wirecard

Customizing Theme on Payment Page
Dynamic Custom CSS URL
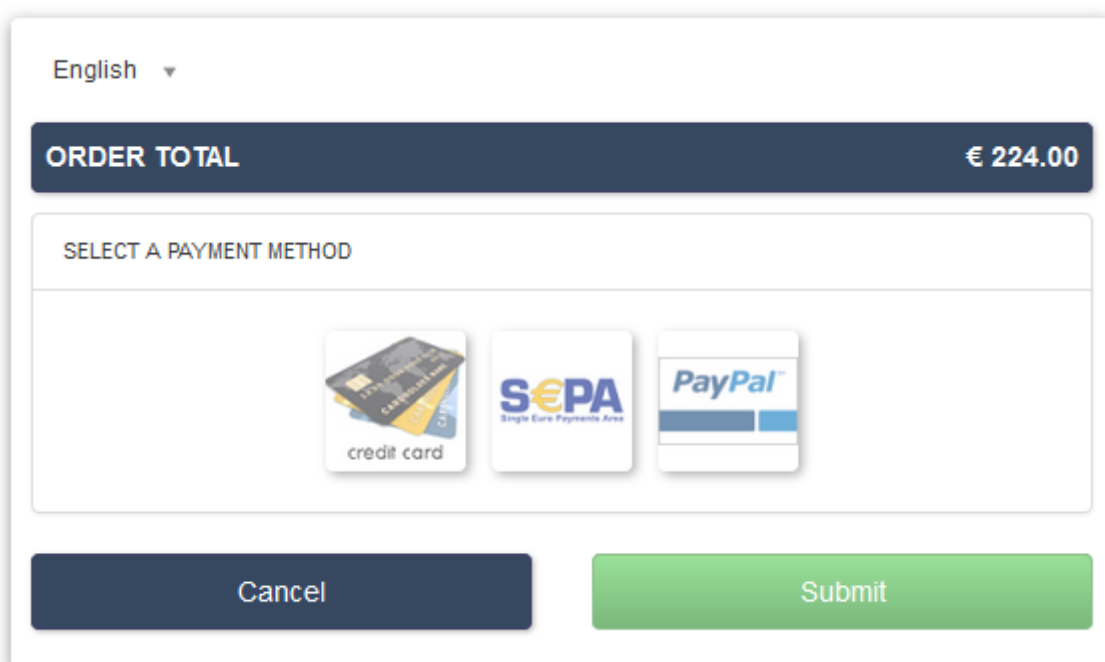
1. Payment page customization

1.1 Short description

CUSTOM CSS URL feature is providing option to send external css file link by every request. Because of this Feature, **Merchant is able to customize his own Payment Page** by editing existing tags styles from external css file. **Merchant is also able to unhide optional field** on Credit Card payment page. And there is an **option to set up custom logo** on Payment Process Page.

Advantage of this functionality is to send different types of CSS for different kind of requests. Second benefit is to customize your theme without asking us for permission. Your developer just need to hold a few rules and restrictions, which are not in fact any kind of obstacles to get your page designed into final deserved looking THEME.

Hosted and Embedded Payment Page is set up to read default css file **embeddedHpp.css.** This file handles basic default Color schemas. Developers must be obvious, because few of basic structures are included in bootstrap files.

Main <form> container is based on **BOOTSTRAP Grid systems,** which are used for creating page layouts through a series of rows and columns that house Payment Page content.
Developers are able to customize **bootstrap.min.css,** so during customizing theme, developer would need to observe bootstrap grid system usage rules. For more detail about bootstrap CSS conditions and usage please visit bootstrap page on http://getbootstrap.com/css/

Default Payment Page Design

1.2 Dynamic Custom CSS URL

Dynamic **Custom CSS URL** is now Hidden Mandatory Field named as " **custom_css_url** ", which is also part of the Signature. Dynamic CSS URL must meet the following criteria.

This functionality has been implemented to overlap default styling definitions ins **embeddedHpp.css.** and **bootstrap.min.css.** Custom css would appear on the bottom of css files in head tag. This fact guarantee casting existed CSS declaration.

1. Custom CSS URL must be shorter than 256 characters including non-alfa-numerical characters. In other cases developer would be notified by error message: Custom CSS URL is too long - more than 256 characters.

2. In general URIs as defined by RFC 3986 (see Section 2: Characters) may contain any of the following characters:
    ```
    ABCDEFGHIJKLMNOPQRSTUVWXYZ abcdefghijklmnopqrstuvwxyz 0123456789
    -._~:/?#[]@!$&'()*+,;=.
    ```
    In other cases developer would be notified by error message: Custom CSS URL has invalid format.

3. Custom CSS file must be smaller than 50 kb. In other cases developer would be notified by error message: Custom CSS is too large. Max. 50 kB

4. Custom URL need to be send without any mistakes, or more precisely URL needs to load. In other cases developer would be notified by error message:
    Custom CSS cannot be fetched

    Example of inappropriate usage.

    ```
    Httpp://www.your-site.com/custom-ctyle.css
    Http://www.your-site.com/custom-ctylecss {missing dot: custom-ctyle.css }
    Http://www.your-site.com/custom-ctyle.php { php is unsupported postfix }
    ```

5. Merchant needs to create a digital signature on his own server as it is the only place where the Secret is stored. The client's server also provides the Merchant Account ID to the mobile app client.

    To calculate the signature, following conditions need to be met:

    *   Fields need to be concatenated,
    *   leading and trailing space removed,
    *   SHA-256 signature follows the combined string

    Please note that the order of the fields is important. Also note, the same values used in the request signature must be placed into the client side form/data (with the exception of the secret key).

An order of the fields.

```
 1. request_time_stamp
 2. request_id
 3. merchant_account_id
 4. transaction_type
 5. requested_amount
 6. requested_amount_currency
 7. redirect_url (optional)
 8. custom_css_url (optional)
 9. ip_address (optional)
10. [secretkey]
```

An example of request signature generation is as follows.

```
request_time_stamp          = '20120430123012'

request_id                  = 'order-12345'

merchant_account_id          = 'b19fb056-d8da-449b-ac85-
cfbfd0558914'

transaction_type             = 'purchase'

requested_amount            = '1.01'

requested_amount_currency   = 'USD'

redirect_url                = 'https://test.com'

custom_css_url              = 'https://test.com/custom.css'

ip_address                  = '127.0.0.1'

secret_key                   = 'efabf47b-e43b-4785-873f-
1c5bc65b7cd2'


Pre SHA-256 string

20120430123012order-12345b19fb056-d8da-449b-ac85-
cfbfd0558914purchase1.01USDhttps://test.comhttps://test.com/custom.
css127.0.0.1efabf47b-e43b-4785-873f-1c5bc65b7cd2


SHA-256 signature

a186cd295f5b0da14aa158090ee8abfcc1ca22961f2c19ea659c4a8f5cbb4a03
```

1.3 Custom CSS file content restrictions

Content of custom CSS file must meet the following criteria. Few elements has to be prohibited due to security reasons.

1. **Custom CSS cannot contains property '-moz-binding'.**
 CSS, attributes starting with "-" describes a proprietary attribute that might not be supported by all browsers. "-moz-binding" is a attribute used to load another XML file to describe the presentation of the element selected. For example, the following code segment imports an external XML file to apply CSS properties. In case of usage, error message would appear:
Custom CSS contains not allowed property '-moz-binding'.

   Example of inappropriate usage.

   ```
   body {
     -moz-binding: url("external-file.xml#xss");
   }
   ```

2. **Custom CSS cannot contains property 'behavior'.**
Element 'behavior' is the IE's way of doing -moz-binding. In case of usage, error message would appear:
Custom CSS contains not allowed property 'behavior'.

   Example of inappropriate usage.

   ```
   body {
     behavior:url("script.htc");
   }
   ```

3. For the same security reasons, which is held for properties "-moz-binding" and "behavior", property 'expression' is not allowed to use. . In case of usage, error message would appear:
Custom CSS contains not allowed property value 'expression'.

4. Some CSS attributes support using "url"s in its declaration. For example, it is able to use a URL when setting the source of an image in CSS. Because of this reason it is prohibited to use 'url(\"javascript:. property. In case of usage, error message would appear:
Custom CSS contains not allowed property value 'url(\"javascript:.

5. In CSS, it is option to import another stylesheet into the current stylesheet by prepending the file you want to include in a "@import" declaration at the start of the file. For example, the following CSS includes another CSS file called "included.css". In case of usage, error message would appear:
Custom CSS contains not allowed @import directive.

wirecard

Example of inappropriate usage.

```
@import url("included.css");
---existing CSS statements
```

6. It is also unable to use unknown directives. In case of usage, error message would appear:
   Custom CSS has been evaluated as harmful because it contains unknown directive.

7. Custom CSS file can't contains unknown properties. In case of usage, error message would appear: Custom CSS cannot be parsed.

2. Input Fields for Credit Card

The following elements are mandatory/optional for sending a request for the payment method Credit Card. All of the Optional fields are hidden by default. There is an option to **unhide Optional fields** in Custom CSS by setting div ID parameter **display** value **to block.**

Example of usage optional email and phone fields for Credit Card Payment Method.

```
/* ****************** ENABLING OPTIONAL FIELDS - START  ****************** */

#hpp-creditcard-form-row-for-email-field {
    display : block;
}

#hpp-creditcard-form-row-for-phone-field {
    display : block;
}

/* ****************** ENABLING OPTIONAL FIELDS - END  ****************** */
```

2.1. Description of fields

Term	Man/ Opt	Type	Value	Div tag ID
First Name	O	Input	String	„ Explicitly shown "
Last Name	M	Input	String	„ Explicitly shown "
Card Type	M	Select	String	„ Explicitly shown "
Card Number	M	Input	Numerical	„ Explicitly shown "
CVV	O	Input	Numerical	„ Explicitly shown "
Expiry Date - Month	M	Select	Numerical	„ Explicitly shown "
Expiry Date - Year	M	Select	Numerical	„ Explicitly shown "
Address (1)	O	Input	String	#hpp-creditcard-form-row-for-street1-field
Address (2)	O	Input	String	#hpp-creditcard-form-row-for-street2-field
City	O	Input	String	#hpp-creditcard-form-row-for-city-field
State/Province Postal Code/Zip	O	Input	String	#hpp-creditcard-form-row-for-state-and-postalcode-fields
Country	O	Select	String	#hpp-creditcard-form-row-for-country-field
E-mail	O	Input	Valid email address	#hpp-creditcard-form-row-for-email-field
Phone	O	Input	Valid phone number	#hpp-creditcard-form-row-for-phone-field

Example of showing all optional fields.

CREDIT CARD

First Name	(optional)	Last Name
John ✔		Doe ✔

Card Type

VISA Visa ▼ ✔

Card Number	CVV	(optional)
Card Number	CVV	⑦

Expiry Date   Month ▼   Year ▼

Address (1)   (optional)

Mullerstrasse 137 ✔

Address (2)   (optional)

Address (2)

City   (optional)

Berlin ✔

State/Province	(optional)	Postal Code/Zip	(optional)
Berlin ✔		13353 ✔	

Country   (optional)

Germany ✖ ✔

E-mail   (optional)

john@doe.com ✔

Phone   (optional)

+421123456789 ✔

Dynamic CSS URL

wirecard
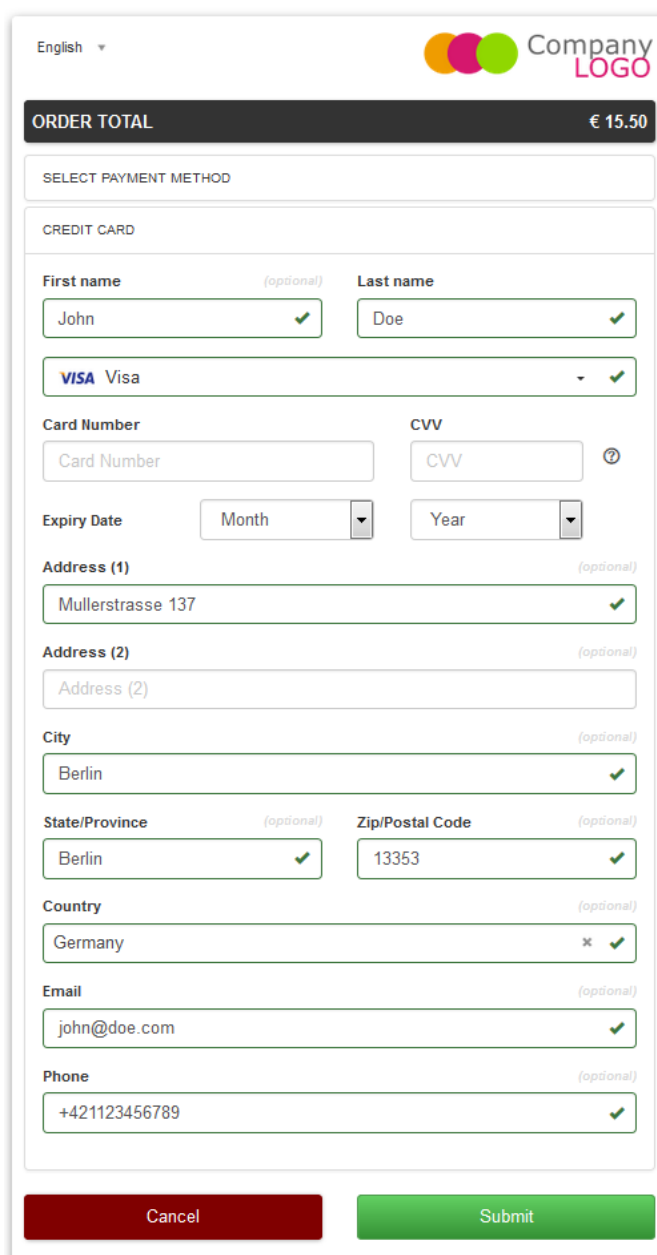
3. CSS samples of customized theme

For the purpose to start you own customized page, we prepared for you few examples, how some changed can be done. In order to get default theme and start customizing your own theme you just need to delete content of custom CSS files, which are described below.

3.1. Example of custom CSS usage

With few lines of CSS you are able to change colors, transitions, backgrounds..etc There is an option to add your own logo eventually picture with custom location of the file.

As you can see on picture bellow there is an option to change bootstrap css properties like in this case bootstrap .checkbox definition has been overlapped by this definition.

```css
/* ****************** CUSTOMIZING SEPA PAYMENT PAGE - START
****************** */
/* Highlighting SEPA payment consent checkbox */
 #sepaDirectDebitForm.form-horizontal .radio, #sepaDirectDebitForm.form-
horizontal .checkbox {
    min-height: 27px;
    background: #EDEDED;
    padding: 8px;
}
/* ****************** ENABLING OPTION FIELDS - END   ****************** */
```

3.2. Content of custom css file

There is prepared offline version of demo page with few comments, how to customize few different tags, to get your deserved theme.

This example is located inside **EE-dynamic-CSS-URL.zip\EE-dynamic-CSS-URL\engine\custom\my-custom-name-css-embeddedHpp.css**

If you're not familiar with few of used properties, please visit http://www.w3schools.com/css/ for more information.

```css
/* ****************** MAIN ELEMENTS THEMING - START  ****************** */
#hpp-logo {
/* customizing logo, this field could be hidden */
        height: 45px;
        width: 200px;
        float: right;
        background-image: url(your-company-logo.png) !important;
        background-repeat: no-repeat;
        background-position: right top;
}

.hpp-template, .hpp-nav > ul.nav > li.active > a {
        background: #5FD923 !important;
        color: white !important;
}

.hpp-template {
        background:-webkit-linear-gradient(#333333,#333333)!important;
        background:linear-gradient(#333333,#333333)!important;
        border: 1px solid #333333 !important;
}

.hpp-template-hover:hover, .hpp-template-focus:focus {
        background:#00D469!important;
        background:-webkit-linear-gradient(#00D168, #00964A)!important;
    background:linear-gradient(#00D168, #00964A)!important;
    box-shadow:0px 0px 11px #00994C!important;
}
#hpp-form-cancel{
background:maroon !important;
border: 1px solid maroon !important;
}
#hpp-form-cancel:hover, #hpp-form-cancel:focus {
   background:red important;
        background:-webkit-linear-gradient(red, red)!important;
    background:linear-gradient(red, red)!important;
    box-shadow:0px 0px 11px red !important;
}

.hpp-datepicker.datepicker-days td.active {
        background-color: #00A754 !important;
}

.hpp-nav > ul.nav > li.active > a::after {
        border-left-color:#00A754!important;
}
```

```css
.hpp-loading-spinner {
    background: url('loading.gif') no-repeat;
}

.hpp-container .panel-heading {
    background: transparent none repeat scroll 0% 0%;
    -webkit-transition: background-color 0.4s;
    -moz-transition: background-color 0.4s;
    -ms-transition: background-color 0.4s;
    -o-transition: background-color 0.4s;
    transition: background-color 0.4s;
    cursor: pointer;
    cursor: hand;
}
.hpp-container .panel-heading:hover {
    background-color: #DBDBDB;
    -webkit-transition: background-color 0.4s;
    -moz-transition: background-color 0.4s;
    -ms-transition: background-color 0.4s;
    -o-transition: background-color 0.4s;
    transition: background-color 0.4s;
    cursor: pointer;
    cursor: hand;
}
/* ****************** MAIN ELEMENTS THEMING - END  ****************** */

/* ****************** ENABLING OPTIONAL FIELDS - START  ****************** */
#hpp-creditcard-form-row-for-street1-field {
    display : block;
}

#hpp-creditcard-form-row-for-street2-field {
    display : block;
}

#hpp-creditcard-form-row-for-city-field {
    display : block;
}

#hpp-creditcard-form-row-for-state-and-postalcode-fields {
    display : block;
}

#hpp-creditcard-form-row-for-country-field {
    display : block;
}

#hpp-creditcard-form-row-for-email-field {
    display : block;
}

#hpp-creditcard-form-row-for-phone-field {
    display : block;
}
/* ****************** ENABLING OPTIONAL FIELDS - END  ****************** */



/* ****************** CUSTOMIZING CREDIT CARD PAYMENT PAGE - START  ******************
*/
/* Setting Optional text to italic and changing color to silver */
form#hpp-creditcard-form small[data-i18n~="optional"] {
    color: #D9D9D9;
    font-style: italic;
    font-weight: 900;
```

```
}
/* ****************** ENABLING OPTION FIELDS - END  ****************** */



/* ***************** CUSTOMIZING SEPA PAYMENT PAGE - START  ****************** */
/* Highlighting SEPA payment consent checkbox */
 #sepaDirectDebitForm.form-horizontal .radio, #sepaDirectDebitForm.form-horizontal
.checkbox {
    min-height: 27px;
    background: #EDEDED;
    padding: 8px;
}
/* ****************** ENABLING OPTION FIELDS - END  ****************** */
```

3.3. **FLAT DESIGN** THEME

Flat design Theme has been prepared for mobile users. As you can see developers are able to change completely design look and structure of payment page.
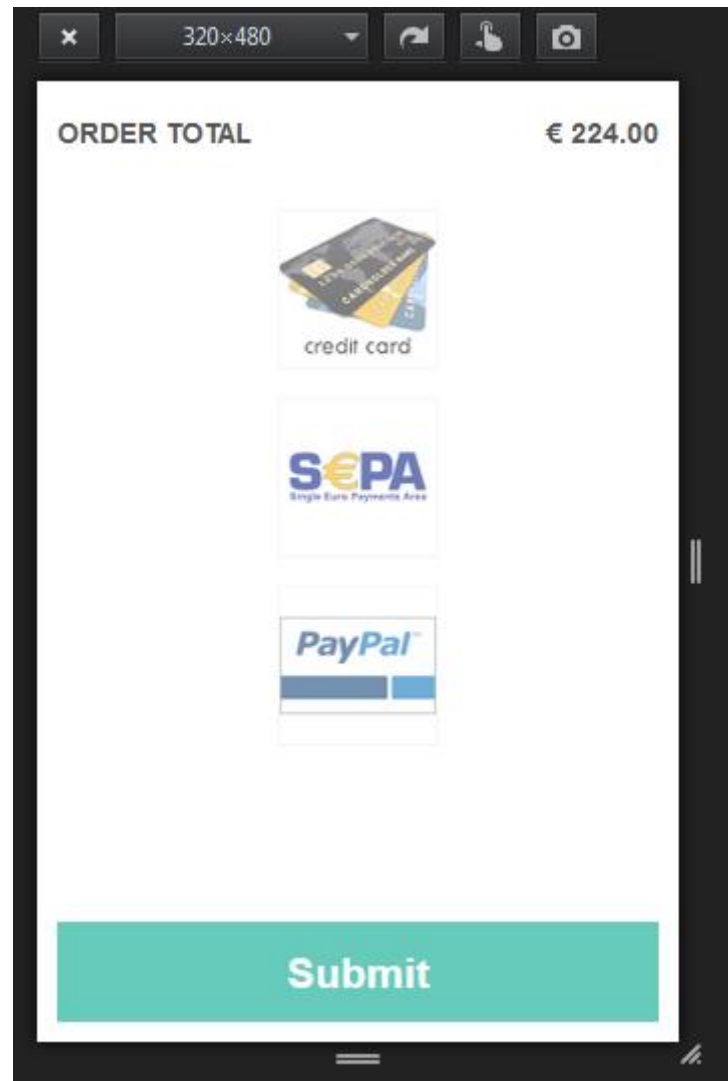
Example of customized Credit Card theme.

Example of customized SEPA Payment page.

Merchants are also allowed to change design of confirm dialog box.

Example of customized SEPA Payment page.

3.4.   FLAT DESIGN CSS file content

This example is located inside ***EE-dynamic-CSS-URL-FLAT-DESIGN.zip\EE-FLAT__DEMO\index_files\FLAT-DESIGN-CUSTOM-embeddedHpp.css***

There is no description inside flat design css file, but if you're not familiar with few of used properties, please visit http://www.w3schools.com/css/ for more information.

```css
html, body, .hpp-container {
       height:99%;
}

*:not(.caret) {
       box-shadow:none!important;
       border:none!important;
       border-radius: 0!important;
}

.form-control {
       background: rgb(241,241,241) !important;
       color:rgb(85,85,85)!important;
}

.form-group input {
       background: rgb(241,241,241) !important;
       color:rgb(85,85,85)!important;
}

.panel-heading {
       padding:0;
       margin:20px 0 30px;
       display:none; /* JS managed visibility */
}

.panel, .panel-group, .panel-body{
       padding:0!important;
       margin:0!important;
}

.hidden-xs, .hidden-sm {
       display:none;
}

.hpp-template {
       background: #072C4A!important;
       color:#fff!important;
       font-size:14px!important;
}

.hpp-template-hover:hover, .hpp-template-focus:focus {
       background: #103757!important;
    box-shadow: none!important;
}

#hpp-header {
       display:none;
```

```css
}

.hpp-btn-success {
        background: #14AF96!important;
        color:#fff!important;
        font-size:14px;
}

.hpp-btn-success:focus, .hpp-btn-success:hover, .hpp-template-bg .panel-heading a:hover
{
        background:#16B89E!important;
}

.hpp-template-bg .panel-heading a {
        padding:13px 15px;
        background: #14AF96!important;
        color:white!important;
}

.hpp-container {
        width:100%;
        padding:0!important;
        margin:0 auto!important;
}

.hpp-container > div {
        padding-left:10px!important;
        padding-right:10px!important;
}

.hpp-btn {
        padding:10px 15px!important;
        font-size:21px!important;
        font-weight: bold;
}

.hpp-grid-selection {
        width: 175px;
}

.hpp-grid-selection > button > img {
        margin:5px;
}

.dropdown-menu {
        z-index:1002;
}

.hpp-grid-selection img,
.hpp-select2-lang-drop, .hpp-select2-drop,
.dropdown-menu {
        border:1px solid #f1f1f1!important;
}

.hpp-select2-lang-container {
        background: rgb(241,241,241) !important;
        color:rgb(85,85,85)!important;
}

.hpp-row-card-num > div:first-child {
        float:left!important;
        width:65%!important;
        padding-right:5px!important;
}
```

```css
.hpp-row-card-num > div:nth-child(2) {
        float:right!important;
        width:35%!important;
        padding-left:5px!important;
}

.hpp-row-exp-date > div:first-child {
        display:none;
}

.hpp-row-exp-date > div:nth-child(2) {
        width:50%;
        padding-right:5px!important;
}

.hpp-row-exp-date > div:nth-child(3) {
        width:50%;
        padding-left:5px!important;
}

.hpp-white-glass {
        position:fixed;
        top:0;
        right:0;
        left:0;
}

#hpp-payment-methods > .hpp-white-glass > .hpp-loading-spinner {
        display:none!important;
}

#hpp-header {
        padding:0!important;
}

.hpp-form-title {
        background: #fff !important;
        color:rgb(85,85,85)!important;
}

.hpp-modal-confirm-dialog .hpp-form-title {
        text-align:center;
        margin:0;
}

#hpp-form-buttons {
        position:fixed;
        bottom:0;
        width:550px;
        margin:0!important;
        padding-top:5px;
        background:#fff;
        z-index: 1001;
}

#hpp-form-buttons > div {
        padding:0!important;
}

#hpp-confirm-dialog-buttons > .hpp-btn {
        width:50%;
}

#hpp-form-buttons > div:first-child {
        float:right!important;
```

```css
        left:0;
        width:100%;
}

#hpp-form-buttons > div:nth-child(2)  {
        display:none;
}

#hpp-form-navigation {
        padding:0!important;
}

#hpp-form-navigation > div {
        border-spacing:0;
}

#hpp-payment-methods {
        margin-top:15px;
        position:relative;
        padding-bottom:70px!important;
}

#hpp-form-previous {
        border-left: 2px solid #fff !important;
}

form div[class^="col"] > label, form div[class^="col"] > label + small {
        display:none;
}

@media screen and (max-width: 560px), screen and (max-height: 560px),
   screen and (max-device-width: 560px), screen and (max-device-height: 560px) {
        #hpp-form-buttons {
                width:100%;
        }
}
```