

## FlexCheckout

## Tabla de contenidos

### 1. Introducción

### 2. Flujo de datos típico

#### 2.1 DirectLink

#### 2.2 Uso con el Administrador de alias

### 3. Integración de FlexCheckout como una tokenización

#### 3.1 Campos de entrada

##### 3.1.1 Buscar OrderID duplicados

#### 3.2 Firma SHA para entrada

#### 3.3 Campos de salida

#### 3.4 Firma SHA para salida

### 4. Personalización

#### 4.1 Personalización básica

##### 4.1.1 Plantilla de Ingenico ePayments predeterminada

##### 4.1.2 Uso de hojas de estilo

#### 4.2 Personalización avanzada

##### 4.2.1 Uso de marcadores de posición

##### 4.2.2 Personalización de contenido de texto con XML

#### 4.3 Administrador de archivos de plantillas

##### 4.3.1 Subir archivos de plantilla

##### 4.3.2 Comprobar y administrar archivos cargados

##### 4.3.3 Integración

### 5. Marcadores de posición

## 1. Introducción

One Page Checkout con FlexCheckout le permite solicitar a sus clientes que introduzcan los detalles de su tarjeta de crédito de forma segura en el sistema PSP, tokenizar estos datos y reutilizar la clave (o el alias) en una futura transacción:

1. FlexCheckout: Redireccione a, o incluya en un iframe, la FlexCheckout en la que el titular de la tarjeta introducirá los datos de la tarjeta en nuestro sistema, donde se almacenarán de forma segura (incluido el CVC, pero solo por un tiempo limitado)
2. DirectLink: Envíe el pedido real con el alias generado, sin necesidad de enviar los datos de la tarjeta.

### Ventajas

- Tendrá todo el control sobre el aspecto de la página de pago gracias al mecanismo de plantilla flexible y, si incluye la FlexCheckout dentro de un I-frame, incluso podrá controlar la secuencia de compra.
- Tiene la ventaja de una página de pago segura que cumple con las normas PCI sin tener que almacenar usted mismo los datos de la tarjeta. Toda la información confidencial se introduce directamente en nuestra página alojada.
- Perfecta integración para los diversos escenarios de compra, incluido One Page Checkout.
- Puede ofrecer servicios de propensión de compra y venta cruzada en la página de compra final.
- Se admiten todas las principales marcas de tarjeta de crédito (VISA, MasterCard, American Express, Diners).

### Opciones requeridas

Tiene que tener habilitada al menos una de las opciones siguientes para utilizar FlexCheckout:

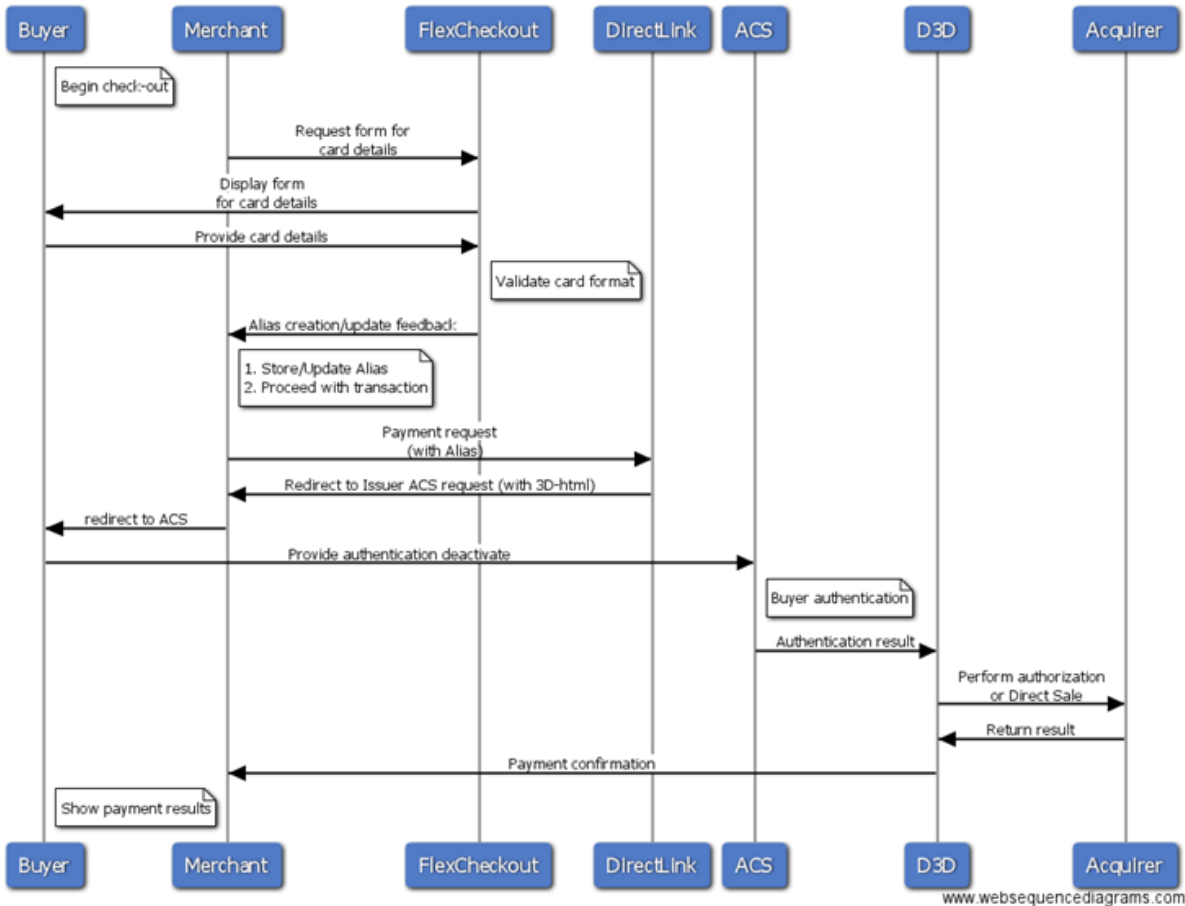
- One Page Checkout (ID de opción: OPC)
- Administrador de alias (ID de opción: REC1, REC2, RECX)

Normalmente también necesita tener activado DirectLink (pagos nuevos), si no está ya habilitado de forma predeterminada en la suscripción.

## 2. Flujo de datos típico

Puede redirigirse a (o incluir en) un iframe el FlexCheckout donde el titular de la tarjeta introducirá los datos de la tarjeta en nuestro sistema. Los datos de la tarjeta, conjuntamente con el CVC, se almacenan de forma segura pero solo durante un tiempo limitado. Para almacenar los datos de tarjeta de forma permanente, puede etiquetar el Alias como alias "Persistente". No obstante, de conformidad con las normas del Consejo de Seguridad de PCI, los CVC solo se pueden almacenar durante dos horas como máximo.

A través de DirectLink, también puede enviar el pedido real con el alias generado, sin enviar los datos de la tarjeta.



**Nota:** En el primer paso, no se realiza ninguna operación en la tarjeta. Nuestro sistema simplemente realiza una validación de formato básica, pero no puede garantizar que la tarjeta siga siendo válida o que tenga suficiente saldo para continuar.

### 2.1 DirectLink

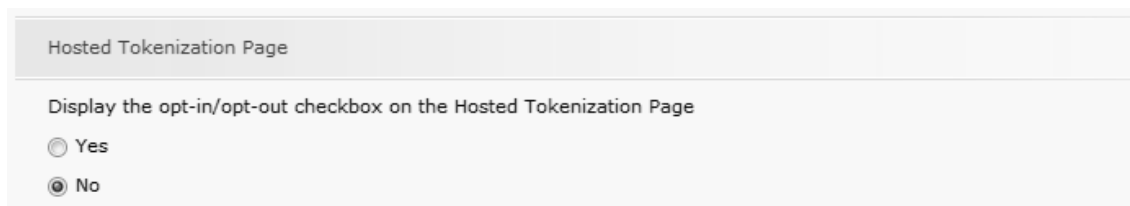
Para utilizar el alias o la clave generados con la FlexCheckout, debe enviar una transacción de DirectLink mediante nuestra implementación de DirectLink estándar. Vaya a [DirectLink](#) para obtener instrucciones de implementación.

Este mecanismo también es compatible con [DirectLink 3-D](#). Si desea más información acerca del uso de alias, vaya a [Administrador de alias](#).

### 2.2 Uso con el Administrador de alias

Si utiliza la opción Administrador de alias, puede poner a disposición una casilla de la FlexCheckout, lo que permite al cliente decidir si guardar o no sus detalles financieros.

Para ello, en la página de configuración del Administrador de alias de su cuenta de Ingenico ePayments, debe habilitar un valor para mostrar la casilla (el valor predeterminado es "No").



Hosted Tokenization Page

Display the opt-in/opt-out checkbox on the Hosted Tokenization Page

Yes

No

> Le recomendamos utilizar esta configuración en una configuración de One-Page-Checkout.

De forma alternativa, si ha integrado un proceso de compra en varios pasos, puede usar el [parámetro "StorePermanently"](#) en su integración, para controlar usted mismo cuándo debe mostrarse el cuadro de marcación.

### 3. Integración de FlexCheckout como una tokenización

Puede utilizar FlexCheckout de dos maneras:

1. Redirigiendo al cliente a FlexCheckout
2. Encapsulando FlexCheckout dentro de un iframe de modo que los clientes permanezcan en su página y crear una experiencia de One Page Checkout completa

**Advertencia:** No aconsejamos integrar FlexCheckout como una solución integrada en la aplicación pura ya que algunas características no funcionarían en algunos smartphones de gama media, sino una solución de sitio web móvil que utilice todas las capacidades de los navegadores.

Cuando el cliente se redirige a FlexCheckout, tendrá que introducir los datos de su tarjeta y enviarlos para tokenización en FlexCheckout. De esta forma, los detalles de la tarjeta nunca pasan por nuestro propio servidor web. Las URL para acceder a FlexCheckout son

- Prueba: <https://ogone.test.v-psp.com/Tokenization/HostedPage>
- Producción: <https://secure.ogone.com/Tokenization/HostedPage>

La FlexCheckout funciona únicamente con codificación de caracteres UTF-8. Si desea trabajar en ISO, póngase en contacto con nuestro Servicio de Atención al Cliente.

Si se utiliza la codificación ISO para calcular el SHA, los comerciantes tendrán que configurarla en el área de administración. Este ajuste valida el SHA entrante y el SHA saliente. También se utilizará para configurar el esquema de cifrado de URL para los parámetros devueltos”.

#### 3.1 Campos de entrada

Campo	Descripción	Formato	Obligatorio
ACCOUNT.PSPID	Identificación del comerciante	AN, 30	Si
ALIAS.ALIASID	Alias del cliente	AN, 50	No
ALIAS.ORDERID	Identificación del pedido	AN, 40	No
ALIAS.STOREPERMANENTLY	<p>Indique si desea almacenar un alias de forma temporal o indefinida. Los posibles valores son:</p> <ul style="list-style-type: none"> <li>• "N": el alias se eliminará transcurridas 2 horas</li> <li>• "Y": el alias se almacenará de forma indefinida para uso futuro</li> </ul> <p>&gt; Encontrará más información a continuación de este cuadro.</p> <p>Si se crea un Alias con el valor N y las transacciones se completan en el plazo de dos horas, la transacción también debe incluir esta combinación de parámetro/valor del alias que se va a eliminar. Si la transacción no contiene esta combinación de parámetro/valor, el alias se mantendrá para uso futuro.</p>	Y / N	No

**Alias.StorePermanently: notas importantes**

- El parámetro Alias.StorePermanently solo debe usarse en combinación con Administrador de alias. De forma predeterminada, el alias se almacenará de forma permanente.
- Si utiliza una integración de One-Page-Checkout, le recomendamos encarecidamente que habilite la opción "casilla suscribirse/no suscribirse" de su configuración del Administrador de alias para mostrar siempre la casilla en la FlexCheckout.
- Si ha activado la opción "opt-in casilla de entrada / salida" en la configuración de su Alias Manager, esto anula el uso del parámetro Alias.StorePermanently

### 3.1.1 Buscar OrderID duplicados

Para impedir que los hackers sustituyan detalles de tarjetas relacionados con una clave concreta (capturando el enlace para activar la solicitud y sustituyendo los datos de la tarjeta por datos de tarjetas auténticas de una tarjeta auténtica), realizamos una comprobación en el OrderID (ALIAS.ORDERID) que envía en la solicitud.

Si detectamos que el OrderID ya se ha utilizado para crear una clave, se rechazará la actualización del alias.

**Importante**

La comprobación en el OrderID solo funciona si se envía el **OrderID (ALIAS.ORDERID)**. Sin embargo, no es suficiente solo el OrderID; también es **necesario el AliasID (ALIAS.ALIASID)**. Si no se envía el AliasID, no se realizará la comprobación y se creará un nuevo alias.

Con la detección de un OrderID duplicado y con la solicitud bloqueada, se mostrará un mensaje de error genérico en el cliente:

"Se ha producido un error al procesar la solicitud. Inténtelo de nuevo más tarde o póngase en contacto con el comerciante."

Los errores generados por un OrderID duplicado se pueden detectar cuando se active el modo de depuración en los Registros de error (póngase en contacto con nuestro Servicio de Atención al Cliente).

## 3.2 Firma SHA para entrada

Para comprobar la integridad de los datos, necesitamos que todas las solicitudes vayan acompañadas de una firma SHA, [de la misma forma que para las e-Commerce transacciones](#).

Nuestro sistema utilizará el algoritmo SHA según se define en la pestaña Parámetros de seguridad global de su página Información técnica. Sigue siendo posible volver a cambiar este algoritmo a SHA-1 o SHA-256.

**Ejemplo**

Campos (en orden alfabético):

Parameters.AcceptUrl: https://www.myshop.com/ok.html

Parameters.ExceptionUrl: https://www.myshop.com/nok.html

Account.PspId: test1

Card.Brand:VISA

Frase de contraseña secreta (según se define en Información técnica): Mysecretsig1875!?

Cadena para la que se va a generar el hash:

ACCOUNT.PSPID=test1Mysecretsig1875!?CARD.BRAND=VISAMysecretsig1875!?PARAMETERS.ACCEPTURL=https://www.myshop.com/ok.htmlMysecretsig1875!?PARAMETERS.EXCEPTIONURL=https://www.myshop.com/nok.htmlMysecretsig1875!?

Firma SHA resultante (SHA-512):

563DC909F70BA5DDD470D69C1B390E7D1C1C47705AC5801B27038446D7033B5787728EA  
754EF72E7FA2436FC5962E34E20DF64E7F9139893A33653F118816818

### 3.3 Campos de salida

Se le pueden devolver los campos siguientes para que reciba el estado de la creación/actualización del alias. Para que se incluyan en la respuesta, deben estar previamente configurados en los parámetros de la respuesta dinámica de su cuenta Ingenico ePayments (Configuración > Información técnica > Respuesta transacción > Alias gateway y Tokenization: Parámetros dinámicos).

El SHASIGN no es opcional y, por tanto, siempre se devolverá.

Campo	Descripción	Longitud máx.
ALIAS.ALIASID	Alias enviado por el comerciante o alias generado por PSP (según el formato de GUID de 32 dígitos. Ejemplo: 34F5302C-85D7-4F35-BDF5-103CCEC2FB61)	50
ALIAS.NCERROR	Código de error	50
ALIAS.NCERRORCARDNO	Código de error para CARDNO	50
ALIAS.NCERRORCN	Código de error para CN	50
ALIAS.NCERRORCVC	Código de error para CVC	50
ALIAS.NCERRORED	Código de error para ED	50
ALIAS.ORDERID	El identificador único del pedido. (enviado por el comerciante o generado por el sistema)	40
ALIAS.STATUS	Resultado de la creación del alias <ul style="list-style-type: none"> <li>• 0=OK</li> <li>• 1=NOK</li> <li>• 2=Alias actualizado</li> <li>• 3=Cancelado por el cliente</li> </ul>	1
ALIAS.STOREPERMANENTLY	Indica la elección entre almacenar el alias de forma temporal (N) o indefinida (Y).	1 (Y/N)
CARD.BIC	Código de identificación del banco, se utiliza solo para domiciliaciones bancarias	8
CARD.BIN	Métodos de pago de tipo tarjeta de crédito	6
CARD.BRAND	Marca del método de pago	25
CARD.CARDHOLDERNAME	Nombre del titular de la tarjeta	50
CARD.CARDNUMBER	Tarjeta con X para sustituir la información confidencial.	35



Campo	Descripción	Longitud máx.
	Ejemplo: XXXXXXXXXXXXXXX1111  Nota: En caso de error, la tarjeta también se enmascarará.	
CARD.CVC	Código de verificación de tarjeta para tarjetas de crédito, con X para reemplazar los datos confidenciales.  Ejemplo: XXX	6
CARD.EXPIRYDATE	Fecha de caducidad, por ejemplo, 0220 (febrero de 2020)	4
PARAMPLUS	Campo de verificación: datos suministrados en la entrada	/
<b>SHASIGN</b>	<b>Firma SHA para salida</b>	<b>128</b>

Los parámetros seleccionados se añadirán a la URL de retorno definido en la solicitud (PARAMETERS.ACCEPTURL o PARAMETERS.EXCEPTIONURL) con el fin de proporcionar la respuesta en la operación.

ALIAS.STOREPERMANENTLY Indica la opción entre almacenar el alias de forma temporal (N) o indefinida (Y).

Cuando se realiza una transacción con este alias, los comerciantes deben volver a enviar este parámetro en el campo ALIASPERSISTEDAFTERUSE. De lo contrario, se almacenará de forma permanente.

### 3.4 Firma SHA para salida

Nuestro sistema devolverá una firma SHA-OUT, de [la misma forma que para las transacciones de e-Commerce](#), para los siguientes parámetros:

ALIAS.ALIASID

ALIAS.NCERROR

ALIAS.NCERRORCARDNO

ALIAS.NCERRORCN

ALIAS.NCERRORCVC

ALIAS.NCERRORED

ALIAS.ORDERID

ALIAS.STATUS

ALIAS.STOREPERMANENTLY

CARD.BIC

CARD.BIN

## FlexCheckout

CARD.BRAND

CARD.CARDHOLDERNAME

CARD.CARDNUMBER

CARD.CVC

CARD.EXPIRYDATE

## 4. Personalización

Este capítulo introduce dos tipos de personalización: básica y avanzada. En Personalización básica, aprenderá a aplicar su propio CSS y a personalizar los colores de texto, colores de fondo y de contenedores, y similares. La siguiente es una página de pago predeterminada que utiliza los archivos CSS predeterminados:

Card number

Cardholder's name

Expiry date

MM / YYYY

Card verification code

Submit Cancel

En la personalización Avanzada, aprenderá más acerca de las plantillas maestra/parciales, a utilizar marcadores de posición y a personalizar el contenido de texto con archivos XML. En función de sus necesidades, las plantillas parciales se crean y se personalizan por separado.

### 4.1 Personalización básica

Una personalización básica implica la configuración de hojas de estilo predeterminadas bajo la plantilla de Ingenico ePayments predeterminada

#### 4.1.1 Plantilla de Ingenico ePayments predeterminada

El archivo CSS puede ser un archivo personalizado con un estilo de Ingenico ePayments predeterminado o bien un archivo CSS cargado en el [Administrador de archivos](#) de plantillas.

Para obtener más información acerca de la personalización de su propio archivo CSS, consulte el uso de las [hojas de estilo](#).

Style sheet

Stylesheet filename

Default Ogone template

Default Ogone template

layer\_styling\_v5.css

SAVE

#### 4.1.2 Uso de hojas de estilo

Puede adaptar el diseño de FlexCheckout mediante la aplicación de un archivo CSS. Primero tendrá que cargar su archivo CSS en el Administrador de archivos de plantilla y vincularlo a su cuenta de Ingenico ePayments. El archivo CSS consta de contenedores de formulario / datos y elementos (como: Fecha de caducidad, CVC, Cuadro de información y etc.) que deben establecerse como el archivo CSS predeterminado. Puede vincular cada elemento que define el formato para cada campo al contenedor de formulario / datos.

En función de las necesidades empresariales, puede modificar la regla de formato dentro de cada elemento. Haga clic aquí para obtener una lista de [CSS elements](#).

A continuación se presenta un formulario/contenedor de datos y elementos de muestra:

```
/*-----FORM\DATA CONTAINER-----*/

#payment-container {width:100%; max-width:768px !important; margin:0 auto;background:#fff;}

#payment-form-container {width:280px !important; padding:0 10px; margin:0 auto;
background:#fff;overflow:hidden;}

#payment-data-container {width:260px !important; padding:20px 10px 5px 10px; margin:0 auto; color:#000;
background:#fff;overflow:hidden;}

#payment-data-container .payment-label { color:#000; font-size:12px; padding-bottom:3px;}

#payment-data-container .payment-input input { width:248px !important; height:35px; color:#000;
font-size:12px; margin-bottom:10px; padding:0 5px; -webkit-border-radius: 3px; -moz-border-radius: 3px;
border-radius: 3px; border:1px solid #000;}

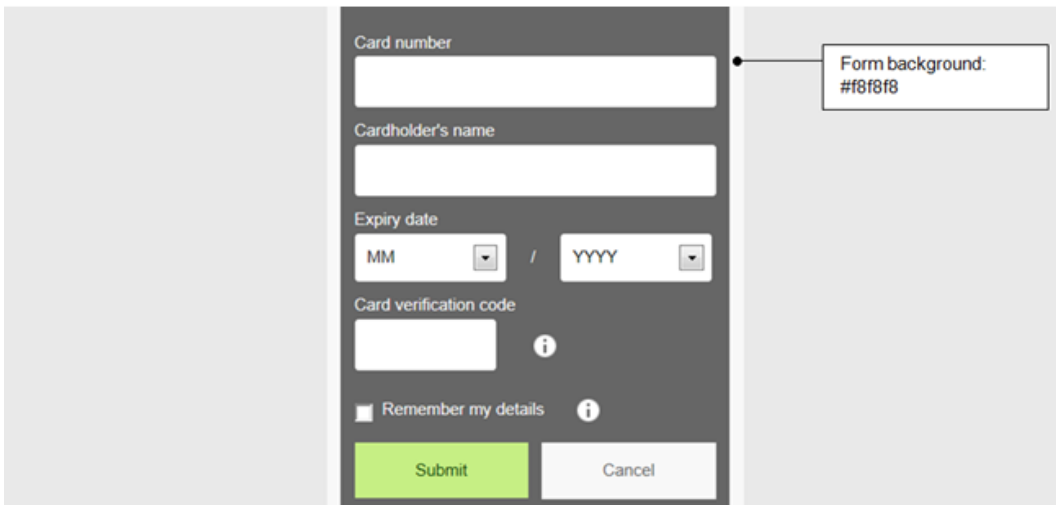
#payment-data-container .payment-input select { width:109px; height:35px; color:#000; font-size:12px; margin-
bottom:10px; padding:7px 5px; -webkit-border-radius: 3px;-moz-border-radius: 3px; border-radius: 3px;
border:1px solid #000;}

#payment-cardnumber-label-container,#payment-accountowner-label-container{ overflow: hidden;}

/*-----FORM\DATA CONTAINER-----*/
```

Puede cambiar la interfaz de usuario predeterminada (como se muestra a continuación) aplicando su propio CSS.

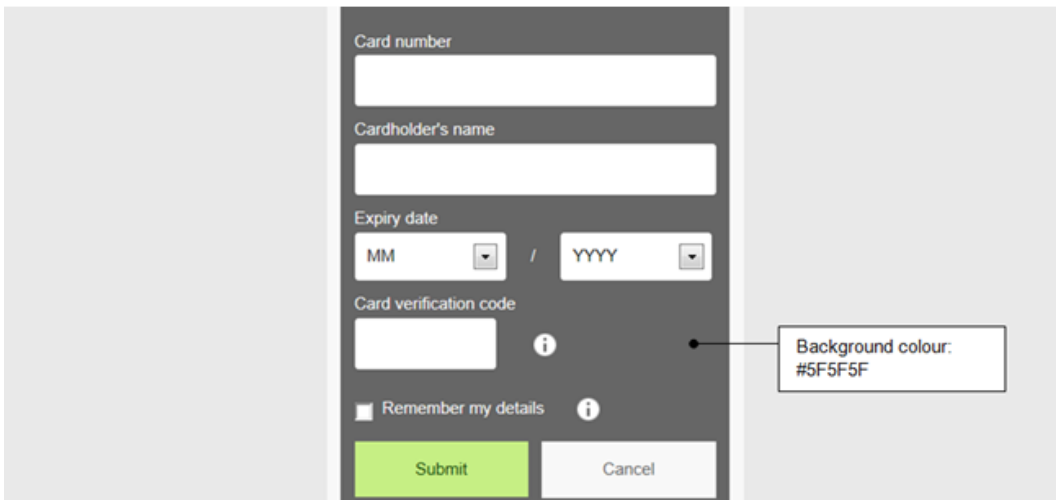
### Color de contenedor de formulario



CSS afectados:

```
.payment-form-container{ Background:#f8f8f8; }
```

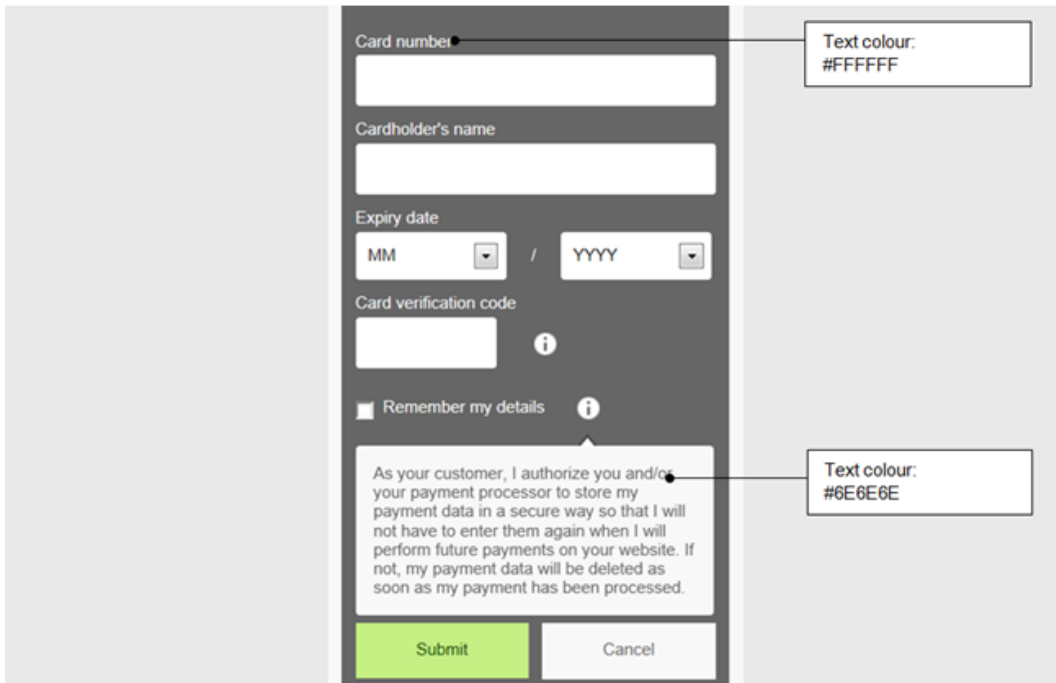
Color de contenedor de formulario



CSS afectados:

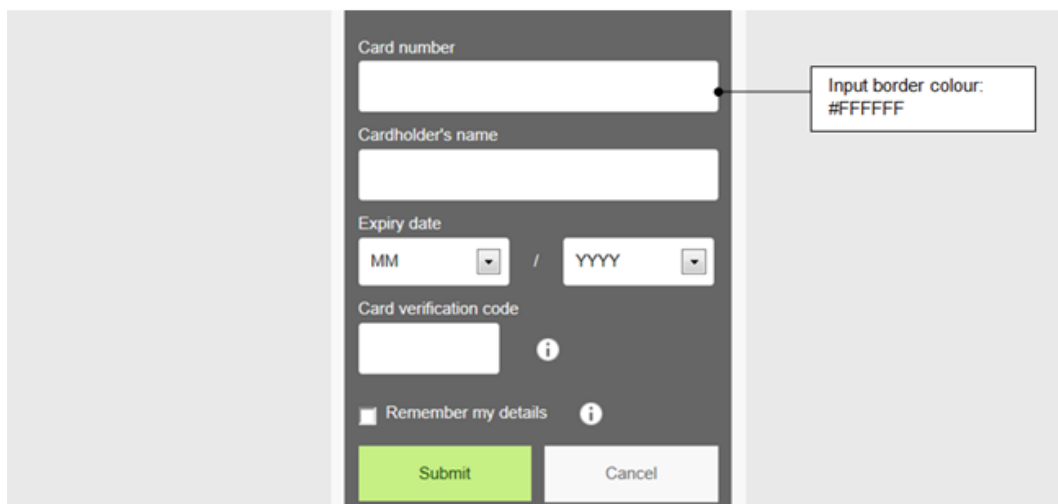
```
#payment-data-container{ Background:#5F5F5F; }
```

Color del texto



CSS afectados: #payment-data-container .payment-label{color:#FFFFFF;} #payment-data-container{color:#6E6E6E;}

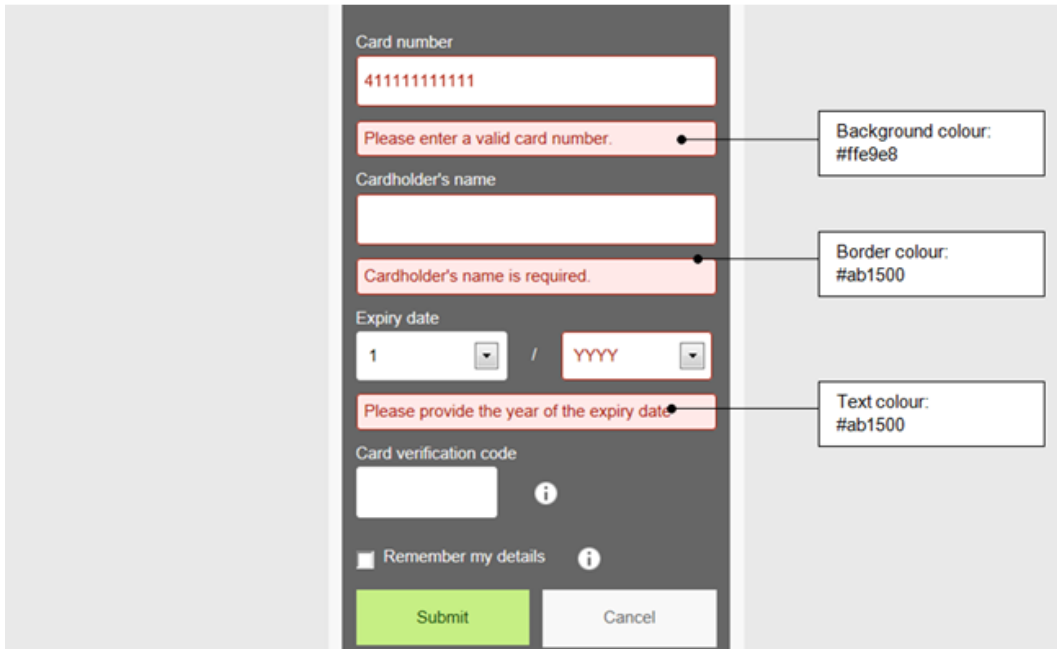
Color de borde de entrada



CSS afectados:

```
#payment-data-container .payment-input input{border:1px solid #FFFFFF;}
```

Cuadro de mensaje de validación

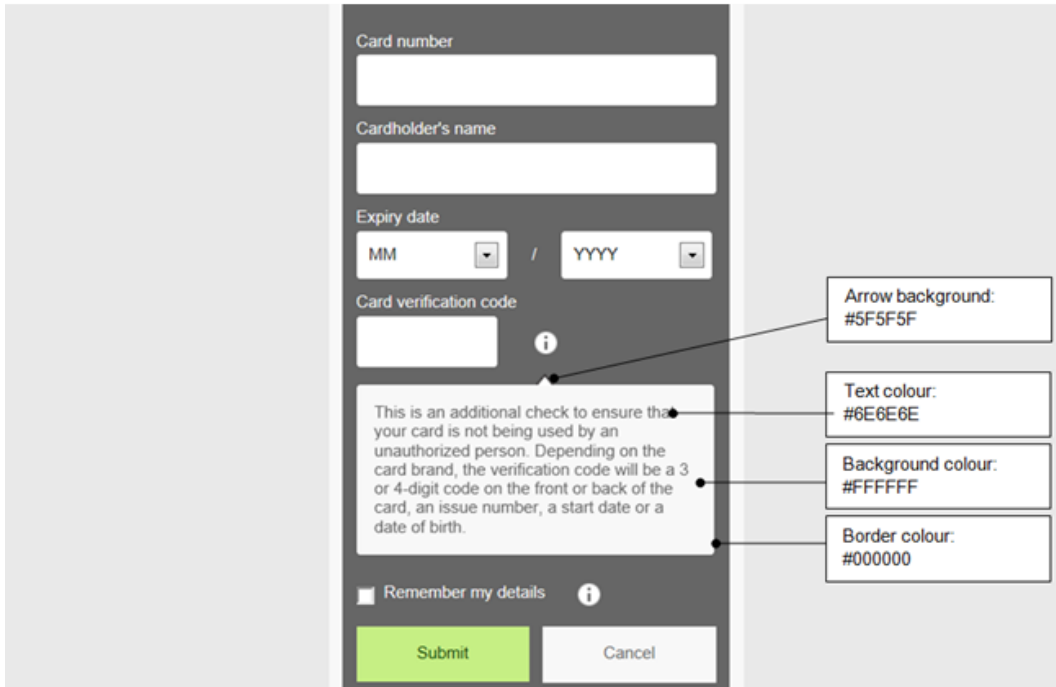


CSS afectados:

```
.payment-error { color:#ab1500; background:#ffe9e8; border:1px solid #ab1500;}  
#payment-data-container .payment-input-error input { color:#ab1500; border:1px solid #ab1500;}  
#payment-data-container .payment-input-error select { color:#ab1500; border:1px solid #ab1500;}
```

Cuadro de información de ayuda





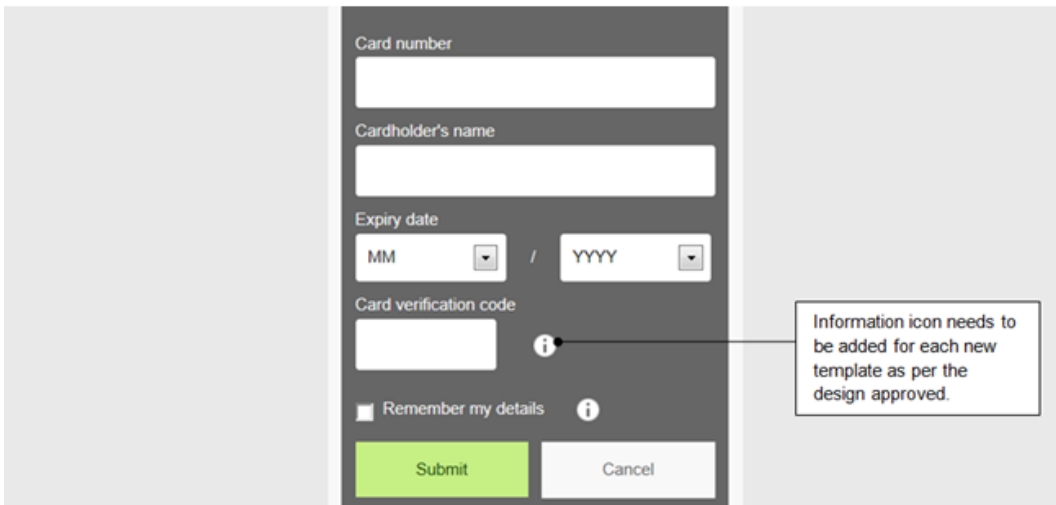
CSS afectados:

```
#payment-cvc-info-container span.arowlmg {background: #5F5F5F url(arrow-top.png) 0 0 no-repeat;}
```

```
#payment-cvc-info-container p.cvc-info-txt{color: #6E6E6E;}
```

```
#payment-cvc-info-container div.help-box {border: 1px solid #000000;background: #FFFFFF;}
```

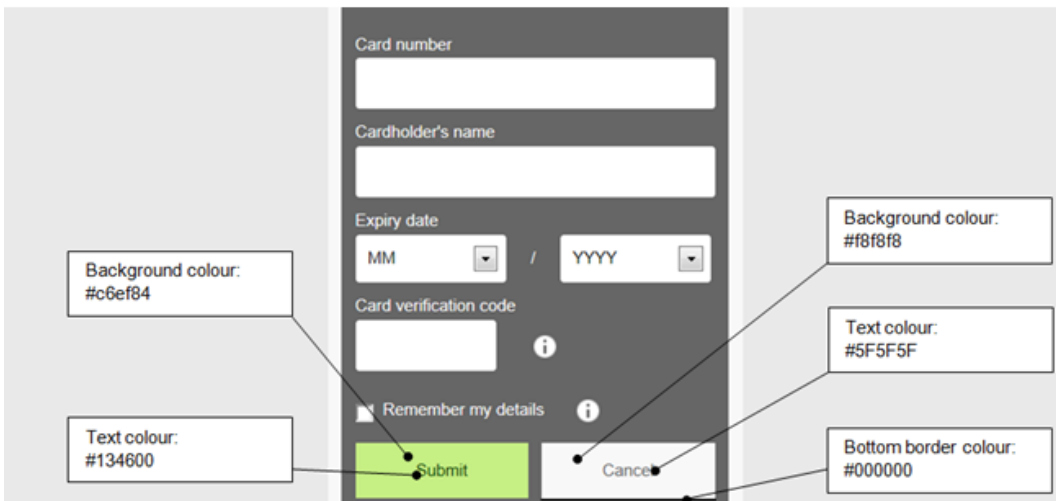
Information icon



CSS afectados:

```
payment-method-info {background:url(payment_info.png) 4px 1px no-repeat;}
```

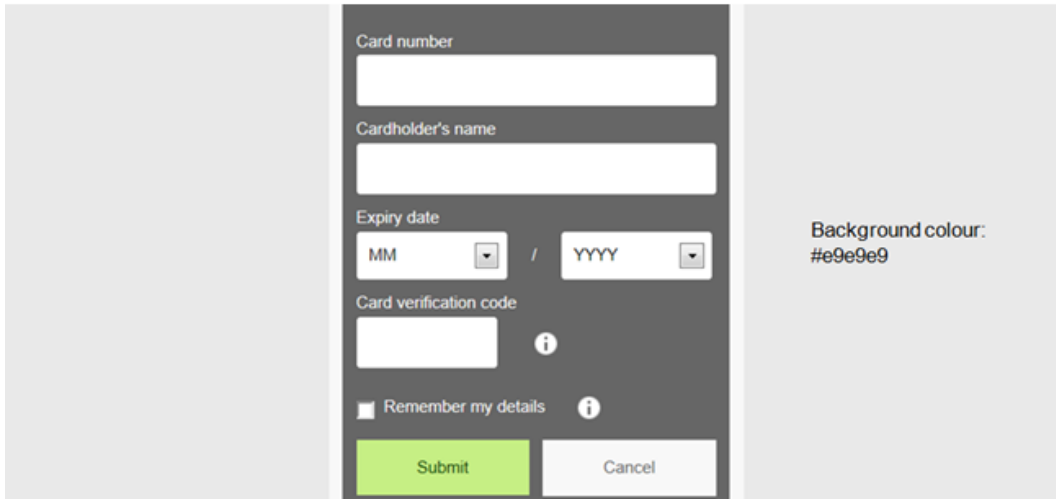
### Botón Enviar/Cancelar



CSS afectados:

```
#payment-data-container #payment-submit {background:#c6ef84;color: #134600;} #payment-cancel-container input {background:#f8f8f8;border-color:#000;}
```

### Color de fondo



CSS afectados:

```
payment-container {background:# e9e9e9} #
```

### CSS de ejemplo

```
#payment-container {background:# e9e9e9}

.payment-form-container{ Background:#f8f8f8; } #payment-data-container{ Background:#666; } #payment-data-container{color:fff;}
#payment-data-container .payment-label{color:fff;} #payment-data-container .payment-input input{border:1px solid #fff;}
.payment-error { color:#ab1500; background:#ffe9e8; border:1px solid #ab1500;}

#payment-data-container .payment-input-error input { color:#ab1500; border:1px solid #ab1500;} #payment-data-container .payment-
input-error select { color:#ab1500; border:1px solid #ab1500;} #payment-cvc-info-container div.help-box {border: 1px solid
#000;background: fff;} #payment-cvc-info-container span.arowwImg {background: #fff url(arrow-top.png) 0 0 no-repeat;} #payment-
cvc-info-container p.cvc-info-txt{color: #000;}
.payment-method-info {background:url(payment_info.png) 4px 1px no-repeat;}

#payment-data-container #payment-submit {background:#c6ef84;color: #134600;} #payment-cancel-container input
{background:#f8f8f8;border-color:#000;}
```

## 4.2 Personalización avanzada

Para permitir una mayor flexibilidad, puede crear y personalizar plantillas maestras y plantillas parciales. En función de sus necesidades, las plantillas parciales se crean y se personalizan por separado. También puede personalizar el contenido de texto con archivos XML.

Puede elegir aplicar una etiqueta `$$$PAYMENT ZONE$$$` que requiere una plantilla sencilla o personalizar su propia plantilla parcial [utilizando marcadores de posición](#).

### 4.2.1 Uso de marcadores de posición

Puedes personalizarlo y dar formato a su propia plantilla maestra (master.html) y plantillas parciales, pueden insertar marcadores de posición directamente en master.html.

También puede definir el aspecto de la página de pago a través de un archivo CSS personalizado (**newuser.css**) que debe estar enlazado a master.html. Si elige personalizar su archivo CSS, el archivo CSS modificado se debe cargar en el Administrador de archivos de plantillas y se debe definir como predeterminado.

Los marcadores de posición se clasifican en tres grupos: marcadores de posición principales, de elemento y agrupados.

#### **Marcadores de posición principales**

La siguiente lista muestra los marcadores de posición principales:

- **\$\$\$PAYMENT\_ZONE\$\$\$**: representa un método de pago o solicitud de marca bien a través de una plantilla personalizada (con una plantilla parcial disponible) o la plantilla predeterminada.
- **\$\$\$TP\_RESOURCES\_URL\$\$\$**: hace que la URL esté disponible en el directorio donde se encuentran los archivos en el Servicio de archivo alojado.

También tiene la opción de utilizar secuencias de comandos personalizadas que tendrán que estar enlazadas a un archivo de JavaScript personalizado. Se puede hacer referencia a este archivo de JavaScript una vez que estén cargados en el Administrador de archivos de plantillas. Por ejemplo, el archivo JavaScript al que se hace referencia en `<script type="text/javascript" src="$$$TP_RESOURCES_URL$$$/existinguser.js"></script>` es **existinguser.js**.

Para utilizar bibliotecas de JavaScript en la plantilla (p. ej., JQuery), debe cargar los archivos de JavaScript en el Administrador de archivos de plantillas. Una vez cargado, utilice **\$\$\$TP\_RESOURCE\_URL\$\$\$** para hacer referencia a las secuencias de comandos. Este paso se debe ejecutar para todos los archivos a los que se haga referencia en la plantilla: CSS, imágenes, fuentes, etc. y se asegurará de que la secuencia de comandos ejecutada sea el JavaScript dedicado.

**Nota:** No todos los datos se pueden insertar/extraer desde un host de terceros que no sea compatible con PCI, por tanto, el uso de JavaScript afectará a la capacidad del comerciante de reunir los requisitos de idoneidad para PCI SAQ A.

- **\$\$\$DISABLE\_MASKING\$\$\$**: deshabilita una máscara de cuadro de entrada.
- **\$\$\$SUBMIT\$\$\$**: representa un botón Enviar para el formulario. Por ejemplo:

```
<div id="payment-submit-container">

    <input id="payment-submit" type="submit" value="Submit" />

</div>
```

- **\$\$\$CANCEL\$\$\$**: representa un botón Cancelar para el formulario. Por ejemplo:

```
<div id="payment-cancel-container">

    <input id="payment-cancel" name="cancel" type="submit" value="Cancel" class="cancel" />

</div>
```

Una plantilla maestra que contenga **\$\$\$PAYMENT\_ZONE\$\$\$** no necesitará incluir los marcadores de posición enviar y cancelar. No obstante, se deben incluir los marcadores de posición enviar y cancelar si se construye una plantilla parcial por método de pago o si se crea una plantilla principal sin el marcador de posición **\$\$\$PAYMENT\_ZONE\$\$\$**.

#### **Marcador de posición de elemento**

El control <elemento> real se representa dentro de un contenedor de elemento: <div id="payment-<input\_name>-<elemento>-container" class="payment-<elemento>">. para una aplicación de estilos CSS sencilla, se aplica la misma convención de nombres a Id. y clase.

- \$\$\$<input\_name><elemento>\$\$\$ : The card number input control.
- \$\$\$CARD NUMBER.INPUT\$\$\$ - CARD NUMBER como el <nombre\_entrada> donde INPUT es el <elemento> html.
- \$\$\$<input\_name>LABEL\$\$\$ : proporciona una etiqueta para el elemento <nombre\_entrada>.

Por ejemplo, el marcador de posición \$\$\$CARD NUMBER LABEL\$\$\$ representa esto:

```
<div id="payment-cardnumber-label-container" class="payment-label">  
  
<label for="payment-cardnumber" id="lbl_CreditCardInputModel_CardNumber" title="CardNumber">Card number</label> </div>
```

Para todas las etiquetas se representará un div que actúe como contenedor:

```
<div id="payment-<input_name>-label-container" class="payment-label">
```

La clase de un contenedor de etiqueta siempre será "payment-label" de modo que el estilo de las etiquetas se pueda definir conjuntamente.

Una etiqueta para un control de etiqueta siempre se representará:

```
<label for="payment-<input_name>" id="lbl_CreditCardInputModel_CardNumber"  
  
title="CardNumber">Card number</label>
```

Para definir el estilo de un control de etiqueta, aplique el archivo CSS al ID del contenedor, no al ID del control.

\$\$\$<input\_name> INPUT\$\$\$ representa el control de entrada para:

```
<div id="payment-<input_name>-input-container" class="payment-input">  
  
<input Id="payment-<input_name>" autocapitalize="off" autocomplete="off"  
  
autocorrect="off" class="inp-txt" id="txt_CreditCardInputModel_CardNumber"  
  
maxlength="40" name="CreditCardInputModel.CardNumber" pattern="[X0-9]*"  
  
spellcheck="False" type="tel" value=""></input>  
  
</div>
```

Para definir el estilo de un control de entrada, aplique el archivo CSS al ID del contenedor, no al ID del control.

\$\$\$<input\_name> INFO\$\$\$ representa un enlace a una descripción del campo <input\_name> field:

```
<div id="payment-<input_name>-info-container" class="payment-info">

    <span id="payment-<input_name>-info"></span>

</div>
```

Enlaza cualquier información para este campo en el idioma seleccionado.

\$\$\$<input\_name> ERROR\$\$\$ representa un contenedor para mensajes de error desde <input\_name>:

```
<div id="payment-<input_name>-error-container">

<span class="field-validation-valid" data-valmsg-for="CreditCardInputModel.CardNumber"

data-valmsg-replace="true"></span>

</div>
```

Los mensajes de validación se pueden insertar aquí.

### Marcadores de posición agrupados

Los marcadores de posición de elemento se pueden insertar como grupo mediante marcadores de posición agrupados. El siguiente es un contenedor representado con un marcador de posición agrupado: <div id="payment-<input\_name>-container"></div>. En el cual insertaremos los marcadores de posición de elemento en este grupo. Efectivamente, representamos esto:

```
<div id="payment-<input_name>-container">

    $$$<INPUT_NAME> LABEL$$$

    $$$<INPUT_NAME> INPUT$$$

    $$$<INPUT_NAME> ERROR$$$

    $$$<INPUT_NAME> INFO$$$

</div>
```

Una plantilla maestra puede tener marcadores de posición dentro <div id="paymentForm">. Los marcadores de posición (tales como \$\$\$CARD NUMBER\$\$\$, \$\$\$CARDHOLDER NAME\$\$\$, \$\$\$EXPIRY DATE\$\$\$, etc.) definen el contenido del formulario de la tarjeta de crédito. Los comerciantes pueden seleccionar los marcadores de posición que mejor se adapten a sus necesidades.

A continuación se muestra un ejemplo de una plantilla maestra con marcadores de posición para un formulario de tarjeta de crédito etiquetado "paymentForm":

```
<!DOCTYPE html><html xmlns="http://www.w3.org/1999/xhtml">  
  
<head>  
  
  <link rel="stylesheet" href="$$$TP_RESOURCES_URL$$$newuser.css"/>  
  
</head>  
  
<body>  
  <div id="paymentForm">  
  
    $$$CARD_NUMBER$$$  
  
    $$$CARDHOLDER_NAME$$$  
  
    $$$EXPIRY_DATE$$$  
  
    $$$CVC$$$  
  
    $$$SUBMIT$$$  
  
    $$$CANCEL$$$  
  
  </div>  
  
</html>
```

Para crear un formulario de domiciliación bancaria, los comerciantes pueden utilizar marcadores de posición de domiciliación bancaria, como el siguiente:

```
$$$ACCOUNT_OWNER$$$  
$$$BANK_ACCOUNT$$$  
$$$BIC$$$
```

El nombre de cuenta bancaria, número IBAN y código BIC representarán un formulario de domiciliación bancaria.

Para los comerciantes que desean una plantilla que admita varios métodos de pago, pueden crear una plantilla maestra y una plantilla parcial para métodos de pago.

Una plantilla maestra es del siguiente modo:

```
<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head>
  <script src="$$$TP RESOURCES URL$$$/jquery-2.1.4.min.js"></script>
  <link rel="stylesheet" href="$$$TP RESOURCES URL$$$/user.css"/>
  <script type="text/javascript" src="$$$TP RESOURCES URL$$$/existinguser.js"></script>
</head>
<body>
  <div id="paymentForm">
    $$$PAYMENT ZONE$$$
  </div>
</body>
</html>
```

Una plantilla parcial es del modo siguiente y se crea con marcadores de posición (\$\$\$CARD NUMBER\$\$\$, \$\$\$CARDHOLDER NAME\$\$\$, \$\$\$EXPIRY DATE\$\$\$, and so on):

```
<div id="paymentForm">
$$$CARD NUMBER$$$
$$$CARDHOLDER NAME$$$
$$$EXPIRY DATE$$$
$$$CVC$$$
$$$SUBMIT$$$
$$$CANCEL$$$
</div>
```

Si los comerciantes no desean representar el formulario de pago predeterminado mediante \$\$\$PAYMENT ZONE\$\$\$, se puede insertar una plantilla parcial en su lugar.

Por ejemplo, si el método de pago solicitado es una tarjeta de crédito, se utilizará una plantilla parcial con el nombre **master.creditcard.htm**. Y si se trata de una domiciliación bancaria, en su lugar se utilizará una plantilla parcial con el nombre **master.directdebit.htm**.

Los comerciantes tendrán que cargar esta plantilla parcial en el Administrador de archivos. También se debe seguir estrictamente una convención de nomenclatura. Por ejemplo, si la plantilla maestra existente se denomina **master.html** y si los comerciantes desean enviar domiciliación bancaria como método de pago, será necesario cargar una plantilla parcial denominada **master\_directdebit.html** en el Administrador de archivos.

#### 4.2.2 Personalización de contenido de texto con XML

También puede personalizar el aspecto del texto en las páginas de tokenización mediante un archivo XML. Si no se aplica un archivo XML, se utilizará el formulario de tokenización. Después de personalizar el archivo XML, tendrá que cargarlo en el Administrador de archivos de plantillas y utilizar el mismo nombre de archivo que la plantilla maestra. Tenga en cuenta que como requisito, se debe cargar de antemano una plantilla maestra.

Con el archivo XML, debe haber también elementos para identificar los campos. Haga clic aquí para obtener una lista [completa lista de elementos](#). El siguiente es un ejemplo de un elemento (CardHolderNameInput) para la introducción de un nombre de titular de tarjeta:



```

<Element ElementId="CardHolderName" >

  <Languages>

    <Language Id="en_us">Card Holder Name only en</Language>

    <Language Id="fr">Card Holder Name only fr</Language>

  </Languages>

</Element>

```

En este elemento se han especificado los idiomas Inglés (EE. UU.) y Francés en el Language Id. La etiqueta ID también especifica el código de cultura ISO.

Si no se especifica ningún elemento, se utilizará en su lugar el texto predeterminado para dicho elemento. Y si no se ha especificado el idioma de un elemento, se utilizará el idioma predeterminado.

Nota:

La primera línea del archivo XML debe empezar siempre por <root>. La última línea del archivo XML debe terminar siempre por </root>

## 4.3 Administrador de archivos de plantillas

Con el Administrador de archivos de plantillas puede gestionar fácilmente sus plantillas y los distintos archivos que la acompañan.

Para empezar a usar el Administrador de archivos, inicie sesión en su cuenta de Ingenico ePayments y vaya a "Configuración" > "Plantilla" > "Administrador de archivos".

### 4.3.1 Subir archivos de plantilla



En "Subir archivos de plantilla", seleccione el botón "Archivos..." para seleccionar los archivos que desee cargar. Puede cargar css e imágenes

(.css, .jpg, .jpeg, .gif, .png), de un máximo de 7 MB por cada archivo, y un total de 10 MB.



Realice su selección y confírmela.

### 4.3.2 Comprobar y administrar archivos cargados

Cuando los archivos acaben de cargarse, los verá en la misma página, en el apartado "Archivos cargados".

Los archivos mostrarán primero el estado "Validando", durante el cual, se realizan las comprobaciones de seguridad y virus necesarias (puede tardar unos minutos).

Podrá utilizar los archivos una vez el estado cambie a "Validado".

Haga clic en el botón de actualización  para comprobar el estado actual de los archivos, o haga clic en el botón  para eliminar el archivo permanentemente.

For creating a template, you reference your resources with the following code: "\$\$TP RESOURCES URL\$\$/[your file name]".

Drag "File Type" or "Status" column header and drop it here to group by that column

File Name	File Type	Size	Status	Upload Date	Action
successful_top.png	Image	4KB	Validating 	2015-07-08 10:13:16	
4ec6d8b8-2af4-4279-bfdc-7fe40dfaf9e6_6.jpg	Image	2035KB	Validated 	2015-06-25 15:52:45	
layer_styling_v5.css	CSS	9KB	Validated 	2015-06-22 14:53:16	
cancel_top.png	Image	2KB	Validated 	2015-06-16 15:55:09	

Un archivo mostrará el estado "Rechazado" si no ha superado la comprobación de seguridad. Puede deberse a un virus o a que la extensión del archivo sea incorrecta, por ejemplo.

### 4.3.3 Integración

En su código, hace referencia a los archivos cargados mediante un código con la siguiente estructura: \$\$TP RESOURCES URL\$\$/[el nombre de su archivo].

## 5. Marcadores de posición

A continuación se muestra una lista completa de los marcadores de posición y su finalidad:

<b>Marcadores de posición principales</b>	Descripción
\$\$\$PAYMENT_ZONE\$\$\$	Una plantilla personalizada a partir de una plantilla parcial (si está disponible) o una plantilla predeterminada que representa la marca o el método de pago solicitados.
\$\$\$TP_RESOURCES_URL\$\$\$	Llama a un archivo que se ha cargado con anterioridad en el área de administración.
\$\$\$DISABLE_MASKING\$\$\$	Representa un cuadro de texto que coincide con los tipos de tarjeta y el formato de entrada.  Deshabilita la máscara del cuadro de entrada al detectar el tipo de tarjeta y el formato de cuadros de entrada en función de la marca de la tarjeta.
\$\$\$SUBMIT\$\$\$	Un botón Enviar que debe estar presente en el formulario.
\$\$\$CANCEL\$\$\$	Un botón Cancelar en el formulario.
\$\$\$CARD_NUMBER\$\$\$	Un número de tarjeta de crédito con validación y máscara adecuado para la marca.
\$\$\$CARD_NUMBER_LABEL\$\$\$	Una indicación del número de una tarjeta de crédito.
\$\$\$CARD_NUMBER_INPUT\$\$\$	El campo donde se introduce una tarjeta de crédito.
\$\$\$CARD_NUMBER_INFO\$\$\$	El campo donde se suministra la información de una tarjeta de crédito.
\$\$\$CARD_NUMBER_ERROR\$\$\$	Un mensaje de error que se muestra cuando se ha introducido incorrectamente un número de tarjeta erróneo.
\$\$\$CARDHOLDER_NAME\$\$\$	Un nombre de un titular de tarjeta.
\$\$\$CARDHOLDER_NAME_LABEL\$\$\$	Una indicación de un nombre de titular de tarjeta.
\$\$\$CARDHOLDER_NAME_INPUT\$\$\$	El campo para un nombre de titular de tarjeta que se va a introducir.
\$\$\$CARDHOLDER_NAME_INFO\$\$\$	El campo para la información de nombre de titular de tarjeta que se va a introducir.
\$\$\$CARDHOLDER_NAME_ERROR\$\$\$	Un mensaje de error que se muestra cuando se ha introducido incorrectamente un nombre de titular de tarjeta.

\$\$\$EXPIRY DATE\$\$\$	El área donde se muestra una fecha de caducidad.
\$\$\$EXPIRY DATE LABEL\$\$\$	Una indicación de una fecha de caducidad.
\$\$\$EXPIRY DATE INPUT\$\$\$	El área donde se introduce una fecha de caducidad.
\$\$\$EXPIRY DATE MONTH\$\$\$	El mes de la fecha de caducidad.
\$\$\$EXPIRY DATE YEAR\$\$\$	El año de la fecha de caducidad.
\$\$\$EXPIRY DATE INFO\$\$\$	El campo donde se suministra la información de la fecha de caducidad
\$\$\$EXPIRY DATE ERROR\$\$\$	Un mensaje de error que se muestra cuando se ha introducido incorrectamente una fecha de caducidad.
\$\$\$CVC\$\$\$	El campo donde se coloca un CVC con su validación adecuada.
\$\$\$CVC LABEL\$\$\$	Una indicación para CVC.
\$\$\$CVC INPUT\$\$\$	El campo donde se introduce una CVC.
\$\$\$CVC INFO\$\$\$	El campo para mostrar la información de CVC.
\$\$\$CVC ERROR\$\$\$	Un mensaje de error que se muestra cuando se ha introducido incorrectamente un CVC.
\$\$\$BANK ACCOUNT\$\$\$	Un número de cuenta para formularios de domiciliación bancaria.
\$\$\$ACCOUNT OWNER\$\$\$	El área donde se muestra un titular de cuenta.
\$\$\$ACCOUNT OWNER LABEL\$\$\$"	Una indicación para un titular de cuenta.
\$\$\$ACCOUNT OWNER INPUT\$\$\$	El campo donde se introduce un titular de cuenta.
\$\$\$STORE PERMANENTLY\$\$\$	Una casilla de verificación para que el comerciante se suscriba o no o almacene permanentemente el alias.
\$\$\$BIC\$\$\$	Una casilla BIC para alias de domiciliaciones bancarias.

## XML elements

<b>Alias form</b>	<ul style="list-style-type: none"> <li>• PageTitle</li> <li>• Submit</li> <li>• Cancel</li> <li>• RememberMyDetails</li> <li>• AliasAuthorizationInfo</li> </ul>
-------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>Credit cards</b>	<ul style="list-style-type: none"> <li>• PageTitle</li> <li>• CardNumber</li> <li>• CardNumberIsRequired</li> <li>• InvalidCardNumber</li> <li>• CardNumberTooLong</li> <li>• InvalidCardNumberCharacters</li> <li>• CardHolderName</li> <li>• CardHolderNameIsRequired</li> <li>• InvalidCardHolderName</li> <li>• CardHolderNameTooLong</li> <li>• ExpiryDate</li> <li>• InvalidExpiryDate</li> <li>• ExpiryMonthsRequired</li> <li>• InvalidExpiryMonth</li> <li>• InvalidExpiryMonthLength</li> <li>• InvalidExpiryMonthCharacters</li> <li>• ExpiryYearsRequired</li> <li>• InvalidExpiryYear</li> <li>• InvalidExpiryYearLength</li> <li>• InvalidExpiryYearCharacters</li> <li>• Cvc</li> <li>• CvcIsRequired</li> <li>• InvalidCvcNumber</li> <li>• CvcTooLong</li> <li>• CvcContainsNonNumericCharacters</li> </ul>
<b>CVC information</b>	<ul style="list-style-type: none"> <li>• VisaCvcInfo</li> <li>• EurocardCvcInfo</li> <li>• MasterCardCvcInfo</li> <li>• DinersClubCvcInfo</li> <li>• DankortCvcInfo</li> <li>• LaserCvcInfo</li> <li>• MaestroCvcInfo</li> <li>• JcbCvcInfo</li> <li>• AmericanExpressCvcInfo</li> <li>• AuroreCvcInfo</li> <li>• SurcoufCvcInfo</li> <li>• PrintempsCvcInfo</li> <li>• ClubMedCvcInfo</li> <li>• OkShoppingCvcInfo</li> <li>• MandarineCvcInfo</li> <li>• KangourouCvcInfo</li> <li>• GoSportCvcInfo</li> <li>• FinarefCvcInfo</li> <li>• AlsoliaCvcInfo</li> <li>• DefaultCvcInfo</li> </ul>
<b>Direct debits</b>	<ul style="list-style-type: none"> <li>• AccountOwnerIsRequired</li> <li>• AccountOwnerTooLong</li> <li>• InvalidAccountOwner</li> <li>• AccountIsRequired</li> <li>• AccountTooLong</li> <li>• InvalidAccount</li> <li>• BicsRequired</li> <li>• BicTooLong</li> </ul>

- InvalidBic
- GiroAccount
- BicName
- Iban
- BankAccount
- InvalidIbanCharacters
- InvalidIbanStructure
- InvalidCheckDigits
- InvalidIbanCountry
- InvalidIbanLength
- InvalidIbanNumber
- InvalidBankAccountNumber
- IbanOrBankAccount
- IncorrectBic
- BlzIsRequired