Guía de Integración TPV eCommerce mediante inSite

Edición septiembre 2018

Santander

Índice de contenidos

1 Objetivo de esta guía

En este documento de describe cómo implementar en una tienda web la conexión **inSite** del TPV eCommerce, un modelo de conexión que permite recoger los datos de pago del cliente sin que éste tenga que abandonar la página web del comercio.

Las ventajas de este tipo de integración son varias y se describen con mayor detalle en el siguiente epígrafe. El objetivo principal es el de disponer de un proceso de pago rápido, sencillo e integrado al máximo en las páginas de la tienda web, adaptado completamente al diseño del comercio online, fácil de usar y de integrar, pero a la vez que mantiene la seguridad sobre los datos de pago introducidos por el cliente.

Esta guía se centra en las particularidades de este este tipo de integración. Para conceptos generales del funcionamiento del servicio de TPV eCommerce por favor consulte la documentación correspondiente.

2 Conceptos y ventajas de la conexión inSite

Con la solución de pago ***inSite*** el comercio o tienda online consigue una serie de ventajas que favorecen el aumento de la conversión de ventas:

- Una **experiencia de pago sencilla y satisfactoria** para sus clientes, al estar **totalmente integrada** en las páginas web del comercio y sin saltos de navegación.
- **Mayor control** del flujo de checkout y pago, ya que toda petición se realiza de forma síncrona por parte del servidor de la tienda web y sin necesidad de procesos asíncronos de "escucha".
- **Facilidad de uso en su integración**,
- **Alto nivel de seguridad,** similar a la solución basada en redirección del cliente hacia una página de pago externa.

En definitiva, además de un proceso de pago totalmente integrado en el checkout al comprador, se permite al comercio una mayor **flexibilidad y control** en el proceso de pago, pudiendo además separar los pasos de captura de datos y ejecución de la operación. A la hora de integrar la conexión **inSite** existen **dos posibilidades**:

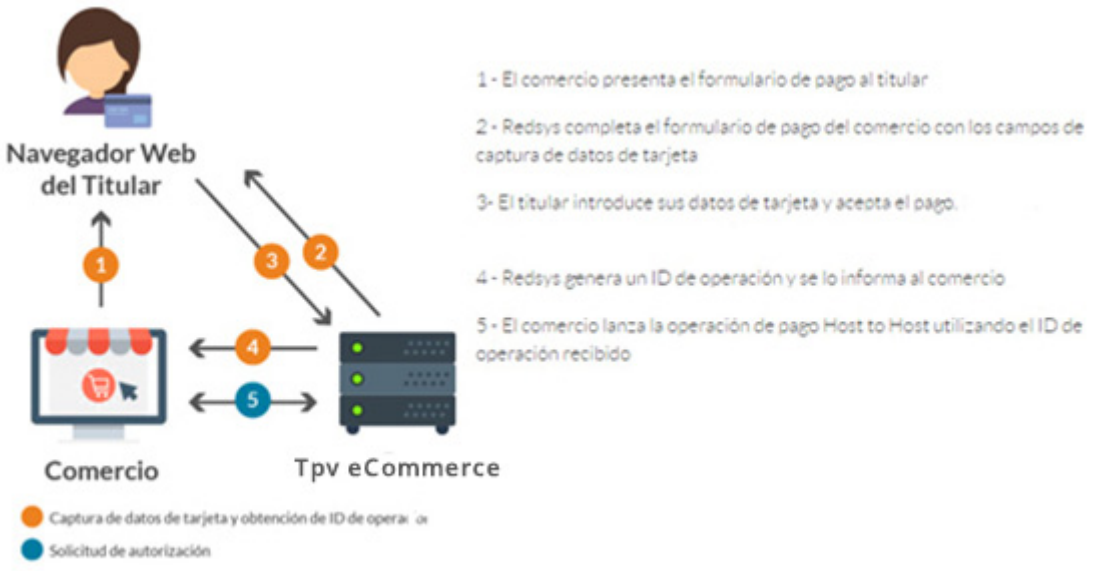
- Integración unificada (todo en uno)
- Integración por elementos independientes

En ambos casos, la integración se puede realizar utilizando fragmentos de código que se exponen como ejemplos, donde sólo se requiere cambiar valores propios como el identificador del comercio o las claves utilizadas. Además, como ayuda adicional se proveen librerías para los principales lenguajes de programación. En la conexión inSite, se facilitan a la tienda online las piezas o "campos" necesarios del formulario de pago de forma que se integran uno a uno (o como un conjunto) perfectamente incrustados en la página checkout de la tienda online y además cada elemento permite personalización del diseño con estilos configurables, en

perfecta sintonía del diseño del resto de la página web del comercio. La seguridad se preserva de forma que el formulario resultante con la información de pago de los clientes queda inaccesible al mismo servidor del comercio o incluso de terceros que hayan podido comprometer el servidor web del comercio.

3 Descripción general del flujo

El siguiente esquema presenta el flujo general de una operación realizada con el nuevo esquema del TPV eCommerce.



En resumen, los datos de pago introducidos por el cliente son enviados desde la página del comercio al TPV eCommerce, donde se almacenan temporalmente y se asocian a un Id de Operación que se devuelve al comercio. Con este Id de Operación (que viene a ser un "alias" de los datos de pago del cliente) el comercio puede solicitar posteriormente y directamente al TPV eCommerce la realización de la operación de pago deseada.

4 Página de Pago – Obtención de ID de operación

Como primer paso para poder integrar los campos de introducción de datos de tarjeta directamente en su propia página web, se debe incluir el fichero Javascript alojado en el servidor del TPV eCommerce con la siguiente línea de código (el fichero varía según se vaya a usar el entorno de pruebas o el entorno de producción real):

- Entorno de Integración para Pruebas

```
<script src="https://sis-i.redsys.es:25443/sis/NC/inte/redsys2.js"></script>
```

- Entorno para Producción:

```
<script src="https://sis.redsys.es/sis/NC/redsys.js"></script>
```

El siguiente paso para incluir los elementos del formulario de pago depende de la alternativa que se desee implementar. A la hora de integrar la conexión **inSite** existen **dos posibilidades**:

a) **Integración unificada (todo en uno)**: Los elementos de pago, como las cajas de introducción de número de tarjeta, fecha de caducidad, CVV2 y botón de pago se incrustan como un solo elemento que se adapta a la página del comercio (responsive) , con diseño ligero y estilos CSS personalizables. Incluye por defecto ayudas interactivas animadas y una buena usabilidad al usuario.

b) **Integración por elementos independientes**: los campos se deben incrustar cada uno de forma independiente dentro de la página web de la tienda web, lo que permite el control total del diseño, posición, etc.

4.1 Integración unificada (todo en uno)

En esta modalidad de la integración inSite se proveerá un único iframe de tamaño muy ajustado en el que se incluirá el formulario de pago al completo. En cuanto a la personalización del mismo, se podrán aplicar los estilos CSS que el comercio requiera a los diferentes elementos.

Incluye elementos interactivos que facilitan la usabilidad, como el reconocimiento de la marca de tarjeta, mostrando el logo de la misma, verificación de los formatos y contenidos conforme los introduce el usuario y resaltando visualmente al momento si alguno es incorrecto (check digit, fecha cad...).

Ejemplo:

Una vez importado el fichero JS, se deberá crear el formulario de pago. Para recoger de forma segura los datos de tarjeta, el TPV eCommerce creará y alojará los campos de introducción de dichos datos.

Se deberán crear un único contenedor, con un id único, ya que se deberá indicar para que se genere iframe con los elementos en él.

```
<div id="card-form"></div>
```

Al finalizar la carga de la página, se deberá ejecutar una función proporcionada por el propio API, incluyéndose en el atributo onLoad de la misma. De esta forma se asegura que en el entorno del procesador se disponga de los datos necesarios para la generación del ID de la forma correcta:

```
<body onload="loadRedsysForm()">
```

A continuación, se incluirá una función de escucha de mensajes (listener) para recibir el ID de operación cuando éste se genere. Para facilitar la integración, el procesador proporciona en su API una función (storeIdOper) en la que se deberá indicar en qué elemento del DOM se debe almacenar el ID de operación una vez sea generado. En este ejemplo se crea un input de tipo "hidden".

```
<input type="hidden" id="token" ></input>
<!-- Listener de recepción de ID de operación -->
window.addEventListener("message", function
receiveMessage(event) {

        storeIdOper(event,"token");
});
```

Una vez preparado el envío de datos y la posterior recepción, se llamará a la función proporcionada para generar los elementos de introducción de datos de tarjeta:

```
<!-- Petición de carga de iframe -->
getInSiteForm('card-form', estiloBoton, estiloBody, estiloCaja, estiloInputs, 'Pagar con TPV', fuc, terminal, merchantOrder);
```

Como parámetros de las funciones se indicará el id del contenedor reservado para su generación, así como el estilo requerido para los diferentes elementos (formato CSS). En esta modalidad, se podrán incluir estilos para diferentes elementos:

- *Botón de pago* → Se permite la personalización completa del botón de pago.
- *Cuerpo del formulario* → Se recomienda utilizar para establecer un color de fondo o modificar el color o estilo de los textos.
- *Caja de introducción de datos* → Se podrá establecer un color de fondo diferenciado para la caja de introducción de datos. El color del texto aplicado en este elemento se aplicará al "placeholder" de los elementos.
- *Inputs de introducción de datos* → Se recomienda su uso si se quiere utilizar un tipo de letra diferente o modificar el color del texto de los campos de introducción de datos.

Adicionalmente, se podrá personalizar de igual forma el texto a incluir en el botón de pago y, por último, se deberá informar el valor del FUC, terminal y número de pedido en la petición de carga del iframe con el formulario de pago. De esta forma, cuando el cliente introduzca sus datos de tarjeta en los elementos generados por el TPV eCommerce y pulse el botón de pago, se generará y almacenará en el formulario del comercio un ID asociado a la operación para que éste formalice la compra sin necesidad de tratar datos de tarjeta.

4.2 Integración de elementos independientes

En esta modalidad de la integración inSite se permitirá a los comercios una total personalización de la página de pago, por lo que podrá colocar los campos de introducción de datos de tarjeta y el botón de pago con total libertad, al generar iframes diferenciados y personalizables con estilos para cada uno de ellos.

Una vez importado el fichero, se deberá crear el formulario de pago. Para recoger de forma segura los datos de tarjeta, el TPV eCommerce creará y alojará los campos de introducción de dichos datos.

Se deberán crear contenedores vacíos, con un id único, ya que se deberá indicar para que se genere el campo de introducción de datos en él.

```
<div class="cardinfo-card-number">
<label class="cardinfo-label" for="card-number">Numero de tarjeta</label>
<div class='input-wrapper' id="card-number"></div>
</div>
<div class="expiry-date">
<div class="cardinfo-exp-date">
<label class="cardinfo-label" for="expiration-date">Mes Caducidad (MM)</label>
<div class='input-wrapper' id="expiration-month"></div>
</div>
<div class="cardinfo-exp-date2">
<label class="cardinfo-label" for="expiration-date2">Ano Caducidad (AA)</label>
<div class='input-wrapper' id="expiration-year"></div>
</div>
<div class="cardinfo-cvv">
<label class="cardinfo-label" for="cvv">CVV</label>
<div class='input-wrapper' id="cvv"></div>
</div>
</div>
```

En este ejemplo, se trata de los elementos con id "*card-number*","*expiration- month*","*expiration-year*" y "*cvv*".

Al finalizar la carga de la página, se deberá ejecutar una función proporcionada por el propio API, incluyéndose en el atributo onLoad de la misma. De esta forma se asegura que en el entorno del TPV eCommerce se disponga de los datos necesarios para la generación del ID de la forma correcta:

```
<body onload="loadRedsysForm()">
```

A continuación, se incluirá una función de escucha de mensajes (listener) para recibir el ID de operación cuando éste se genere. Para facilitar la integración, el TPV eCommerce proporciona en su API una función (storeIdOper) en la que se deberá indicar en qué elemento del DOM se debe almacenar el ID de operación una vez sea generado. En este ejemplo se crea un input de tipo "hidden".

```
<input type="hidden" id="token" ></input> <!-- Listener de recepción de ID de operación
--> window.addEventListener("message", function receiveMessage(event) {

storeIdOper(event,"token");

});
```

Una vez preparado el envío de datos y la posterior recepción, se llamará a las funciones proporcionadas para generar los elementos de introducción de datos de tarjeta:

```
<!-- Petición de carga de iframes --> getCardInput('card-number',estilos);
getExpirationMonthInput('expiration-month' estilos);
```

```
getExpirationYearInput('expiration-year', estilos); getCVVInput('cvv', estilos); getPayButton('boton', estilos, 'Pagar con Redsys', fuc, terminal, merchantOrder);
```

Como parámetros de las funciones se indicará el id del contenedor reservado para su generación, así como el estilo requerido para el mismo (formato CSS). Adicionalmente, se podrá personalizar el texto a incluir en el botón de pago y, por último, se deberá informar el valor del FUC, terminal y número de pedido en la petición de carga del iframe con el botón de pago.

De esta forma, cuando el cliente introduzca sus datos de tarjeta en los elementos generados por el TPV eCommerce y pulse el botón de pago, se generará y almacenará en el formulario del comercio un ID asociado a la operación para que éste formalice la compra sin necesidad de tratar datos de tarjeta.

5 Solicitud de operación a partir de ID de operación

Una vez recibido y almacenado el ID de operación por parte del comercio según se ha descrito en los apartados anteriores, podrá lanzar la operación de autorización utilizando la API de conexión directa con el TPV eCommerce (webservice).

Se pone a disposición de los comercios de librerías que simplifican esta conexión para Java y PHP. Su descarga está disponible en la sección de descargas de la siguiente web:

https://sis-d.redsys.es/web-comercio/descargas.html

La descarga de las librerías incluye documentación de ayuda para su uso.

5.1 Implementación sin uso de las librerías de ayuda

Si no se desea utilizar las librerías de ayuda o se quiere implementar para otros lenguajes de programación, pueden implementar directamente la llamada webservice, al TPV eCommerce. Para obtener más información, se recomienda consultar la documentación de Integración vía Webservice. **En este caso, es importante tener en cuenta que al estar implementando la conexión inSite, en la petición webservice se enviará el parámetro *Ds_Merchant_idOper* generado en los pasos descritos anteriormente, en lugar recoger y enviar directamente los datos de tarjeta**.

6 Autenticación 3DSecure

Los comercios que utilicen la conexión **inSite** tienen la posibilidad de incluir el protocolo 3DSecure (3DS) para autenticar a los titulares y obtener un nivel adicional de protección ante fraude.

Incluir la autenticación 3DS implica redirigir la navegación del cliente hacia el servidor de autenticación del banco/entidad emisora de la tarjeta para que éste pueda solicitar las

credenciales necesarias. Este paso debe realizarse en un paso posterior al de recoger los datos de tarjeta descrito en los apartados anteriores.

Para utilizar la autenticación 3DS, el terminal del Tpv eCommerce debe estar configurado por parte de su entidad financiera para soportar autenticación 3D Secure. Igualmente podría ser necesario que por configuración del TPV eCommerce este paso no solo sea opcional, sino que sea requerido para la correcta autorización de las operaciones (si tiene dudas consulte con nosotros).

Para proceder a autenticar al cliente con 3DS, en la petición directa vía Webservice descrita en el punto anterior, deberá indicarse expresamente mediante el parámetro opcional *Ds_Merchant_DirectPayment* que deberá llevar asignado el valor "3DS", lo que indica que el comercio está preparado para manejar las autenticaciones 3DS.

Incluir este parámetro implica la activación del siguiente flujo:

1. El TPV eCommerce no procesa inmediatamente la solicitud de operación en cuanto la recibe, sino que, en su lugar, se verifica la posibilidad de iniciar el protocolo de autenticación (tanto para el comercio -configuración 3DS permitida- como para la tarjeta introducida por el cliente -capacidad 3DS disponible-). Si el terminal no está configurado para permitir 3DS o la tarjeta no requiere/soporta autenticación por 3DS, se procede a procesar la solicitud de operación sin requerir autenticación 3DS y se responderá de forma normal (flujo de 1 paso) del resultado de la misma.
2. Se obtiene, entre otros datos, la URL a la cual el comercio debe redirigir al navegador del cliente para su autenticación
3. El TPV eCommerce devuelve al comercio los datos necesarios para que inicie la redirección del titular a la URL de autenticación del banco emisor de la tarjeta.
4. El servidor del comercio debe redirigir la navegación del cliente hacia la URL del banco de la tarjeta, donde se procederá a la autenticación 3DS.
5. El servidor del comercio recibirá de vuelta la navegación del cliente tras el proceso de autenticación, incluyendo los datos resultado
6. El servidor del comercio deberá completar la transacción con un segundo mensaje webservice al TPV eCommerce, incluyendo los datos obtenidos como resultado de la autenticación 3DS.

Ejemplo:

```
<REQUEST>
        <DATOSENTRADA>

        <DS_MERCHANT_IDOPER>455097a74c21b761be86acb26c32609dce222e66</DS_MERCHANT_IDOPER>
                <DS_MERCHANT_ORDER>1234ased1711</DS_MERCHANT_ORDER>
                <DS_MERCHANT_AMOUNT>42</DS_MERCHANT_AMOUNT>
                <DS_MERCHANT_CURRENCY>978</DS_MERCHANT_CURRENCY>
                <DS_MERCHANT_MERCHANTCODE>999008881</DS_MERCHANT_MERCHANTCODE>
                <DS_MERCHANT_TERMINAL>1</DS_MERCHANT_TERMINAL>
                <DS_MERCHANT_TRANSACTIONTYPE>0</DS_MERCHANT_TRANSACTIONTYPE>
                <DS_MERCHANT_DIRECTPAYMENT>3DS</DS_MERCHANT_DIRECTPAYMENT>
        </DATOSENTRADA>
        <DS_SIGNATUREVERSION>HMAC_SHA256_V1</DS_SIGNATUREVERSION>
        <DS_SIGNATURE>PcQKZ9rHY1Q5Ta6Sf1uy64Skm7ZA53Oq5lEXM9QIj0Q=</DS_SIGNATURE>
</REQUEST>
```

En la respuesta de la petición WS se informarán, además de los datos de la operación (aún no realizada), los siguientes parámetros necesarios para redirigir al cliente a autenticarse con su banco emisor:

Ds_AcsUrl → URL del ACS al que hay que redirigir al usuario.

Ds_PaRequest → Parámetro necesario para identificar la operación

Ds_MD → Identificador de sesión.

Ejemplo:

```
<RETORNOXML>
        <CODIGO>0</CODIGO>
        <OPERACION>
                <Ds_Amount>42</Ds_Amount>
                <Ds_Currency>978</Ds_Currency>
                <Ds_Order>1234ased1711</Ds_Order>
                <Ds_Signature>8in9WK+f5wuT/CGvHYLZqpi6jHtMB/FjmKR0JJWkz8Y=</Ds_Signature>
                <Ds_MerchantCode>999008881</Ds_MerchantCode>
                <Ds_Terminal>1</Ds_Terminal>
                <Ds_TransactionType>0</Ds_TransactionType>
                <Ds_AcsUrl>https://sas-i.redsys.es:26443/sas/Secure</Ds_AcsUrl>
        <Ds_PaRequest>eJxVksFSwjAQhl+lU2e8SZKaQotLmGrr6KHIAB48xnQH6kALSav17U2gKB4y2W+T7P7
zb2Da7bbe J2pT1tXEZwPqe1ipuiir9cR/XT3eRP5UwGqjEdMlqlajgByNkWv0ymLi68LIsvuSpmSVGoajgMcB
ZRHldMgY47cRpTQeUV/APFngQUDfS9hWgwDIGW1RrTayagRIdbh/ngkehmwUA+kRdqifU7HMFnk2T/Js8fCUzFZA
Tmmo5A6Fqi2psvaur26jO2+vW3yXxlsgkOM5qLqtGv0trEwgZ4BWb8WmafZjQra1kttNbRogLgvkT9a8dZGxVbqy
EPlH0r2kCc3Tty5fZXSWvrI8Tb7smgBxN6CQDYqzGR6Nx4yPeQjkmAe5c+0FHXDrQg+wdz2SE7j8JYM1X9vZfIt4
FFn1ZwLs9nWF9oZ98BtDgUZZ/f32J/7hyTmsGmva5aA4C4LImX08cTVLaw2P2amoAyDuLennSPpfYKN/v+MHGte/
sQ==</Ds_PaRequest>
                <Ds_MD>d65b7415dcda4c1d1f92a864be56e5987f3702fd</Ds_MD>
        </OPERACION>
</RETORNOXML>
```

Una vez recibida la respuesta, el comercio deberá hacer una redirección al cliente (a través de un formulario POST) a la URL recibida, indicando los siguientes parámetros:

- PaReq → Se informará el valor recibido en el parámetro Ds_PaRequest de la petición WS.
- MD → Se informará el valor recibido en el parámetro *Ds_MD* de la petición WS.
- TermUrl → Se informará la URL del comercio a la que el servicio de autenticación de la tarjeta (ACS) retornará la navegación con la respuesta.

Una vez que el usuario de la tarjeta realice la autenticación en la URL de su banco emisor, la respuesta (vía redirección) del ACS al servidor del comercio contendrá los siguientes parámetros:

- PaRes→Parámetro necesario para enviar al TPV eCommerce una vez finalizado el proceso.
- MD→Identificador de sesión

Una vez recogidos los parámetros por el servidor de la tienda, debe lanzarse una nueva petición WS hacia el TPV eCommerce con los datos obtenidos del resultado de la autenticación del cliente, informando ambos parámetros adicionales en los campos *Ds_Merchant_PaResponse* y *Ds_Merchant_Md*.

```
<REQUEST>
        <DATOSENTRADA>

        <DS_MERCHANT_PARESPONSE>eJzFWFmzosqyfudXrOjz6O3NICrscK0TxaSABTILbwjIIIMCyvDrL66pV
3f0jd3n3odLh
FKVZCZfZWV9ldT6332RP92jukmr8vkb/hf27SkqgypMy/j5m2UK36lv/35Zm0kdRZwRBbc6elnDqGn8OHpK
w+dvddj4ad/5TYqXwXKxIkiawHAKI7EljuPknMIwjF5h317We6BHzatNU1wmm4VfBiS+nC+xHwbEYupM1wqfDN5R
vUyg/iLW6Ed3en0dJH7Zvqz94MqIygu5WOAreo2+d9dFVIvci8HrkN8DyOvsFijmGn0Tr9Ef9vvbo9VMQ+rT8AVm
oFc5gEHO7aHJYwpn4ZAD3fR7XqMPjXXot9HLB9wnjP4bJ/8mF2v0Vb6+PNyBorpNvskJ8df+egpdPUV2eKFX1Br9
7K2j/lKV0aQxGXy21+gPaBe/fMF+vuaT70m6Ng8v
….</DS_MERCHANT_PARESPONSE>
                <DS_MERCHANT_MD>d65b7415dcda4c1d1f92a864be56e5987f3702fd</DS_MERCHANT_MD>
                <DS_MERCHANT_ORDER>1234ased1711</DS_MERCHANT_ORDER>
                <DS_MERCHANT_AMOUNT>42</DS_MERCHANT_AMOUNT>
                <DS_MERCHANT_CURRENCY>978</DS_MERCHANT_CURRENCY>
                <DS_MERCHANT_MERCHANTCODE>999008881</DS_MERCHANT_MERCHANTCODE>
                <DS_MERCHANT_TERMINAL>1</DS_MERCHANT_TERMINAL>
                <DS_MERCHANT_TRANSACTIONTYPE>0</DS_MERCHANT_TRANSACTIONTYPE>
        </DATOSENTRADA>
        <DS_SIGNATUREVERSION>HMAC_SHA256_V1</DS_SIGNATUREVERSION>
        <DS_SIGNATURE>j7PusjxEhs39dvx3Oju3KjZ6+G2JnIX06o/reW9BXxM=</DS_SIGNATURE>
</REQUEST>
```

La respuesta de esta petición contendrá el resultado final de la operación.

```
<RETORNOXML>
<CODIGO>0</CODIGO>
<OPERACION>
<Ds_Amount>42</Ds_Amount>
<Ds_Currency>978</Ds_Currency>
<Ds_Order>1234ased1711</Ds_Order>
<Ds_Signature>KVwNaWSqsJWUlzpCWQENZbhHn6VJ8gRkw5CtF//cfeg=</Ds_Signature>
<Ds_MerchantCode>999008881</Ds_MerchantCode>
<Ds_Terminal>1</Ds_Terminal>
<Ds_Response>0000</Ds_Response>
<Ds_AuthorisationCode>745638</Ds_AuthorisationCode>
<Ds_TransactionType>0</Ds_TransactionType>
<Ds_SecurePayment>1</Ds_SecurePayment>
<Ds_Language>1</Ds_Language>
<Ds_Card_Type>C</Ds_Card_Type>
<Ds_MerchantData></Ds_MerchantData>
<Ds_Card_Country>724</Ds_Card_Country>
<Ds_Card_Brand>1</Ds_Card_Brand>
</OPERACION>
</RETORNOXML>
```


Santander