Interfaces Externas con Bancos

Índice

1. OBJETIVO
2. DESARROLLOS EN LOS BANCOS
2.1 Archivo conciliatorio 2
3. DESCRIPCIÓN DE LAS INTEGRACIONES
4. ARQUITECTURA DE LA INTEGRACIÓN
5. ESPECIFICACIÓN TÉCNICA INTERFACES
5.1 Generación de token privado 5
5.2 Variables 5
5.3 Petición de token 5
5.4 Servicio de obtención de tokens 7
6. SERVICIOS
6.1 Llamada a los servicios 8
6.2 Servicio de Recuperación de Deuda 8
6.3 Servicio de Confirmación de Pago 9
6.4 Servicio de Anulación de Pago 10
7. CONSIDERACIONES GENERALES DE CONECTIVIDAD

1. OBJETIVO

Este documento describe la integración entre Onesait Customers (OC) y los sistemas de los Bancos u otras entidades que actúen como canales de pago para el cliente. Conteniendo la especificación técnica detallada de los conectores a utilizar para la integración.

2. DESARROLLOS EN LOS BANCOS

Cada Banco deberá implementar la siguiente funcionalidad:

a) **Formulario/aplicación** (externo o incluido en aplicación actual del Banco si la tiene) que le permita escanear el código de barras del Documento de Cobro emitido (o introducir la referencia de pago del cliente manualmente).
b) Una vez introducida la información, se realizará la **Invocación automática** a Web Service API Rest de OC con la referencia de pago *(Servicio de recuperación de Deuda).*
c) Mostrar **información recuperada** del Web Service en el formulario, mínimo mostrar:
   - ✓ Nombre cliente (customerName)
   - ✓ Deuda de la cuenta del Cliente (accountBalance)

   Opcionalmente se puede mostrar también:
   - ✓ Importe última Factura (lastBillAmount)
   - ✓ Fecha vencimiento última Factura (lastBillDueDate)

d) Cuando el cliente entrega el importe que corresponda (puede ser pago total o parcial), **se Confirmará el pago** desde el Formulario.
e) Al confirmar el pago, se realizará la **Invocación automática** a Web Service API Rest de OC con la referencia, fecha, importe e identificación del Banco para confirmar el pago (*Servicio de confirmación de Pago).*
f) Si el sistema OC devuelve un fallo, no se podrá aceptar el pago en el Banco.
g) **Diariamente** los bancos deben enviar un archivo **conciliatorio** con detalle de todos los pagos realizados en el día.

2.1 Archivo conciliatorio

El formato de este archivo contempla un registro de cabecera con los totales a enviar y los registros del detalle de los pagos realizados.

A continuación, indicamos la estructura del archivo.

Cabecera:

Cabecera (174 caracteres)				
Posición	**Campo**	**Tipo**	**Long**	**Descripción**
1 – 1	Tipo de Registro	AN	1	Identifica el registro, “C” para la cabecera
2 – 9	Fecha de Cobro	N	8	Fecha de Cobro (YYYYMMDD) Es la fecha del día que se realizaron los pagos (por parte del cliente) que se informan. Debe ser un archivo conciliatorio por día con los pagos del día.
10 – 13	Código del Recaudador	N	4	Id. del Recaudador, asignado por SEDAPAL.
14 – 21	Número de recibos	N	8	Número de recibos cobrados
22 - 36	Importe total	N	15	Importe total cobrado. Los dos últimos dígitos se asumen como los decimales.
37 - 44	Fecha generación archivo	N	8	Fecha de generación del archivo (YYYYMMDD)
45 - 50	Hora generación archivo	N	6	Hora de generación del archivo (HHMMSS)
51 - 174	Uso futuro	AN	124	Para uso futuro. Enviar ceros

Detalle:

Detalle (174 caracteres)					
Pos	**Campo**	**Tipo**	**Obligatorio**	**Long**	**Descripción**
1 – 1	Tipo de Registro	AN	SI	1	Identifica el registro, "D" para los detalles
2 – 3	Tipo Terminal	N	SI	2	Tipo de terminal que origina la transacción ver posibles valores en la siguiente tabla: **Código** / **Descripción** 2 ATM 4 BOX – Electronic Cash Register 7 IVR – Interactive Voice Response 14 POS 15 WEB 18 Terminal Administrativo 51 MasterCard 52 NET 54 Kiosko 55 HEPS 56 Banco 90 Ventanilla
4 – 11	Identificación Terminal	AN	SI	8	Código que identifica al terminal que origina la transacción, puede ser su número de serie
12 – 19	Fecha Transacción	N	SI	8	PaymentDate (parte de fecha) - Fecha de la transacción (YYYYMMDD), del pago.
20 – 25	Hora Transacción	N	SI	6	PaymentDate (parte de hora) - Hora de la transacción (HHMMSS)
26 – 31	Número Trace	N	SI	6	transactionId - Número secuencial de la transacción
32 – 46	Número de cuenta	N	NO	15	accountReference Se rellena con ceros a la izquierda.
47 – 61	Número Documento	N	SI	15	collectionReference - Número que identifica al documento. Referencia de cobro de 10 dígitos. Se debe rellenar con ceros a la izquierda. Es el "collectionReference", se obtiene del documento (boleta/factura) del cliente que está pagando. Se envía en el ws "externalPayment".
62 – 73	Importe Documento	N	SI	12	amount - Monto total pagado. Las 2 últimas posiciones se asumen como decimales
74 – 76	Código Moneda	N	SI	3	currency - Código ISO de la moneda. En este caso es el valor fijo PEN
77 – 84	Fecha Emisión	N	NO	8	Fecha de emisión del documento, formato YYYYMMDD.
85 – 92	Fecha Vencimiento	N	NO	8	Fecha de vencimiento del documento, formato YYYYMMDD.
93 – 132	Nombre Cliente	AN	NO	40	Nombre del Cliente. Rellenar con espacios a la derecha.
133 –136	Código de Agencia	N	SI	4	paymentCenter - Código de Agencia donde se ejecutó la transacción.

					Es un código especifico de cada banco que se les informará (es el paymentCenter que se informa en el ws "externalPayment").
137– 174	Uso Futuro	AN	NO	38	Para uso futuro, debe ir con ceros

Consideraciones:

a) El formato del nombre del archivo será el siguiente: **COL9999YYYYMMDD.txt**, donde:
   - COL: Acrónimo de cobranza online.
   - 9999: Código de Adquiriente asignado por Sedapal (centro de cobro)
   - YYYYMMDD: Fecha de cobro.

b) En el archivo NO se reciben anulaciones. Las diferencias entre los pagos recibidos a través de los web services y el archivo conciliatorio, se visualizarán en el sistema OC según las 2 situaciones siguientes:
   - los cobros que han llegado por web service durante el día pero no en el archivo de conciliación del final del día
   - los cobros que han llegado en el fichero de confirmación, pero no los recibimos por web service durante el día.

   Sedapal deberá revisar estas diferencias, anulando si no correspondían o gestionándolos como erróneos, según el caso.

Ejemplo

Adjuntamos archivo de muestra del archivo conciliatorio.

COL395420230725.txt

3. DESCRIPCIÓN DE LAS INTEGRACIONES

	PETICIÓN SISTEMA BANCO	RESPUESTA DEL OC
Servicio de Recuperación de Deuda	Este método será utilizado por los bancos para **recuperar la deuda** de una Cuenta. **Datos** principales del webservice: □ código identificador del Banco □ referencia de pago	Devolverá datos del cliente titular de la Cuenta y su deuda. O error en caso de no pasar validaciones (ej: referencia de pago inválida).
Servicio de Confirmación de Pago	Este método será utilizado por los bancos para **confirmar el pago** realizado para una Cuenta **Datos** principales del webservice: □ fecha real de pago □ código identificador del Banco □ referencia de pago □ importe pagado, □ identificador del pago en el sistema origen	Devolverá OK en caso de aplicación de pago satisfactoria, y también devolverá el saldo de deuda para que pueda imprimirse en el justificante de pago. O error en caso de no pasar validaciones (ej: referencia de pago inválida).
Servicio de Anulación de Pago	Este método será utilizado por los bancos para **la anulación del pago** realizado para una Cuenta **Datos** principales del webservice: □ fecha real de pago □ código identificador del Banco □ referencia de pago	Devolverá OK en caso de aplicación de anulación de pago satisfactoria, y también devolverá el saldo de deuda. O error en caso de no pasar validaciones (ej: referencia de pago inválida).

	□ importe pagado, □ identificador del pago en el sistema origen □ código del cobro a anular □ mensaje	

4. ARQUITECTURA DE LA INTEGRACIÓN

a) **Accesibilidad** mediante url pública para los Bancos colaboradores
b) **Servicios de negocio** API Rest expuestos en API Manager (WSO2), componente de la plataforma de integración incluida
c) **Securización** del proceso mediante combinación de técnicas proporcionado por plataforma OC:
   - **Cifrado** de comunicaciones mediante HTTPS y certificado de CA válido (a proporcionar por la empresa Cliente de Onesait Customers)
   - Uso de **token privado** otorgado a cada banco
   - Aplicación de algoritmos de aseguramiento de tokens
   - Monitorización de peticiones realizada por cada banco
   - **Procesamiento Online** de las peticiones recibidas desde los Bancos:

5. ESPECIFICACIÓN TÉCNICA INTERFACES

5.1 Generación de token privado

Antes de cada llamada a un api de Onesait Customers es obligatorio llevar a cabo la generación de un token previo que se incluirá en cada petición para asegurar la procedencia de la petición y que supere los controles de seguridad realizados en la plataforma de integración.

5.2 Variables

La empresa cliente de Onesait Customers proveerá los **valores** de las siguientes variables que serán exclusivos para cada banco:

```
String user = "Bank01";
String pass = "Bank01passw";
String scope = "apim:subscribe";
String consumerKey = "XlogRfS3Nt5oI0TV5_TxCGsolz0a";
String consumerSecret = "GIASch2fTmIxHQKGbYuBfAmIAloa";
String TOKEN_URL = "http://172.16.10.84:8281/api/token";
```

Nota: los valores aquí indicados son ejemplos incluidos meramente a efectos ilustrativos. Los valores finales serán informados en el apartado "**7 Consideraciones Generales de Conectividad**", cuando estén disponibles.

5.3 Petición de token

En el sistema del banco que vaya a integrarse con Onesait Customers debe incluirse la **generación de Token** mediante una petición a la plataforma de integración. Se proporciona código java con el detalle de la misma:

```
URL tokenEndpointURL;
JSONObject accessTokenRequest, tokenData = null;
Map<String, String> authenticationRequestHeaders = new HashMap<String, String>();
String basicAuthHeader = consumerKey + ":" + consumerSecret;
String encodedBasicAuthHeader = DatatypeConverter.printBase64Binary(basicAuthHeader.getBytes("UTF-8"));
authenticationRequestHeaders.put("Authorization", "Basic " + encodedBasicAuthHeader);
tokenEndpointURL = new URL(TOKEN_URL);
```

```
String requestGrantType = "grant_type=password&username=" + user + "&password=" + pass + "&scope=" + scope;
accessTokenRequest = new JSONObject(HttpRequestUtil.doPost(tokenEndpointURL,requestGrantType, authenticationRequestHeaders));
String dataNodeInformation = accessTokenRequest.getString("data");
tokenData = new JSONObject(dataNodeInformation);
String accessTOKEN = tokenData.getString("access_token");
System.out.println("ACCESS_TOKEN : " + accessTOKEN);
```

5.4 Servicio de obtención de tokens

<table>
<tr><th></th><th colspan="2">Detalle</th></tr>
<tr><td>Tipo</td><td colspan="2">POST</td></tr>
<tr><td>Servicio</td><td colspan="2">token</td></tr>
<tr><td>Entrada</td><td colspan="2">La llamada al servicio debe proporcional la siguiente información:
<table>
<tr><th>Campo</th><th>Formato</th><th>Descripción</th></tr>
<tr><td>User</td><td>string</td><td>Usuario, su valor será proporcionado, ejemplo: "Bank01".</td></tr>
<tr><td>pass</td><td>String</td><td>Clave, su valor será proporcionado, ejemplo: "Bank01passw".</td></tr>
<tr><td>scope</td><td>String</td><td>Alcance, su valor será proporcionado, ejemplo: "apim:subscribe".</td></tr>
<tr><td>consumerKey</td><td>String</td><td>Clave banco, su valor será proporcionado, ejemplo: "XlogRfS3Nt5ol0TV5_TxCGsolz0a".</td></tr>
<tr><td>consumerSecret</td><td>String</td><td>Clave secreta, su valor será proporcionado, ejemplo: "GIASch2fTmIxHQKGbYuBfAmlAloa".</td></tr>
</table></td></tr>
<tr><td>URL</td><td colspan="2">"http://"+ {TOKEN_URL} + "/api/token" + "grant_type=password&username=" + {user} + "&password=" + {pass} + "&scope=" + {scope} + " Authorization: Basic " + base64({consumerKey} + “:” + {consumerSecret})</td></tr>
<tr><td>JSON (result)</td><td>{
"access_token",
"refresh_token",
"scope",
"token_type",
"expires_in"
}</td><td>
- token recibido
- Token alternativo para generar otro token
- scope recibido
- tipo recibido
- tiempo expiración (milisegundos)</td></tr>
</table>

Ejemplo:

Llamada (usando curl):

```
Request POST 'https://ocapiext-hom.nsc.sedapal.com.pe/api/token' \
--header 'Authorization: Basic WGxvZ1JmUzNOdDVvSTBUVjVfVHhDR3NvbHowYTpHSUFTY2gyZlRtSXhIUUtHYll1QmZBbWxBbG9h'\
--data-urlencode 'grant_type=password' \
--data-urlencode 'username=Bank01' \
--data-urlencode 'password=Bank01passw' \
--data-urlencode 'scope=apim:subscribe'
```

Respuesta:

```
Result:
   {
      "access_token": "82398e9e-3361-3a0f-a6cd-dd9e164d9818",
      "refresh_token": "bd70c2a4-328e-3ebf-b37e-6cd91816c040",
      "scope": "apim:subscribe",
      "token_type": "Bearer",
      "expires_in": 2886
   }
```

6. SERVICIOS

6.1 Llamada a los servicios

Tras la obtención de un token, se debe incluir una **cabecera** http en las peticiones a los servicios:

- Nombre: "Authorization"
- Valor: "Bearer " + ${token};

También se incluirá en la cabecera el identificador del centro de cobro, que será el mismo valor que el usuario que utilizarán para obtener el token:

- Nombre: "X-Incms-Origin-D"
- Valor: ${username} (correspondería al valor "Bank01" en el ejemplo planteado en el punto anterior del documento)

6.2 Servicio de Recuperación de Deuda

<table>
<tr><th></th><th colspan="3">Detalle</th></tr>
<tr><td>Tipo</td><td colspan="3">GET</td></tr>
<tr><td>Servicio</td><td colspan="3">/{accountReference}/balance</td></tr>
<tr><td>Entrada</td><td colspan="3">la Referencia de la Cuenta (accountReference – String (15)) se envía en la propia URL
Considerando que el "accountReference":
• Para el caso de facturas del sistema anterior (OpenSGC) será un código de 7 dígitos (el actual nro. de suministro)
• Para el caso de facturas del nuevo sistema (OC) será un código de 9 dígitos (que corresponde con lo que en la nueva factura se indica como "Referencia de cuenta").</td></tr>
<tr><td>Cabecera</td><td colspan="3">Authorization Bearer ${access_token}
X-Incms-Origin-D ${username}</td></tr>
<tr><td>JSON salida</td><td colspan="2">{
"data": {
"customerName": "string (40)",
"accountReference": "string (15)",
"accountBalance": "decimal (18,2)",
"lastBillAmount": "decimal (18,2)",
"lastBillDueDate": "long",

"currencyId": "int (3)",
"currencyIsoCode": "string (3)"
}
}</td><td>- Nombre del Cliente
- Referencia de Cobro (número de cuenta)
- Saldo de deuda de la Cuenta
- Importe de la última factura
- Fecha de vencimiento de la última factura en formato timestamp milisegundos
- Identificador de la divisa
- Código ISO de la divisa (Para soles es el valor fijo PEN)</td></tr>
<tr><td>Validaciones</td><td colspan="3">• Existe la Referencia de Cobro ("accountReference")</td></tr>
<tr><td>Códigos de respuesta</td><td>HTTPSTATUS
200
422</td><td>CODE
200
422</td><td>MSGUSER
No mostrado. Devuelto en el JSON de salida.
Mensaje con el detalle del error (en función de la validación).</td></tr>
</table>

Ejemplo:

Servicio: /ACC0000001380/balance

Cabecera:
Authorization Bearer 82398e9e-3361-3a0f-a6cd-dd9e164d9818
X-Incms-Origin-D Bank01

Result:

```
{
   "data": {
      "customerName": "CLIENTE0000001380 CL0000001380"
      "accountReference": "ACC0000001380",
      "accountBalance": 1652.29,
      "lastBillAmount": 1652.29,
      "lastBillDueDate": 1704841200000,
```

```
        "currencyId": 1,
        "currencyIsoCode": "PEN"
    }
}
```

6.3 Servicio de Confirmación de Pago

	Detalle	
Tipo	POST	
Servicio	/externalPayment	
Cabecera	Content-Type application/json Accept application/json Authorization Bearer ${access_token} X-Incms-Origin-D ${username}	

JSON entrada	{
	"paymentDate": "long",
	"collectionReference": "string (20)",
	"amount": “decimal(18,2)”,
	"currency": "string (3)",
	"transactionId": "string (36)",
	
	"paymentCenter": "string (10)"
	}
		- Fecha de pago en formato timestamp milisegundos
		- Referencia del documento de cobro
		- Importe pagado por el Cliente
		- Divisa (Valor fijo - PEN)
		- Código identificador de la transacción en el sistema origen (Banco)
		- ${username} Código que identifica dónde se ha registrado el pago (se definirá un código distinto por Banco y Cuenta bancaria)

JSON salida	{
	"data": {
	"collectionId": “int (15)”,
	
	"confirmationMessage": "string (100)",
	"collectionStatus": "string (20)",
	"accountBalance": “decimal(18,2)”
	}
	}
		- Identificador del registro de la operación de cobro en OC. Al recibir este dato confirmará que el pago se ha realizado con éxito.
		- Mensaje de OK
		- Estado del registro de la operación de cobro en OC
		- Saldo después de la operación

Validaciones	• Campos obligatorios: “paymentDate”, “collectionReference”, “amount”, “transactionId” y “currency” • Que el Centro de Cobro (“paymentCenter”) existe y esté activo • Que la Referencia de Cobro (“collectionReference”) existe. • No se duplique el “transactionId” para un mismo Centro de Cobro	

Códigos de respuesta	HTTPSTATUS	CODE	MSGUSER
	200 422	200 422	No mostrado. Devuelto en el JSON de salida. Mensaje con el detalle del error (en función de la validación).

Ejemplo:

Servicio: /externalPayment

Cabecera:

Authorization Bearer 82398e9e-3361-3a0f-a6cd-dd9e164d9818
X-Incms-Origin-D Bank01
Content-Type: application/json
Accept: application/json

Request:

```
{
    "paymentDate": 1707492086000,
    "collectionReference": "2023ROOTDC0000000733",
    "amount": 1652.29,
    "currency": "PEN",
    "transactionId": "1",
    "paymentCenter": " Bank01"
```

}
Result – OK: Code 200
{
"data": {
"collectionStatus": "002ESTCOLL"
"accountBalance": 6324485,
"collectionId": 914294,
"confirmationMessage": "10 : InCMS-BL-CB001532. El cobro se ha realizado correctamente"
}
}
Result – KO: Code 422
{
"httpStatus": 422,
"code": "CB001522",
"helpLink": "CB001522",
"errorSequence": "2742ff8:18d40f6030c:-7fc4",
"msgUser": "InCMS-BL-CB001522. No existe el código de la agencia contenido en el cobro"
}

6.4 Servicio de Anulación de Pago

	Detalle		
Tipo	POST		
Servicio	/sedCancelExternalPayment		
Cabecera	Content-Type application/json Accept application/json Authorization Bearer ${access_token} X-Incms-Origin-D ${username}		

JSON entrada	{
	"paymentDate": "long",
	"collectionReference": "string (20)",
	"amount": "decimal(18,2)",
	"currency": "string (3)",
	"transactionId": "string (36)",
	"paymentCenter": "string (10)",
	"collectionId": "int (3)",
	"cancelMessage": "string (100)"
	}
			- Fecha de pago en formato timestamp milisegundos
			- Referencia de cobro del documento de cobro
			- Importe pagado por el Cliente
			- Divisa (Valor fijo - PEN)
			- Código identificador de la transacción en el sistema origen (Banco)
			- ${username} Código que identifica dónde se ha registrado el pago (se definirá un código distinto por Banco y Cuenta bancaria)
			- Código del cobro a anular
			- Mensaje de Motivo de cancelación

JSON salida	{
	"data": {
	"collectionId": "int (15)",
	"confirmationMessage": "string (100)",
	"collectionStatus": "string (20)",
	"accountBalance": "decimal (18,2)"
	}
	}
			- Identificador del registro de la operación de cobro en OC que ha sido anulada. Esta acción no requiere confirmación de Sedapal
			- Mensaje de OK
			- Estado del registro de la operación de cobro en OC
			- Saldo después de la operación

Validaciones	• Campos obligatorios: todos. • Que los datos recibidos (paymentDate, collectionReference, amount, currency, transactionId, paymentCenter), sean exactamente igual a los datos del cobro a anular (collectionId). • Que el cobro no esté anulado.		
Códigos de respuesta	**HTTPSTATUS**	**CODE**	**MSGUSER**
	200 422	200 422	No mostrado. Devuelto en el JSON de salida. Mensaje con el detalle del error (en función de la validación).

Ejemplo:

Servicio: /sedCancelExternalPayment

Cabecera:
Authorization Bearer 82398e9e-3361-3a0f-a6cd-dd9e164d9818
X-Incms-Origin-D Bank01
Content-Type: application/json
Accept: application/json
Request:

{
"paymentDate": 1707492086000,
"collectionReference": "2023ROOTDC0000000733",
"amount": 1652.29,
"currency": "PEN",
"transactionId": "1",
"collectionId": 914294,
"paymentCenter": " Bank01" ,
"cancelMessage": “Solicitado por el cliente”
}

Result – OK: Code 200

{
"data": {
"collectionStatus": "002ESTCOLL"
"accountBalance": 6322832.71,
"collectionId": 914294,
"confirmationMessage": "InCMS-BL-CB001538. La cancelación del cobro se ha realizado correctamente"
}
}

Result – KO: Code 422

{
"httpStatus": 422,
"code": "CB001522",
"helpLink": "CB001522",
"errorSequence": "2742ff8:18d40f6030c:-7fc4",
"msgUser": "InCMS-BL-CB001522. No existe el código de la agencia contenido en el cobro"
}

7. CONSIDERACIONES GENERALES DE CONECTIVIDAD

Ambiente de Pruebas

A continuación, se indican las consideraciones técnicas para la ejecución de pruebas.

Los servicios web antes indicados estarán desplegados bajo las siguientes condiciones.

Ambiente: Homologación, este ambiente de Sedapal está reservado para el proyecto, se definirán aquí juegos de datos para pruebas.

URL: https://ocapiext-hom.nsc.sedapal.com.pe
IP: 10.100.195.30
Puerto: 443

Obs:
Esta dirección se utilizará para todas aquellas entidades que tengan una línea dedicada con Sedapal.
Para el caso del archivo conciliatorio se mantendrá su envío por vía email (inclusive para las pruebas).
Al momento de desplegar las aplicaciones pueden variar sus firmas, si hubiesen diferencias serán informadas oportunamente.