**int\_wirecard**

***Version <19.1.0>***



Inhaltsverzeichnis

[1. Summary 3](#_Toc5697545)

[2. Component Overview 3](#_Toc5697546)

[2.1 Functional Overview 3](#_Toc5697547)

[2.2 Limitations 3](#_Toc5697548)

[2.3 Compatibility 3](#_Toc5697549)

[3 Implementation guide 3](#_Toc5697550)

[3.1 Setup 3](#_Toc5697551)

[3.2 Configuration 4](#_Toc5697552)

[3.2.1 Add cartridges to site(s) 4](#_Toc5697553)

[3.2.2 Import metadata 4](#_Toc5697554)

[3.3 Custom code 4](#_Toc5697555)

[3.3.1 forms/default/billing.xml 4](#_Toc5697556)

[3.3.2 template/default/checkout/billing/paymentmethods.isml 4](#_Toc5697557)

[3.3.3 controllers/COBilling.js 5](#_Toc5697558)

[3.3.4 Controllers/COPlaceOrder.js 6](#_Toc5697559)

[3.3.5 7](#_Toc5697560)

[4 Maintenance 7](#_Toc5697561)

[4.1 Custom site preferences 7](#_Toc5697562)

[5 Support 7](#_Toc5697563)

[6 Release History 8](#_Toc5697564)

# Summary

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

# Component Overview

## Functional Overview

## Limitations

## Compatibility

# Implementation guide

## Setup

For the integration of wirecard in a sitegenesis storefront you have to import the cartridges int\_wirecard and int\_wirecard\_controllers.

In order to deploy the cartridges, import the cartridges into UX Studio by following these steps:

1. In UX Studio select from the menu File 🡪 Import and choose General 🡪 Existing Projects into Workspace then click next.
2. In the next window make sure the checkbox Select archive file: is selected and click on the Browse button on the right.
3. Select the archive file containing the integration and click Open then Finish.
4. If you have an active commerce cloud server connection the Studio will ask you to link the cartridge to it. Click on Yes.
5. To manually assign and deploy the cartridge to a server connection right click on it and click on Properties. Then select Project References and check the project containing the integration. The cartridge should now be automatically uploaded.

## Configuration

### Add cartridges to site(s)

To activate the integration in SiteGenesis, the cartridge has to be assigned to the storefront site in the Business Manager. In order to do this, follow these steps:

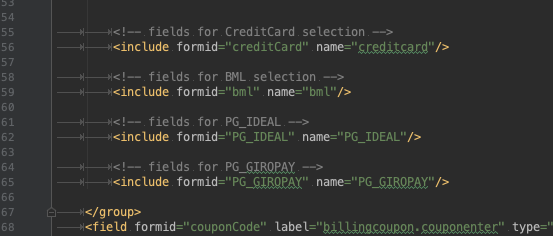
1. Login to Business Manager and click on Administration.
2. Click on Sites and then Manage Sites.
3. Click on the site name and then on the top tab menu on Settings.
4. In the text field Cartridges append this text “:**int\_wirecard:int\_wirecard\_controllers**”.
5. Repeat steps 3 to 4 for each Instance Type by selecting the appropriate type from the drop down menu (Production, Staging, Sandbox / Development).

### Import metadata

## Custom code

### forms/default/billing.xml

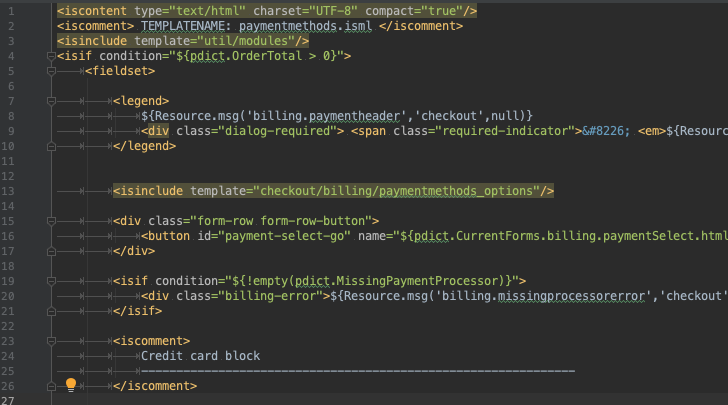
Append the wirecard payment forms by adding the respective form includes to the group paymentMethods:



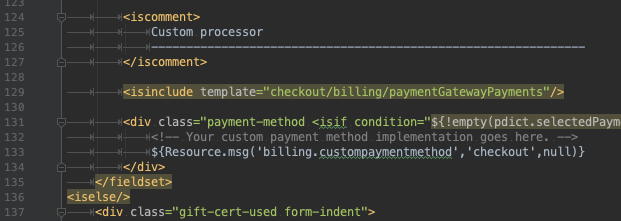
### template/default/checkout/billing/paymentmethods.isml

For having the wirecard payment options available in the billing step, you will have to replace the complete <div class=”payment-method-options form-indent”>..</div> with just one line: <isinclude template=”checkout/billing/paymentmethods\_options”/>

The result should look like this:



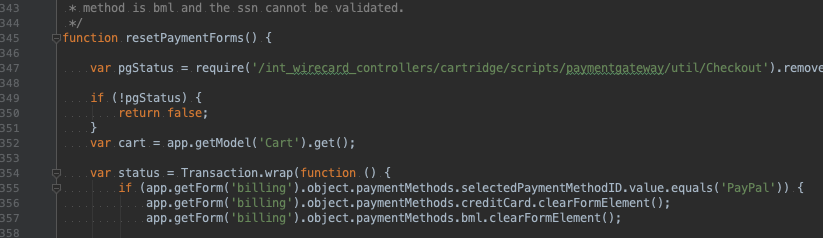
At the end of the same file you will have to include another template:



<isinclude template=”checkout/billing/paymentGatewayPayments”/>

### controllers/COBilling.js

Within the method “resetPaymentForms” append the following at the beginning:

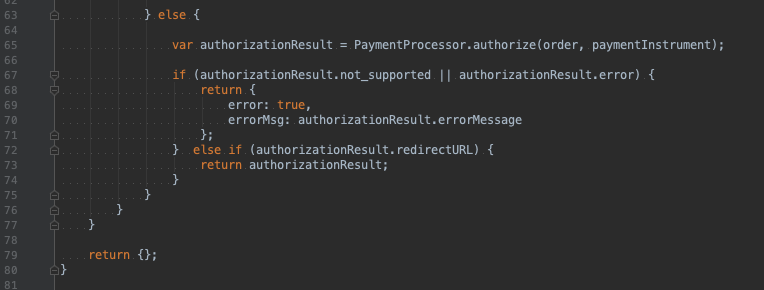


*var pgStatus = require('/int\_wirecard\_controllers/cartridge/scripts/paymentgateway/util/Checkout').removePaymentInstrumentsFromBasket();  
  
if (!pgStatus) {  
 return false;  
}*

This will remove the payment gateway payment methods from the cart when initializing the billing step.

### Controllers/COPlaceOrder.js

Within the method **handlePayments** append the following:



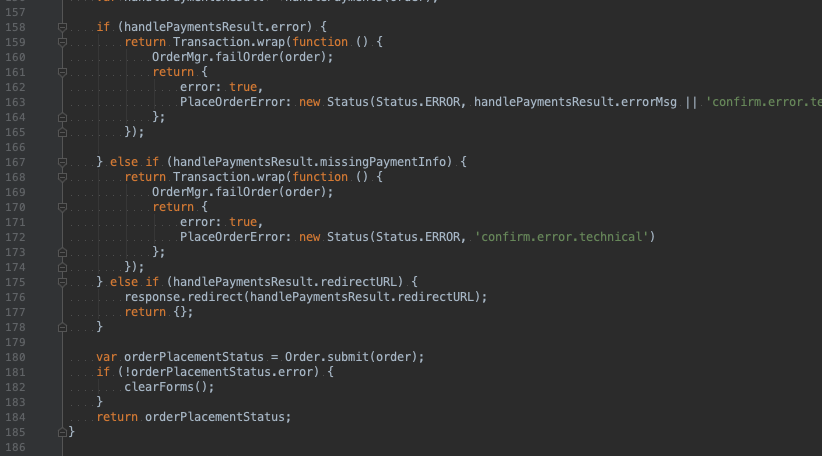
In the case when the initial request to the wpg triggers a redirect to a payment page, the authorization result object will contain a url which will be used in the frontend as redirect target for the customer.

*} else if (authorizationResult.redirectURL) {  
 return authorizationResult;  
}*

In the method start you will have to make the following changes in the control structure if (handlePaymentsResult.error):

*if (handlePaymentsResult.error) {  
 return Transaction.wrap(function () {  
 OrderMgr.failOrder(order);  
 return {  
 error: true,  
 PlaceOrderError: new Status(Status.ERROR, handlePaymentsResult.errorMsg || 'confirm.error.technical')  
 };  
 });  
  
} else if (handlePaymentsResult.missingPaymentInfo) {  
 return Transaction.wrap(function () {  
 OrderMgr.failOrder(order);  
 return {  
 error: true,  
 PlaceOrderError: new Status(Status.ERROR, 'confirm.error.technical')  
 };  
 });  
} else if (handlePaymentsResult.redirectURL) {  
 response.redirect(handlePaymentsResult.redirectURL);  
 return {};  
}*

The result should look like this:



### 

# Maintenance

## Custom site preferences

# Support

# Release History

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Changes** |
|  |  |  |