'Amerika - Pago Externo' web application. The browser address bar shows the URL: 10.6.6.160:8086/AmkPagoExternoWSSClient/faces/PagoExterno.xhtml. The page features a login section at the top with fields for 'Usuario' (filled with 'bbva') and 'Password' (filled with 'Amerika2015\*'). Below the login section is a sidebar titled 'Operaciones' with links for 'Consultar Saldo', 'Registrar Pago', and 'Anular Pago'. The main content area is titled 'Consultar Saldo' and contains a 'Poliza' field with the value '34915' and a 'Consultar Saldo' button. Below this are two tables: 'Cuotas Convenio' and 'Facturas'. Both tables have headers with columns for 'Periodo', 'Total', 'Interes', 'Valor Apl', 'Valor Apl-Int', 'Saldo' (for Cuotas) and 'Periodo', 'Documento', 'Fecha Venc.', 'Total', 'Iva', 'Pagado', 'Base Grv', 'Base no Grv', 'Saldo' (for Facturas). Both tables currently display 'No records found.'

Luego de consultar el saldo, obtendremos el siguiente resultado



# Manual de integración con servicio web de pagos de Amerika

**Login**

Usuario: \*

Password: \*

**Operaciones**

- Consultar Saldo
- Registrar Pago
- Anular Pago

**Consultar Saldo**

Poliza: \*

**Datos**

Base Gravable:	0,09
Base no Gravable:	19,76
Saldo Total:	214,09

**Cuotas Convenio**

Periodo	Total	Interes	Valor Apl	Valor Apl-Int	Saldo
201510	48,58	1,48	0	0	48,58
201511	48,58	1,12	0	0	48,58
201512	48,58	0,75	0	0	48,58
201601	48,49	0,36	0	0	48,49

**Facturas**

Periodo	Documento	Fecha Venc.	Total	Iva	Pagado	Base Grv	Base no Grv	Saldo
201510	3331692	14-10-2015	19,86	0,01	0	0,09	19,76	19,86

Podemos observar en las imagenes el resultado de la operacion consultar saldo, con diversa informacion de facturas y cuotas de convenio.

Examinando el código fuente podemos observar los siguientes pasos:

```
SaldoRespuesta response = port.consultarSaldo(usuario, password, inmCodigo);  
if (!response.getCuotasConvenio().isEmpty() || !response.getFacturas().isEmpty()) {  
    respuesta.setBasegrab(response.getBaseGrabableTotal());  
    respuesta.setBasenograb(response.getBaseNoGrabableTotal());  
    respuesta.setSaldot(response.getSaldoTotal());  
    respuesta.setCuotas(response.getCuotasConvenio());  
    respuesta.setFacturas(response.getFacturas());  
} else {  
    respuesta.setLblmsgconsulta(response.getCabeceraRespuesta().getMensajeRespuesta());  
}
```

Podemos observar que a la variable *response* se le asigna el resultado de la operación *consultarSaldo* de la variable *port*, para este ejemplo se valida que el listado de facturas y de cuotas de convenios no estén vacíos, luego se toman los totales (saldo, bases e iva), en caso de problemas en el servicio, se toman el código y mensaje de error del servicio.