



Lograr la integración del pago móvil vía webview

Guía de implementación

Versión del documento 1.5

Contenido

1. HISTORIAL DEL DOCUMENTO.....	3
2. PRESENTACIÓN.....	4
3. CINEMÁTICA DE PAGO.....	5
4. INTEGRACIÓN DEL PAGO.....	6
5. FASE 1: EL SERVIDOR DEL VENDEDOR.....	7
5.1. Creación del formulario de pago.....	7
Transmitir los datos del comprador.....	10
Transmitir los datos del pedido.....	11
Transmitir los datos de entrega.....	13
5.2. Calcular la firma.....	14
5.3. Transferencia de la solicitud de pago.....	16
5.4. Recepción de la URL de notificación.....	16
5.5. Procesamiento de la notificación de fin de pago (IPN).....	16
5.6. Ejemplo de código.....	17
6. FASE 2: LA APLICACIÓN MÓVIL.....	18
6.1. Cámara para escanear tarjeta.....	18
6.2. NFC para escanear tarjeta.....	19

1. HISTORIAL DEL DOCUMENTO

Versión	Autor	Fecha	Comentario
1.5	Lyra Network	12/05/2020	<ul style="list-style-type: none">Integración en el servidor: adición de los campos del formulario para transmitir los datos del pedido, el comprador y la entrega.Adición del cálculo de la firma.Rediseño de la integración en la aplicación móvil.Adición del soporte de escaneo de la tarjeta mediante cámara y NFC.
1.4	Lyra Network	16/10/2019	Renombrar clases en ejemplos de código
1.3	Lyra Network	16/10/2018	Rediseño integral del documento.
1.2	Lyra Network	16/03/2018	Adición del capítulo ¿Cómo puede la aplicación nativa detectar el fin del pago? Actualización del capítulo Enviar la solicitud de pago desde el servidor del vendedor : adición del parámetro <i>vads_theme_config</i>
1.1	Lyra Network	04/01/2018	Adición del capítulo sobre la optimización del desempeño
1.0	Lyra Network	10/11/2017	Versión inicial

Este documento y su contenido son estrictamente confidenciales. No es contractual. Cualquier reproducción y/o distribución total o parcial de este documento o de su contenido a una entidad tercera está estrictamente prohibido o sujeta a una autorización escrita previa de Lyra Network. Todos los derechos reservados.

2. PRESENTACIÓN

PayZen Le ofrece una solución única para la integración del pago móvil a sus aplicaciones.

Nuestra solución abarca las aplicaciones nativas iOS y Android. Se basa en el uso del componente **webview**.

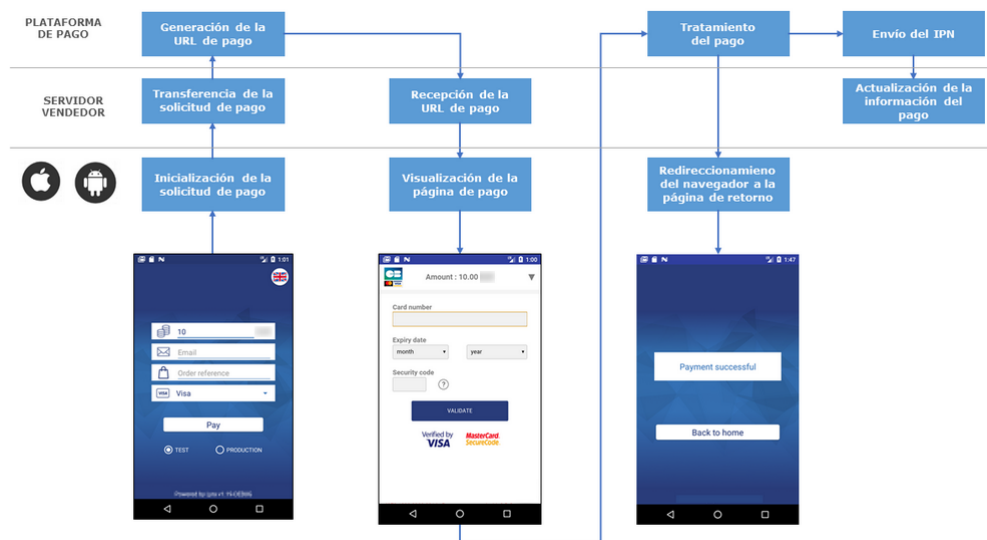
Una webview permite mostrar contenido ya disponible en la web dentro de la aplicación.

De esta forma, la solución PayZen de pago móvil vía webview ofrece varias ventajas al comerciante:

- Una configuración única para el entorno web y móvil.
Puede volver a utilizar la misma configuración de los pagos de su sitio web.
Los medios de pago activados, la reglas anti-fraude, etc., se retoman en la aplicación móvil.
- Coherencia en la visualización de la información del proceso del comprador.
Nuestras páginas de pago ofrecen capacidad de respuesta, por lo cual se pueden adaptar a los diferentes terminales de sus clientes (móvil, tableta o computadora de escritorio).
- Cuenta con un alto nivel de seguridad gracias a nuestro certificado PCI DSS y a la gestión de 3D integrada en el proceso de pago.

PCI DSS (= Payment Card Industry Data Security Standard) es la norma de seguridad de la industria de las tarjetas de pago. Es una norma de seguridad de datos para los principales grupos de tarjetas de pago como Visa, MasterCard, American Express, Discover y JCB.

3. CINEMÁTICA DE PAGO



El comprador valida su canasta.

1. La aplicación móvil realiza una solicitud de pago al servidor del comerciante.
2. El servidor del vendedor envía una solicitud de pago a la plataforma.
3. La plataforma genera una URL de pago y la transmite a su vez a la aplicación móvil.
4. El servidor del vendedor envía la URL de pago a la aplicación móvil.
5. La aplicación móvil abre la página de pago mediante WebView.
6. El comprador ingresa los datos de su tarjeta y hace clic en **Validar**.
7. La plataforma procede al pago y transmite la notificación de pago al servidor del vendedor.
8. El sitio web del vendedor analiza el resultado del pago.
9. El comprador es redirigido automáticamente a la aplicación del vendedor.

4. INTEGRACIÓN DEL PAGO

Algunos ejemplos de códigos están disponibles para facilitar la integración:

Servidor Comerciante <https://github.com/lyra/webview-payment-sparkjava-integration-sample>

iOS <https://github.com/lyra/webview-payment-ios-integration-sample>

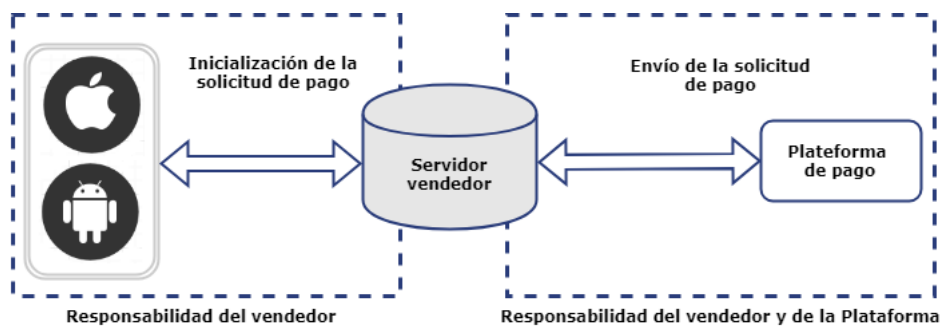
Android <https://github.com/lyra/webview-payment-android-integration-sample>

IMPORTANTE

Asegúrese de haber leído los comentarios en los archivos `Léame` antes de iniciar la aplicación. Los archivos `MainActivity.kt` y `app-configuration.properties` deben modificarse según las instrucciones descritas en los comentarios.

La integración se divide en dos etapas:

- integración de los intercambios entre el servidor del vendedor y la plataforma de pago
- integración de los intercambios entre la aplicación móvil y el servidor del vendedor



5. FASE 1: EL SERVIDOR DEL VENDEDOR



5.1. Creación del formulario de pago

El servidor del vendedor recibe una solicitud de pago de la aplicación móvil y debe transmitirla a la plataforma de pago.

Para ello, el sitio del comerciante genera un formulario de pago HTML que luego enviará a la plataforma de pago.

La integridad de los datos intercambiados está protegida por un intercambio de firmas alfanuméricas entre la plataforma de pago y el sitio web del comerciante.

El servidor del comerciante transmitirá la firma alfanumérica en el formulario de pago (consulte el capítulo Calcular la firma en la página 14).

IMPORTANTE

Todos los datos del formulario deben estar codificados en UTF-8.

De esta forma, los caracteres especiales (acentos, puntuación, etc.) serán interpretados correctamente por la plataforma de pago.

En el caso contrario, el cálculo de la firma será erróneo y el formulario será rechazado.

Para crear el formulario de pago:

1. Utilice todos los campos presentes en el cuadro para crear la solicitud de pago.

Nombre del campo	Descripción	Formato	Valor
vads_site_id	Identificador de la tienda	n8	Ejemplo: 12345678
vads_currency	Código numérico de la moneda que se utilizará para el pago, según la norma ISO 4217 (código numérico)	n3	Ejemplo: 978 para el euro (EUR)
vads_amount	Monto del pago en su unidad monetaria más pequeña (el centavo para el euro)	n..12	Ejemplo: 3000 para 30,00 euro
vads_cust_email	Dirección de correo electrónico del comprador	ans..150	Ejemplo: abc@example.com
vads_payment_cards	Tipo de tarjeta.	String	Ejemplo: VISA (Consulte la <i>Guía de implementación de la API de formularios</i> para la lista de valores posibles).
vads_order_id	Número del pedido	ans..64	Ejemplo: 2-XQ001
vads_version	Versión del protocolo de intercambio con la plataforma de pago	enum	V2
vads_theme_config	Permite aumentar el desempeño al desactivar elementos de la página	map	SIMPLIFIED_DISPLAY=true

Nombre del campo	Descripción	Formato	Valor
	de pago como el selector de idioma, los logotipos de la parte inferior de la página, etc.		
vads_trans_date	Fecha y hora del formulario de pago en el huso horario UTC	n14	Respete el formato AAAAMMDDHHMMSS Ejemplo: 20200101130025
vads_trans_id	Número de la transacción	an6	Ejemplo : xrT15p
vads_payment_config	Tipo de pago	enum	SINGLE para un pago único MULTI para un pago en vencimientos
vads_page_action	Acción a realizar	enum	PAYMENT
vads_ctx_mode	Adquisición de los datos en la plataforma de pago	enum	TEST o PRODUCTION
vads_action_mode	Modo de adquisición de la información del medio de pago	enum	INTERACTIVE
signature	Firma que garantiza la integridad de las solicitudes intercambiadas entre el sitio web vendedor y la plataforma de pago. El cálculo de su valor se indica aquí: Calcular la firma en la página 14.	ans44	Ejemplo: NrHSHyBBBc +TtcauudspNHQ5cYcy4tS4ljvdC0ztFe8=

2. Utilice los campos a continuación para gestionar el regreso a la aplicación móvil al final del pago.

Un pago puede terminar en 4 estados diferentes:

- Pago aceptado
- Pago rechazado
- Pago en error
- Pago abandonado por el comprador

Debe asociar una URL a cada estado:

Nombre del campo	Descripción	Formato	Valor
vads_url_success	URL donde se redirigirá al comprador si el pago es exitoso .	ans..1024	Ejemplo: http://webview.success
vads_url_refused	URL donde se redirigirá al comprador si el pago es denegado .	ans..1024	Ejemplo: http://webview.refused
vads_url_cancel	URL donde se redirigirá al comprador en caso de abandono o expiración (timeout).	ans..1024	Ejemplo: http://webview.cancel
vads_url_error	URL donde se redirigirá al comprador en caso de error .	ans..1024	Ejemplo: http://webview.error

3. Utilice los campos a continuación para configurar los plazos de redirección a la aplicación móvil al final del pago.

Nombre del campo	Descripción	Formato
vads_redirect_success_timeout	Define el plazo de espera antes del redirección, luego de un pago exitoso. Este plazo se expresa en segundos y debe estar entre 0 y 300 segundos. Asigne a este campo el valor "0" para no mostrar el recibo del pago y redirigir automáticamente al comprador a la aplicación móvil.	n..3
vads_redirect_error_timeout	Define el plazo de espera antes del redirección, luego de un pago denegado.	n..3

Nombre del campo	Descripción	Formato
	Este plazo se expresa en segundos y debe estar entre 0 y 300 segundos. Asigne a este campo el valor "0" para no mostrar la página de rechazo del pago y redirigir automáticamente al comprador a la aplicación móvil.	

4. Añade los otros campos opcionales en función de sus necesidades (consulte los siguientes capítulos).

Transmitir los datos del comprador

El vendedor puede transmitir la información del comprador (dirección de correo electrónico, civilidad, número de teléfono, etc.). Estos datos constituirán la información de facturación.

Todos los datos que se transmitirán a través del formulario de pago se mostrarán en el Back Office Vendedor consultando el detalle de la transacción (pestaña **Comprador**).

Utilice los campos opcionales a continuación según la necesidad. *Estos campos se devolverán en la respuesta con el valor pasado en el formulario.*

Nombre del campo	Descripción	Formato	Valor
vads_cust_email	Dirección de correo electrónico del comprador	ans..150	Ejemplo: abc@example.com
vads_cust_id	Referencia del comprador en el sitio web vendedor	an..63	Ejemplo: C2383333540
vads_cust_title	Estado civil del comprador	an..63	Ejemplo: Monsieur
vads_cust_status	Estado	enum	PRIVATE: para un particular COMPANY: para una empresa
vads_cust_first_name	Nombre	ans..63	Ejemplo: Laurent
vads_cust_last_name	Apellido	ans..63	Ejemplo: Durant
vads_cust_legal_name	Razón social del comprador	an..100	Ejemplo: D. & Cie
vads_cust_cell_phone	Número de teléfono móvil	an..32	Ejemplo: 06 12 34 56 78
vads_cust_address_number	Número de calle	ans..64	Ejemplo: 109
vads_cust_address	Dirección postal	ans..255	Ejemplo: Rue de l'innovation
vads_cust_address2	Segunda línea de dirección	ans..255	Ejemplo:
vads_cust_district	Barrio	ans..127	Ejemplo: Centre ville
vads_cust_zip	Código postal	an..64	Ejemplo: 31670
vads_cust_city	Ciudad	an..128	Ejemplo: Labège
vads_cust_state	Estado / región	ans..127	Ejemplo: Occitanie
vads_cust_country	Código del país según ISO 3166 alpha-2	a2	Ejemplo: "FR" para Francia, "PF" para la Polinesia Francesa, "NC" para la Nueva Caledonia, "US" para Estados Unidos.

Nota

Los campos **vads_cust_phone** y **vads_cust_cell_phone** aceptan todos los formatos:

Ejemplos:

- 0123456789
- +33123456789
- 0033123456789
- (00.571) 638.14.00
- 40 41 42 42

Transmitir los datos del pedido

El vendedor puede transmitir la información del pedido (número del pedido, descripción, contenido del carrito de compra, etc.).

Estos datos se mostrarán en el Back Office Vendedor consultando los detalles de la transacción.

1. Utilice los campos opcionales a continuación según la necesidad. Estos campos se devolverán en la respuesta con el valor pasado en el formulario.

Nombre del campo	Descripción	Formato	Valor
vads_order_info	Información adicional sobre el pedido	an..255	Ejemplo: Código interphone 3125
vads_order_info2	Información adicional sobre el pedido	an..255	Ejemplo: Sin ascensor
vads_order_info3	Información adicional sobre el pedido	an..255	Ejemplo: Express
vads_nb_products	Número de artículos que se encuentran en el carrito de compras	n..12	Ejemplo: 2
vads_product_ext_idN	Código de barras del producto en el sitio web vendedor. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).		Ejemplo: vads_product_ext_id0 = "0123654789123654789" vads_product_ext_id1 = "0223654789123654789"
vads_product_labelN	Descripción del artículo. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).	ans..255	Ejemplo: vads_product_label0 = "Estancia 3 días con fecha" vads_product_label1 = "Concierto privado"
vads_product_amountN	Monto del artículo expresado en la unidad más pequeña de la moneda. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).	n..12	Ejemplo: vads_product_amount0 = "32150" vads_product_amount1 = "10700"
vads_product_typeN	Tipo del artículo. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).	enum	Ejemplo: vads_product_type0 = "TRAVEL" vads_product_type1 = "ENTERTAINMENT"
vads_product_refN	Referencia del artículo. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).	an..64	Ejemplo: vads_product_ref0 = "1002127784" vads_product_ref1 = "1002127693"
vads_product_qtyN	Cantidad del artículo. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).	n..12	Ejemplo: vads_product_qty0 = "1" vads_product_qty1 = "1"
vads_product_vatN	IVA del artículo. N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).	n..12	

2. Complete el campo **vads_nb_products** con la cantidad de artículos que se encuentran en el carrito de compra.

Nota:

Este campo se convierte en obligatorio para que se tenga en cuenta el carrito de compra.

*Al completarlo, se mostrará la pestaña **Carrito de compra** en los detalles de una transacción desde Back Office Vendedor.*

*Sin embargo, si los otros campos que comienzan con **vads_product_** no se completan, la pestaña no contendrá ninguna información. Por este motivo, al completar el campo **vads_nb_products**, se vuelve obligatorio llenar los otros campos que comienzan con **vads_product_**. par **vads_product_**.*

3. Complete el campo **vads_product_amountN** con el monto de varios artículos que se encuentran en el carrito de compra en la menor unidad de la moneda.

N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).

4. Complete el campo **vads_product_typeN** con el valor correspondiente al tipo de artículo.

N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).

Valor	Descripción
FOOD_AND_GROCERY	Alimentos y productos comestibles.
AUTOMOTIVE	Automotriz / Motocicleta
ENTERTAINMENT	Entretenimiento / Cultura
HOME_AND_GARDEN	Casa y jardín
HOME_APPLIANCE	Equipo de la casa
AUCTION_AND_GROUP_BUYING	Subastas y compras a granel
FLOWERS_AND_GIFTS	Flores y regalos
COMPUTER_AND_SOFTWARE	Computadoras y software
HEALTH_AND_BEAUTY	Salud y belleza
SERVICE_FOR_INDIVIDUAL	Servicios humanos
SERVICE_FOR_BUSINESS	Servicios de negocios
SPORTS	Deportes
CLOTHING_AND_ACCESSORIES	Ropa y accesorios
TRAVEL	Viajes
HOME_AUDIO_PHOTO_VIDEO	Sonido, imagen y video
TELEPHONY	Telefonía

5. Complete el campo **vads_product_labelN** con la etiqueta de cada artículo que se encuentra en el carrito de compra.

N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).

6. Complete el campo **vads_product_qtyN** con la cantidad de cada artículo que se encuentra en el carrito de compra.

N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).

7. Complete el campo **vads_product_refN** con la referencia de cada artículo que se encuentra en el carrito de compra.

N corresponde al índice del artículo (0 para el primero, 1 para el segundo...).

8. Compruebe el valor del campo **vads_amount**. Debe corresponder al monto total del pedido.

Transmitir los datos de entrega

El vendedor puede transmitir los datos de entrega del comprador (dirección, civilidad, número de teléfono, etc.).

Estos datos se mostrarán en el Back Office Vendedor consultando los detalles de la transacción (pestaña **Entrega**).

Utilice los campos opcionales a continuación según la necesidad. *Estos campos se devolverán en la respuesta con el valor pasado en el formulario.*

Nombre del campo	Descripción	Formato	Valor
vads_ship_to_city	Ciudad	an..128	Ejemplo: Bordeaux
vads_ship_to_country	Código del país según la norma ISO 3166 (obligatorio para activar una o varias acciones si el perfil Control del país de entrega está activado).	a2	Ejemplo: FR
vads_ship_to_district	Barrio	ans..127	Ejemplo: La Bastide
vads_ship_to_first_name	Nombre	ans..63	Ejemplo: Albert
vads_ship_to_last_name	Apellido	ans..63	Ejemplo: Durant
vads_ship_to_legal_name	Razón social	an..100	Ejemplo: D. & Cie
vads_ship_to_phone_num	Número de teléfono	ans..32	Ejemplo: 0460030288
vads_ship_to_state	Estado / región	ans..127	Ejemplo: Nouvelle aquitaine
vads_ship_to_status	Define el tipo de dirección de entrega.	enum	PRIVATE : para entrega a un particular COMPANY : para entrega a una empresa
vads_ship_to_street_number	Número de calle	ans..64	Ejemplo: 2
vads_ship_to_street	Dirección postal	ans..255	Ejemplo: Rue Sainte Catherine
vads_ship_to_street2	Segunda línea de dirección	ans..255	
vads_ship_to_zip	Código postal	an..64	Ejemplo: 33000

5.2. Calcular la firma

Para poder calcular el valor del campo **firma** debe disponer:

- de los campos cuyos nombres comienzan con **vads_**;
- del tipo de algoritmo elegido en la configuración de la tienda;
- de la **clave**.

El valor de la clave está disponible en el Back Office Vendedor en el menú **Configuración > Tienda > pestaña Clave(s)**.

El tipo de algoritmo se define en su Back Office Vendedor en el menú **Configuración > Tienda > pestaña Configuración**.

Para máxima seguridad, se recomienda utilizar el algoritmo HMAC-SHA-256 además de una clave alfanumérica.

El algoritmo SHA-1 está en desuso, pero se mantiene por razones de compatibilidad.

Atención: No debe utilizar las claves de API REST para calcular la firma de su formulario de pago.

Solo el algoritmo HMAC-SHA-256 se implementa en nuestro ejemplo de código.

Para calcular la firma:

1. Ordene los campos cuyo nombre comienza con **vads_** en orden alfabético.
2. Asegúrese de que todos los campos estén codificados en UTF-8.
3. Concatene los valores de estos campos separándolos con el carácter "+".
4. Concatene el resultado con la clave de prueba o de producción separándolos con el carácter "+".
5. De acuerdo con el algoritmo de firma definido en la configuración de su tienda:
 - a. si su tienda está configurada para usar "SHA-1", aplique la función de hash **SHA-1** en el string obtenido en el paso anterior. **Depreciado.**
 - b. si su tienda está configurada para usar "HMAC-SHA-256", calcule y codifique en formato Base64 la firma del mensaje usando el algoritmo **HMAC-SHA-256** con los siguientes parámetros:
 - la función hash SHA-256,
 - la clave de prueba o de producción (según el valor del campo **vads_ctx_mode**) como clave compartida,
 - el resultado del paso anterior como mensaje a autenticar.
6. Guarde el resultado del paso anterior en el campo **signature**.

Ejemplo de parámetros enviados a la plataforma de pago:

```
<form method="POST" action="https://secure.payzen.eu/vads-payment/entry.silentInit.a">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="5124" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_config" value="SINGLE" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20170129130025" />
<input type="hidden" name="vads_trans_id" value="123456" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="ycA5Do5tNvsnKdc/eP1bj2xa19z9q3iWPpy9/rpesfS0=" />

<input type="submit" name="pagar" value="Pagar" />
</form>
```

Este ejemplo de formulario se desglosa de la siguiente manera:

1. Se organizan en orden **alfabética** los campos cuyo nombre comienza con **vads_** :

- vads_action_mode
- vads_amount
- vads_ctx_mode
- vads_currency
- vads_page_action
- vads_payment_config
- vads_site_id
- vads_trans_date
- vads_trans_id
- vads_version

2. Se concatena el valor de estos campos con el carácter "+":

```
INTERACTIVE+5124+TEST+978+PAYMENT+SINGLE+12345678+20170129130025+123456+V2
```

3. Se agrega el valor de la clave de prueba al final del string, separado por el carácter "+". En este ejemplo, la clave de prueba es **1122334455667788**

```
INTERACTIVE+5124+TEST+978+PAYMENT+SINGLE+12345678+20170129130025+123456+V2+1122334455667788
```

4. Si usa el algoritmo SHA-1, aplíquelo al string resultante.

El resultado a transmitir en el campo firma es: **59c96b34c74b9375c332b0b6a32e6deec87de2b**

5. Si su tienda está configurada para usar "HMAC-SHA-256", calcule y codifique en formato Base64 la firma del mensaje usando el algoritmo **HMAC-SHA-256** con los siguientes parámetros:

- la función hash SHA-256,
- la clave de prueba o de producción (según el valor del campo **vads_ctx_mode**) como clave compartida,
- el resultado del paso anterior como mensaje a autenticar.

El resultado a transmitir en el campo firma es:

ycA5Do5tNvsnKdc/eP1bj2xa19z9q3iWPpy9/rpesfS0=

5.3. Transferencia de la solicitud de pago

La API de creación del pago está disponible en modo POST en la siguiente dirección:

<https://secure.payzen.eu/vads-payment/entry.silentInit.a>

IMPORTANTE

La URL de la API de creación de pago es distinta de la URL de la página de pago, tal como se describe en la *Guía de implementación de la API de formularios*.

5.4. Recepción de la URL de notificación

La plataforma de pago devuelve una respuesta en formato JSON que contiene un código de estado HTTP de éxito o error.

Éxito

En caso de éxito, la plataforma de pago emite un código de estado HTTP 200 **OK**.

La respuesta contiene la URL de pago hacia la cual la aplicación debe redirigir al comprador.

```
{
  "status": "INITIALIZED",
  "redirect_url": "https://secure.payzen.eu:443/vads-payment/
exec.refresh.a;jsessionid=CE2Cb9daEDe7f6dBF31FE65e.vadpayment01bdx"
}
```

Error

En caso de error, la plataforma de pago emite un código de estado HTTP 400 **Bad Request** o 500 **Internal Server Error**.

La respuesta contiene los detalles del error.

```
{
  "status": "ERROR",
  "error": { "code": "09", "value": "Missing or invalid parameter value" }
}
```

Para más información, consulte la lista de los códigos de error de la API de formularios:

<https://payzen.io/fr-FR/error-code/error-00.html>

5.5. Procesamiento de la notificación de fin de pago (IPN)

Una vez realizado el pago, la plataforma de pago informa el resultado de la transacción al servidor del vendedor.

Los datos se envían a la URL de notificación definida en el Back Office Vendedor.

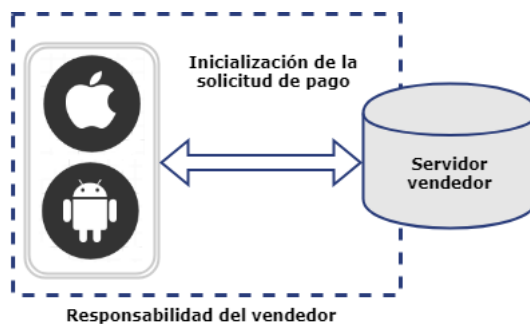
Consulte la *Guía de implementación de la API de formularios* para más información sobre cómo configurar las reglas de notificación y el análisis de los datos transmitidos.

5.6. Ejemplo de código

Consulte el ejemplo completo para desplegar el servidor del comerciante:

<https://github.com/lyra/webview-payment-sparkjava-integration-sample>

6. FASE 2: LA APLICACIÓN MÓVIL



La integración en su aplicación móvil debe seguir los siguientes pasos:

1. Iniciar la solicitud de pago transmitiendo los datos necesarios,

La aplicación genera una “payload” (carga útil) con los datos del carrito de compra, los datos personales del comprador, la información de entrega y transmite la solicitud de pago al servidor del comerciante a través de una petición POST.

2. Visualización de la página de pago en WebView,

La aplicación inicia una webview y abre la página de pago utilizando la URL retornada por la plataforma de pago.

3. Detección del fin del pago.

La aplicación móvil debe analizar las diferentes URL que pasan por la webview. Como las URL de retorno las define el servidor del comerciante, usted mantiene el control sobre la cinemática de pago y puede decidir en qué momento volver a su aplicación nativa.

Encuentre todos los detalles de la integración en el siguiente enlace:

iOS <https://github.com/lyra/webview-payment-ios-integration-sample/>
Android <https://github.com/lyra/webview-payment-android-integration-sample>

6.1. Cámara para escanear tarjeta

Puede utilizarse la cámara digital del equipo móvil para completar previamente los datos de la tarjeta en el momento del pago.

En el siguiente enlace, encontrará un ejemplo completo de integración:

iOS https://github.com/lyra/webview-payment-iOS-integration-sample/tree/card_scanning
Android https://github.com/lyra/webview-payment-android-integration-sample/tree/card_scanning

Este ejemplo utiliza librerías externas desarrolladas por terceros.

IMPORTANTE

Lyra Network No garantiza ni es responsable de la calidad de las librerías externas.
El uso de dichas librerías no es compatible con PCI-DSS.

6.2. NFC para escanear tarjeta

Puede utilizarse el módulo NFC del equipo móvil para completar previamente los datos de la tarjeta en el momento del pago.

Aquí, encontrará un ejemplo completo de integración:

iOS No disponible.

Android https://github.com/lyra/webview-payment-android-integration-sample/tree/card_scanning_by_nfc

Este ejemplo utiliza librerías externas desarrolladas por terceros.

IMPORTANTE

**Lyra Network No garantiza ni es responsable de la calidad de las librerías externas.
El uso de dichas librerías no es compatible con PCI-DSS.**