

Installation Manual for PAYONE WooCommerce

This manual describes the installation and usage of the PAYONE extension for WooCommerce.

Release Date: Mon, 07 Aug 2023 14:01:06 +0200

Version: 4.0.86

wallee AG
General-Guisan-Strasse 47
CH-8400 Winterthur

E-Mail: info@sellxed.com

Telefon:

CH: +41 (0)44 505 13 60

DE: +49 (0)40 2999 60117

UK: +44 (0)161 850 6890

US: +1 (0)205 557 5869

© copyright by wallee AG Mon, 07 Aug 2023 14:01:06 +0200

####conditional####

Table of Contents

1	Introduction	5
2	PAYONE Channels	6
2.1	Installation Process	6
2.2	System Requirements	7
3	PAYONE - Backend Configuration	8
3.1	Payment Portal Setup	8
3.1.1	Shop Portal Administration: Tab General	8
3.1.2	Shop Portal Administration: Tab Extended	8
3.1.3	Shop Portal Administration: API-Parameter	9
3.2	Authorization Methods	10
3.3	Account Activation	10
4	Module Installation and Update in the WooCommerce Shop	11
4.1	Installation	11
4.2	Updates and Upgrades	11
4.2.1	Update Checklist	11
4.2.2	Update Instructions	12
5	Module Configuration in the WooCommerce Shop	13
5.1	Configuration of the Main Module	13
5.2	Defining the URL for the Transaction Feedback	13
5.7	Redirection / Payment Form Error	15
5.4	Direct Capturing of Transactions	13
5.5	Uncertain Status	14
5.5.1	Setting the order state	14
5.6	Optional: Validation	14
6	Settings / Configuration of Payment Methods	16
6.1	General Information About the Payment Methods	16
6.2	Information on Payment Status	16
6.2.1	Order status "pending" / imminent payment (or similar)	16
6.2.2	Order status "cancelled"	16
7	The Module in Action	17
7.1	Useful Transaction Information on the Order	17

7.2	Usage of the Alias Managers / Token Solution	17
7.3	Capturing of Orders	18
7.4	Refunds	20
7.5	Transaction Process	21
7.6	Using Invoice Details of a Processor	21
	7.6.1 WooCommerce Order Confirmation (E-Mail)	22
	7.6.2 WooCommerce Invoice (PDF)	22
	7.6.3 WooCommerce-Backend (Transaction details)	22
	7.6.4 WooCommerce Success-Page	22
7.7	Setup a Cron Job to Activate the Timed Operations	23
8	Testing	25
	8.1 Test Data	25
9	Errors and their Solutions	30
	9.1 Page Not Found Error prior to the redirection to Payment Page	30
	9.2 The Referrer URL appears in my Analytics Tool	30
10	Compatibility with Third-Party Plugins	31
	10.1 Fee's and discount's within WooCommerce	31
	10.2 Multilingual WooCommerce sites	31
	10.3 Birthday and gender in WooCommerce	31
11	Error Logging	32
	11.1 Log Levels	32
	11.2 Log Location	32
12	Advanced Information	33
	12.1 Transaction Object	33

1 Introduction

This manual describes how to install configure and use the payment module for WooCommerce and PAYONE.

Before you get started, make sure to have all necessary data at hand.

- You received a user name and a password from PAYONE. Log in to the PAYONE Merchant Interface. (<https://pmi.pay1.de/>)
- WooCommerce payment module from sellxed.com/shop
- Login data for your server and shop

2 PAYONE Channels

According to the functions of the module which you want to use, you have to request the different channels at PAYONE. With the numbers of channels come various costs on the part of the PAYONE. Here you'll find an overview of the different functions and their corresponding channels.

- **Channel Frontend:** Payment page
- **Channel Client-API:** Hidden, Alias Manager, Ajax, Moto
- **Server-API:** Refund, Capture, Cancel

Note that you must use at least PHP version 5.6 for our plugins. PHP 8 or higher is currently not supported.

2.1 Installation Process

This document contains all information necessary to install the module. Make sure to follow the described steps strictly in order to ensure the safe use of the module in compliance with all security regulations.

1. Configuration of the PAYONE test administration surface. The test platform can be found under <https://pmi.pay1.de/>
2. Configuration of the main settings
3. Configuration of the payment methods
4. Conducting a test order by means of the [Test Data](#) enclosed at the end of this document

Installation Service

Our payment plugins should have per default the correct settings for most of our customers' preferences. That means once you have entered the required credentials in the plugin configuration to connect your account to your website, the plugin should be fully operational. Should you be willing to receive detailed information on a setting you do not know, you may contact our support team who will be able to assist you further.

Our support team is at your disposal during regular business hours at: <http://www.sellxed.com/support>. Furthermore, you have the option of ordering our installation service. We will make sure the plugin is installed correctly in your shop: <http://www.sellxed.com/shop/de/integration-und-installation.html>

.htaccess Directory Protection

In order to test the module, any kind of directory protection or IP blocking on your server must be deactivated. This is crucial as otherwise the payment feedback of PAYONE might not get through to the shop.

2.2 System Requirements

In general, the plugin has the same system requirements as WooCommerce. Below you can find the most important requirements of the plugin:

- PHP Version: 5.4.x or higher
- OpenSSL: Current version with support for TLS 1.2 or higher.
- fsockopen: The PHP function fsockopen must be enabled. The plugin must be able to connect to external systems over the Internet.
- PHP Functions: All common PHP functions must be enabled.

3 PAYONE - Backend Configuration

First, log on to the PAYONE Merchant Interface (<https://pmi.pay1.de/>)

3.1 Payment Portal Setup

Under **Configuration > Payment Portals** you can set up the payment portal for your shop. Among other things you define the TransactionStatus URL. This is the URL that the parameters for further processing are sent to after payment has taken place.

3.1.1 Shop Portal Administration: Tab General

General settings such as portal name, Shop URL and the sender e-mail can be defined in the tab 'General'. These settings can be defined independently.

Make sure that the Server API, Front-end and Client API are activated in the Channel settings as shown in the screenshot.

The screenshot displays the 'Shop portal administration' interface. At the top, there is a navigation bar with 'PAYONE WEBSITE' and a 'HELP' button. The main header includes the 'PAYONE MERCHANT INTERFACE' logo and a user login status: 'You are logged in as info.19405 (Account: customweb_TEST [19405])' with 'LOGOUT' and 'deutsch' options.

The left sidebar contains a menu with categories like 'CUSTOMER ACCOUNTS', 'CHANNELS', 'CONFIGURATION', 'PAYMENT PORTALS', 'REPORTING', 'EXPORT', 'DOWNLOADS', 'ACCOUNTING', and 'ADMINISTRATION'. The 'PAYMENT PORTALS' section is expanded to show 'ACCOUNTS', 'TRANSACTION STATUS', 'REPORTING', 'EXPORT', 'DOWNLOADS', 'ACCOUNTING', and 'ADMINISTRATION'.

The main content area is titled 'Shop portal administration' and shows 'Portal: Customweb / Woocommerce'. It has tabs for 'General', 'Extended', 'Options', 'API-Parameter', and 'Riskcheck'. The 'General' tab is active and contains the following fields and options:

- Portal name*:** Customweb / Woocommerce
- URL:** http://dev.customweb.com
- Sender email*:** info@customweb.com (For emails to the end customer)
- Select channels:**
 - FinanceGate Server-API
 - FinanceGate Frontend
 - FinanceGate Client-API
- Additional response-data:**
 - Mode Live:** Switch on for additional values in API responses (Mode = Live).
 - Switched on:
 - Switched off:
 - Mode Test:** Switch on for additional values in API responses (Mode = Test).
 - Switched on:
 - Switched off:

At the bottom right of the form, there are 'Delete portal' and 'Save' buttons. The footer of the page includes '© PAYONE' and 'Top of page ↑'.

Figure 3.1: Tab 'General'

3.1.2 Shop Portal Administration: Tab Extended

In this tab you define the TransactionStatus URL. The URL you need to enter can be found directly on the general information site of the module in your shop. More information can be found in the section [Transaction-Feedback](#)). The remaining fields can be left empty. They will be filled by the module.

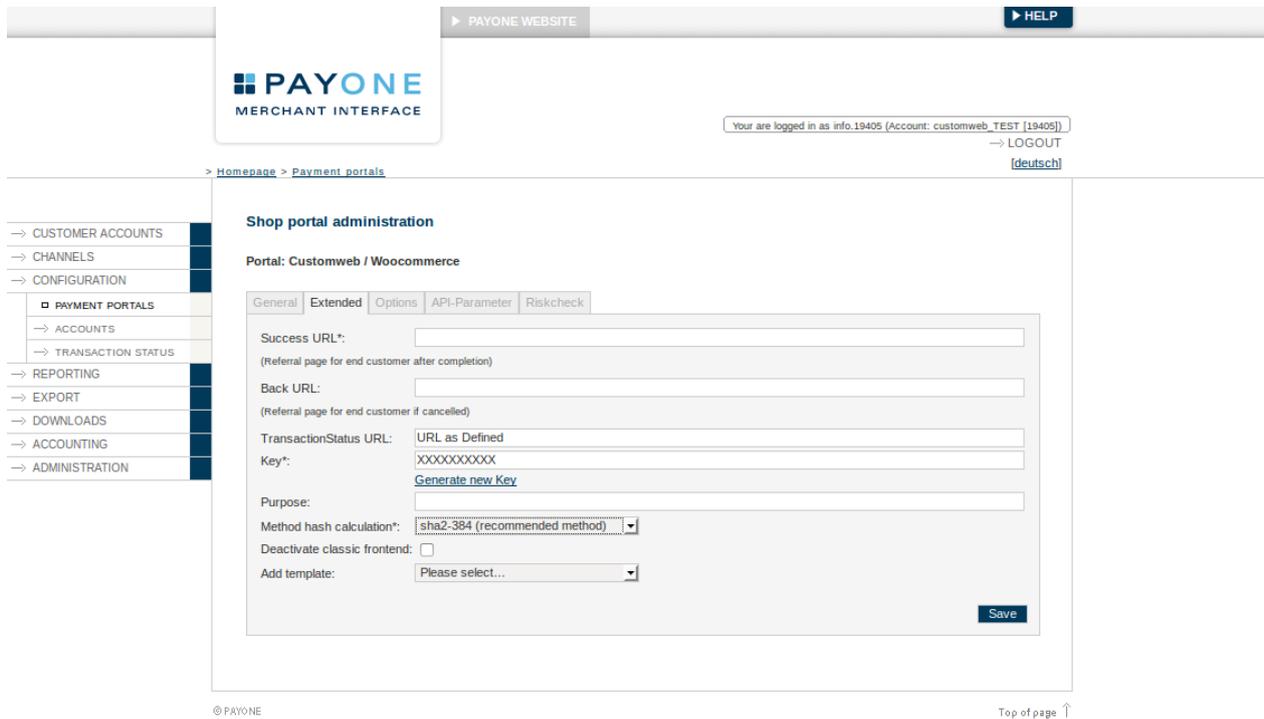


Figure 3.1: Extended configuration of the payment portal

3.1.3 Shop Portal Administration: API-Parameter

This tab contains all relevant information to enter in the [Main Module](#).

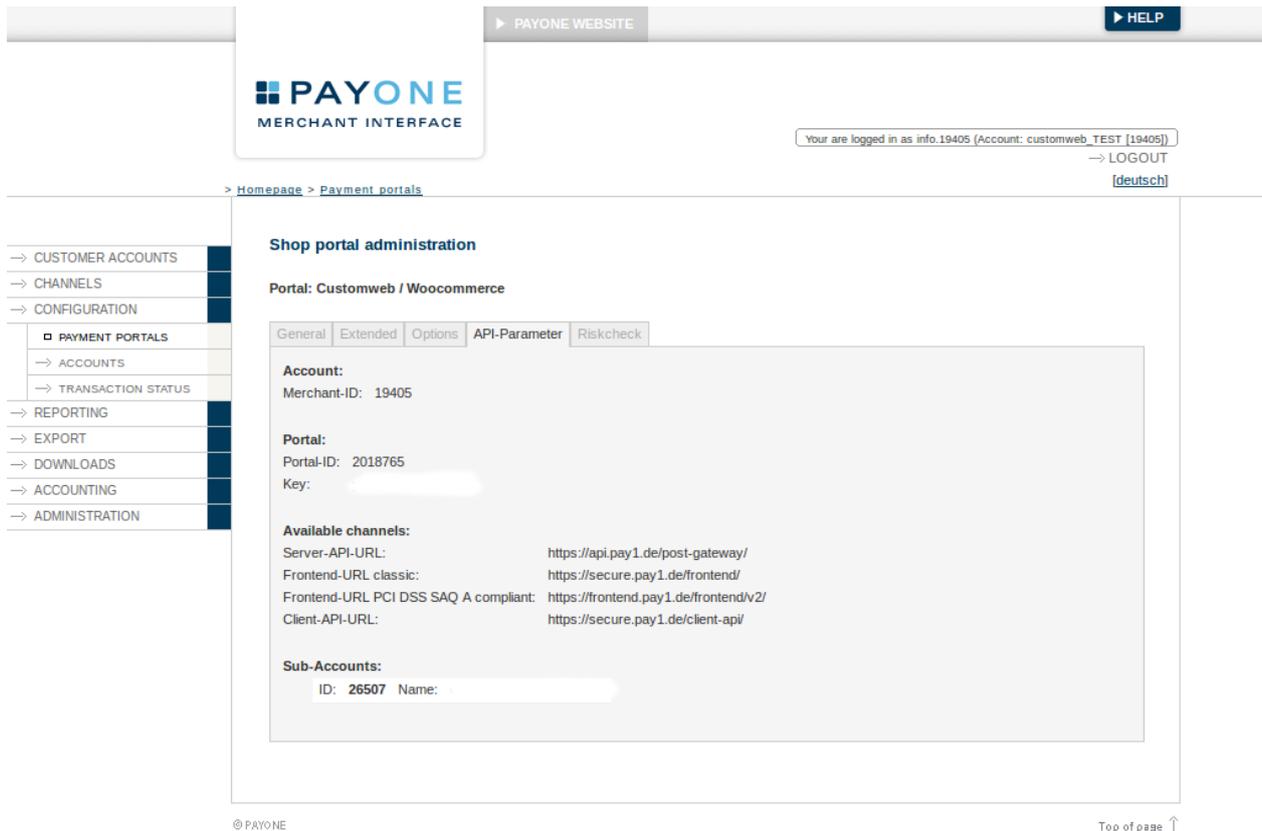


Figure 3.1: API-Parameter Configuration.

3.2 Authorization Methods

PCI 3.1 - SAQ A-EP

Please note that with the changes in PCI new rules apply regarding the applicable self assessment questionnaire. If you host the forms to entering of the credit cards directly in your webshop (Hidden Authorization) or Ajax authroization new rules apply. If you want to continue to work with SAQ-A, you have to use Payment Page or the Widget Authorization.

In the configuration of the payment method you have the possibility to switch between various authorization methods. These regard primarily the depiction of the credit card forms in the shop. You'll find further information about that in chapter [authorization methods](#).

3.3 Account Activation

Further information on the payment method configuration and the implementations of the settings can be found [here](#). After the activation and configuration of the payment methods you can conduct a test by means of the [Test Data](#) provided in this manual. As soon as the tests As soon as the tests have been successful you can contact PAYONE to demand the activation of your account. You will be able to process payments immediately.

4 Module Installation and Update in the WooCommerce Shop

4.1 Installation

At this time you should already be in possession of the module. Should this not be the case, you can download the necessary files in your customer account in the [sellxed shop](#) (Menu "My Downloads Downloads"). In order to install the module in your shop, please carry out the following steps:

1. Download the plugin. The download can be found in your sellxed.com account under "My Downloads".
2. Unzip the archive you have just downloaded.
3. In the unzipped folder navigate to the folder "files"
4. For some shops there are different versions of the plugin provided. If this is the case open the folder which corresponds to your shop version.
5. Using your preferred FTP client upload **entire content** of this folder into the root directory of your shop. For some shops there is a specific folder containing the plugins. If that is the case upload the plugin into this folder. Make sure that the folders aren't replaced but merely merged.
6. If you haven't yet done so, log back into your shop.

4.2 Updates and Upgrades

You have direct and unlimited access to updates and upgrades during the duration of your support contract. In order to receive constant information about available updates we ask you to subscribe to our RSS feed that we publish for your module.

More information regarding the subscription of this RSS feed can be found under: http://www.sellxed.com/en/updates_upgrades.

We only recommend an update if something doesn't work in your shop, if you want to use new feature or if there is a necessary security update.

4.2.1 Update Checklist

We ask you to strictly comply with the checklist below when doing an update:

1. Always do a backup for your database and your files in your shop
2. Use always a test system to test the update process.
3. Wait until all the files are copied to the shop, clear the cache if there is one in your shop and then visit the configuration page of the main module so that the update process will be initialized.

Do not do updates directly in the live environment

Please test the update procedure first in your test shop. Our support team is able and willing to help you if you experience problems with the update process. However, if you decide to perform the update directly in your live shop there is the possibility of a downtime of the shop of more than two days depending on the availability of our support if you do not want to book our [complementary support](#).

Depending on the version it could be that the database has to be migrated. We recommend you therefore, to perform the updates in times when the shop is not visited too frequently by your customers.

4.2.2 Update Instructions

Please always read the update instruction. Those instructions can be found in the changelog. If there are no special remarks, you can proceed by just overwriting the files in your system.

5 Module Configuration in the WooCommerce Shop

The configuration consists of two steps. The first step is the configuration of the main module with all the basic settings (cf. [Configuration of the Main Module](#)). During the second step you can then carry out individual configurations for each [payment method](#). This allows for full flexibility and perfect adaptation to your processes.

Create backups!

Please create a backup of the main directory of your shop. In case of problems you will then always be able to return your shop to its original state.

We furthermore recommend testing the integration on a test system. Complications may arise with third party modules installed by you. In case of questions, our support is gladly at your disposal.

5.1 Configuration of the Main Module

You can activate the plugin by clicking "activate" under the module WooCommerce PAYONE in the menu **Plugins**.

After the Installation of the plugin a new menu on the dashboard will appear called **PAYONE**. Enter all data in the corresponding fields. Each option is, furthermore, explained in short info texts in the shop.

5.2 Defining the URL for the Transaction Feedback

To ensure correct payment processing in your shop, at least one Feedback URL has to be provided in the PAYONE configuration. The URL to be entered can be found in your WooCommerce Shop under: **PAYONE > Setup > Setup**

5.3 Configuration of the Payment Module

After having successfully configured the main module, you can find the settings for the individual payment methods in your shop under **WooCommerce > Settings > Checkout** (In earlier version it was **WooCommerce > Settings > Payment Gateways**). Each payment method is listed individually. Install the payment methods you wish to offer to your customers. You can carry out individual settings for each payment method and thereby optimally adapt the payment methods to your existing processes. The most central options are described in more detail further below.

By clicking on **Enable** the payment method is activated in your shop. For more detailed information on the settings, please refer to information provided directly within the module configuration.

5.4 Direct Capturing of Transactions

The option "Capture" allows you to specify if you wish to debit payments directly or if you first wish to authorise them and then debit the payment at a later point.

Depending on your acquiring contract, a reservation is only guaranteed for a specific period of time. Should you fail to debit the payment within that period, the authorisation may therefore no longer be guaranteed. Further information on this process can be found below.

Different settings between PAYONE and the module

It may be that settings saved in the payment modules overwrite settings saved in PAYONE.

5.5 Uncertain Status

You can specifically label orders for which the money is not guaranteed to be received. This allows you to manually control the order before shipment.

5.5.1 Setting the order state

For each payment method you may select in which state the order should be set to depending on the booking state. This is the initial state of the order.

5.6 Optional: Validation

Note: It can be that this option is not visible in your module. In this case just ignore this section.

With the option 'Validation' you can define the moment when the payment method should be made visible to the customer during the checkout process. This setting is relevant for modules where the usage depends on the customer's compliance with specific preconditions. For example, if a solvency check has to be carried out or if the payment method is only available in certain countries. In order for the credit check or address validation to also work with European characters, the charset of the "Blowfish mode" must be set to "UTF-8" for certain PSP settings.

You have the choice between these options:

- **Validation before the selection of the payment method:** A validation verification is carried out before the customer selects the payment method. If the customer does not fulfill the requirements, the payment method is not displayed
- **Validation after selection of the payment method:** The verification of the compliance occurs after the selection of the payment method and before the confirmation of the order
- **During the authorisation:** The validation verification is carried out by PAYONE during the authorisation process. The payment method is displayed in any case

5.7 Redirection / Payment Form Error

If the successful redirection to the payment page suddenly stops working or you are only able to see the header and footer of your template instead of the payment form, you might be affected of the following issue. As a first step, you should update your payment plugin to the latest version, deactivate it in the backend of WooCommerce and activate it again.

If you are using **WPML** it can happen that the page for the redirection of the customer to PAYONE payment page results in a '404 Page Not Found' error. In this case, you will have to duplicate our checkout page under 'Pages' in the corresponding languages. In order to do so please follow these steps:

1. Go to the WooCommerce backend and select: Pages > All Pages. There you should find a Page called "PAYONE Checkout".
2. Delete all existing pages called "PAYONE Checkout".
3. In the plugin section of your Wordpress dashboard, deactivate the WooCommerce PAYONE plugin and activate it again. This will help you to generate a new redirection page.
4. Navigate again to Pages > All Pages. Open the configuration of the new redirection page.
5. On the right hand side under 'languages' you can now duplicate the content of this page for all of your installed languages. When you have successfully saved the changes, this page will be active for all of your languages and the error should have disappeared.

6 Settings / Configuration of Payment Methods

6.1 General Information About the Payment Methods

The plugin contains the most common payment methods. In case a desired payment method is not included per default, please contact us directly.

In order to be able to use a payment method, it must be activated in your account with PAYONE as well as in your shop. Information about the configuration of the payment methods can be found further above.

Below you can find important information for specific payment methods that deviate from the standard process.

6.2 Information on Payment Status

For each payment method you can define an initial payment status (status for authorized payments etc.). You hereby define the payment status for each state depending on the processing type of the order (captured, authorized, etc.). It's the initial status which the order assumes. Depending on the mutation carried out by you, the status can change.

Important info regarding Order Status

Never set the status to **Pending PAYONE** or any similar pending status which is implemented by the module.

6.2.1 Order status "pending" / imminent payment (or similar)

Orders with the status 'pending PAYONE' are pending orders. Orders are set to that status if a customer is redirected in order to pay but hasn't returned successfully or the feedback hasn't reached your shop yet (Customer closed window on the payment page and didn't complete payment). Depending on the payment method these orders will automatically be transformed into cancelled orders and the inventory will be cleared (so long as the Cronjob is activated). How long this takes depends on the characteristics of the payment method and cannot be configured.

If you have a lot of pending orders it usually means that the notifications from your webserver to PAYONE are being blocked. In this case check the settings of your firewall and ask the Hoster to activate the IPs and User Agents of PAYONE.

6.2.2 Order status "cancelled"

Orders with the status "cancelled" have either been set to that status automatically due to a timeout, as described above, or have been cancelled directly by the customer.

7 The Module in Action

Below you will find an overview of the most important features in the daily usage of the PAYONE module.

7.1 Useful Transaction Information on the Order

You can find an overview of the transaction information in within the order detail view. Among others, this information allows for the definite attribution of the orders to their corresponding transaction, seen in the backend of PAYONE.

#	Transaction Number	Date	Payment method	Is authorised	Amount	Customer note	Add
8	26_8	2013-10-04 11:29:35	Saferpay Test Card	Yes	200.00		Less details
Authorisation amount				200.00			
Currency				EUR			
Payment method				Saferpay Test Card			
Payment ID				dznhzlbUOMIGAYWG618AIWKp5MA			
Captured amount				200			
Refunded amount				200			
Transaction authorised				Yes			
Transaction uncertain				No			
Transaction captured				Yes			
Transaction paid				Yes			
Alias				xxxx xxxx xxxx 0111			
Card number				xxxx xxxx xxxx 0111			
Card expiry date				12/2015			
Card Type				Saferpay Test Card			
3D Secure				Successful			
Captures							

Figure 7.1: Transaction Information in WooCommerce.

7.2 Usage of the Alias Managers / Token Solution

With the Alias Manager, your customers can securely save their credit cards with PAYONE for later purchases. You can enable this by activating the option "Alias Manager" in the configuration of the [Payment Method](#). The customer can then choose from his or her saved credit cards without having to re-enter all the details.

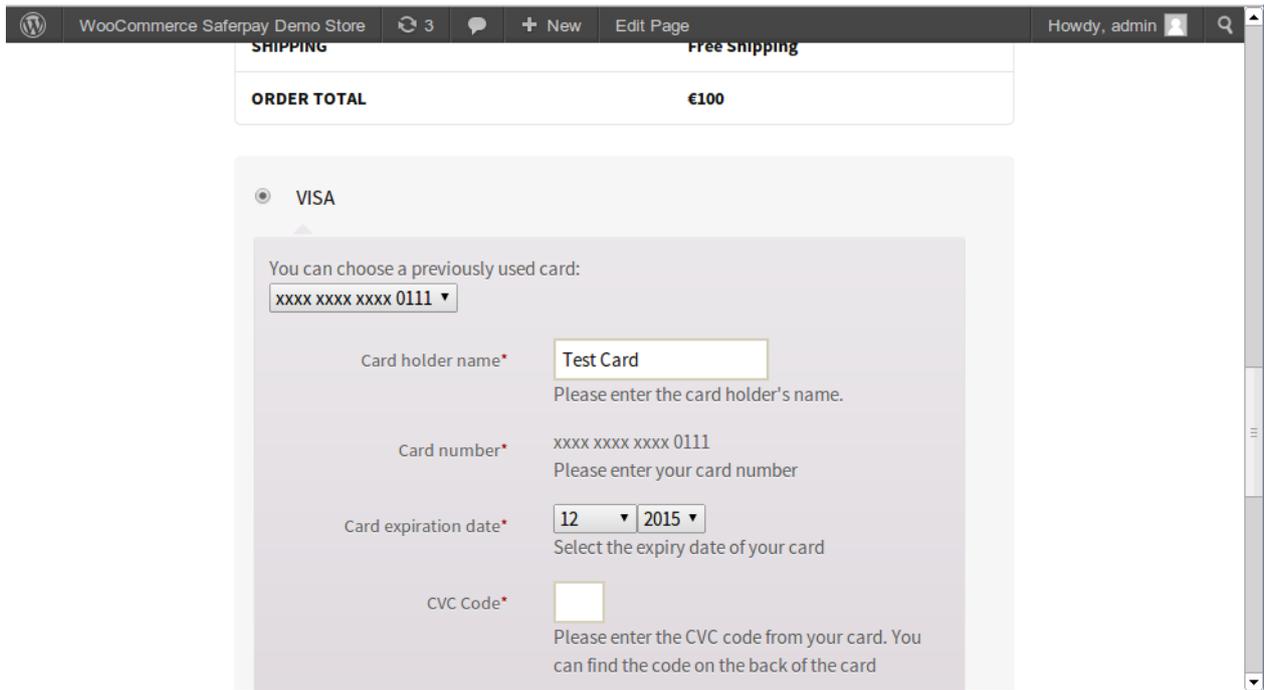


Figure 7.1: Alias Manager in WooCommerce

Alias Manager Options

The usage of the Alias Managers requires the activation of the correct option with PAYONE. To do so, please contact the support directly.

7.3 Capturing of Orders

Please note

The transaction management between your shop and PAYONE is not synchronized. If you capture payments with PAYONE, the status in the shop will not be updated and a second capturing in the shop is not possible.

In order to be able to capture orders you must make sure that you have set the option 'Capturing' to 'deferred' in the [payment method configuration](#).

Open the order and look for the PAYONE transaction tab at the bottom. Capturing via the internal operation of WooCommerce is not supported.

Customweb Transactions

#	Transaction Number	Date	Payment Method	Authorised	Amount
6	16	2017-01-06 10:23:49	MasterCard	Yes	10.00

[Less Details](#)

Authorisation Amount	10	Currency	CHF
Payment Method	MasterCard	Payment ID	452Wtjb3E7d7tASKGxK1bCEWxhdA
Test Transaction	Yes	Transaction authorised	Yes
Transaction uncertain	No	Transaction paid	Yes
Payment Method Name	MasterCard	Payment Details	xxxx xxxx xxxx 0000
Card holder name	Customweb	Card Expiry Date	12 / 20
Card Country Origin	DE	Acquirer Reference	000000
3ds Authentication	Yes	3ds Liability Shift	Yes
Authorisation Method	PaymentPage		

Previous actions

Date	Action	Message
2017-01-06 10:24:21	authorization	The amount of 10.00 is authorized.

Figure 7.1: Transaction details in the transaction tab of a specific order.

Click on "More Details" to view the transaction details. By clicking **Capture** button, you get into the following context.

Name	SKU	Type	Tax	Quantity	Total Amount (excl. Tax)	Total Amount (incl. Tax)
Testproduct	Testproduct	product	0 %	1	10.00	10.00

Total captured amount: 10.00CHF

Figure 7.1: Deferred/Manual capture of transactions within WooCommerce

Enter the amount of your choice that you wish to capture from the customer's card. By clicking **Capture** a direct capturing of the order occurs with PAYONE.

Please be aware that cancelling the transaction does not cause a status change of the order.

7.4 Refunds

You can refund already captured transactions and automatically transmit them to PAYONE.

Please note

That the default WooCommerce functionality is not supported by our plugin.

Customweb Transactions

#	Transaction Number	Date	Payment Method	Authorised	Amount
4	13	2016-11-04 14:50:07	MasterCard	Yes	20.00
Less Details					
Authorisation Amount	20	Currency	EUR		
Payment Method	MasterCard	Payment ID	b3YMh8bj8d4pvAnj0ddGAGvj3t9A		
Captured amount	20	Test Transaction	Yes		
Transaction authorised	Yes	Transaction uncertain	No		
Transaction captured	Yes	Transaction paid	Yes		
Payment Method Name	MasterCard	Payment Details	xxxx xxxx xxxx 0000		
Card holder name	Customweb	Card Expiry Date	12 / 20		
Card Country Origin	DE	Acquirer Reference	000000		
3ds Authentication	Yes	3ds Liability Shift	Yes		
Authorisation Method	PaymentPage				
Refund					
Captures					
Date		Amount		State	
2016-11-04 14:51:34		20		succeed	
Previous actions					
Date	Action	Message			
2016-11-04 14:51:33	authorization	The amount of 20.00 is authorized.			
2016-11-04 14:51:34	capturing	The amount of 20.00 is captured.			

Figure 7.1: Transactionsdetails via Orders > PAYONE Transactions.

By clicking **Refund** button, you get into the following context.

Name	SKU	Type	Tax	Quantity	Total Amount (excl. Tax)	Total Amount (incl. Tax)
Testproduct	Testproduct	product	0 %	2	20.00	20.00
Total refunded amount:						20.00CHF
Close transaction for further refunds						<input type="checkbox"/>
<input type="button" value="Back"/>				<input type="button" value="Refunding"/>		

Figure 7.1: Refund of transactions within WooCommerce

Just like for the capturing, you open the order and click on the magnifying glass next to the order of your choice. Enter any refund amount of your choice in the field provided for this purpose. By clicking **refund** the refund request will directly be sent to PAYONE and the specified amount will be credited on the customer's credit card.

Please note that you can't refund more than 100% of the original amount.

Furthermore, please be aware that processing a refund will not cause a status change of the order.

7.5 Transaction Process

Each action affiliated with transaction management is logged by the module.

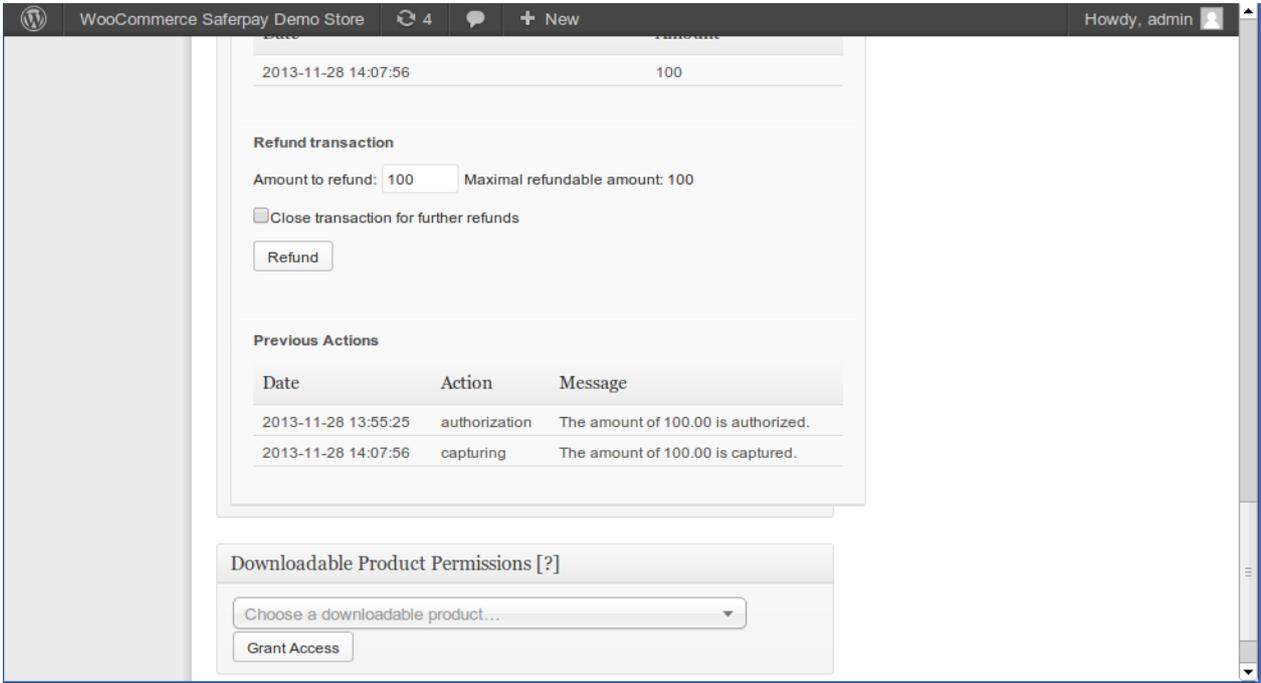


Figure 7.1: Transactions in the Overview of the Transaction Process

7.6 Using Invoice Details of a Processor

In the following context you can view or embed the "payment details" of for example an "Open Invoice" transaction:

7.6.1 WooCommerce Order Confirmation (E-Mail)

The "payment information" will be visible in the default "order confirmation e-mail" of WooCommerce.

7.6.2 WooCommerce Invoice (PDF)

The "payment information" will be visible in the default WooCommerce-Invoice.

7.6.3 WooCommerce-Backend (Transaction details)

You can view the transaction details in WooCommerce under **WooCommerce > Orders > PAYONE transactions > More Details**.

The screenshot displays the 'Customweb Transactions' interface. At the top, there is a table with columns: #, Transaction Number, Date, Payment Method, Authorised, and Amount. A single transaction is listed with ID 3, Transaction Number 12, Date 2017-03-07 15:12:01, Payment Method 'Open Invoice', Authorised 'Yes', and Amount 10.00. A 'Less Details' button is visible to the right of the table row.

Below the table, there are several sections of details:

- Authorisation Amount:** 10
- Payment Method:** Open Invoice
- Test Transaction:** Yes
- Transaction paid:** Yes
- Currency:** EUR
- Payment ID:** 3575423
- Transaction authorised:** Yes
- Transaction uncertain:** No

Payment Information:

- Bank: Customweb Test Bank
- Account Holder: Customweb GmbH
- IBAN: DE2501200000TEST000000000003
- BIC: TESTBIC0003
- Reference Number: BP3575423/2749

Buttons for 'Capture' and 'Cancel' are located below the payment information.

Previous actions:

Date	Action	Message
2017-03-07 15:12:22	authorization	The amount of 10.00 is authorized.

Figure 7.1: Transaction details within WooCommerce.

7.6.4 WooCommerce Success-Page

Thank you. Your order has been received.

- Order Number: **12**
- Date: **March 7, 2017**
- Total: **€10.00**
- Payment Method: **Open Invoice**

Order Details

Product	Total
<u>Test</u> × 1	€10.00 (ex. tax)
Subtotal:	€10.00 (ex. tax)
Payment Method:	Open Invoice
Total:	€10.00

Customer Details

Email:	info@customweb.com
Telephone:	12345678

Billing Address

André Müller
 Schöneeggstrasse 2
 8004 Zürich

Payment Information

Bank: Customweb Test Bank
 Account Holder: Customweb GmbH
 IBAN: DE2501200000TEST000000000003
 BIC: TESTBIC0003
 Reference Number: BP3575423/2749

Figure 7.1: Payment information on the WooCommerce "Success-Page".

7.7 Setup a Cron Job to Activate the Timed Operations

To activate the timed operations of the plugin (e.g. update service, deleting pending orders, etc.) make sure that you set up the WooCommerce Cron engine. Especially the update function allows

you to automatically retrieve additional information or changes of your order directly via the API of PAYONE. Please note it could be that in order to use the update feature it may be necessary that PAYONE activates additional options in your account.

The module uses the standard cron engine of WooCommerce. More information regarding the set up can be found [here](#).

8 Testing

Before switching from test to live mode it is important that you test the module extensively.

Testing

Do not forget to switch the operating mode from test to live after having successfully tested the module.

8.1 Test Data

In the following section you can find the test data for the various payment methods:

SOFORT

Sofort	
IBAN	DE85123456782599100003
BIC	TESTTEST
Bank account	2599100003
Bank code	12345678

giropay

Sofort	
IBAN	DE46940594210000012345
BIC	TESTDETT421
ID / PIN	sepatest1 / 12345
NR / TAN	MAS_Test / 123456

Direct Debits

Hidden Mandate	Accept
IBAN	DE00123456782599100004
BIC	TESTTEST
Hidden Mandate	Denied - Invalid BIC
IBAN	DE00123456782599100004
BIC	TESTTESX

Credit / Debit Card

VISA	Verified by Visa	Visa 3D
Card number	4012 0010 3714 1112	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
3-D Secure Password	12345	
VISA	CH	Visa CH
Card number	4111 1111 1111 1111	
Expiry Date:	12/2020	

Name	Hans Muster	
CVC:	123	
VISA	DE	Visa DE
Card number	4111 1310 1011 1111	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
VISA	AT	Visa AT
Card number	4111 1210 1111 1111	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
MasterCard	MasterCard SecureCode	Mastercard 3D
Card number	5453 0100 0008 0200	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
3-D Secure Password	12345	
Mastercard		Mastercard
Card number	5500 0000 0000 0004	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
American Express		American Express
Card number	340 0000 0000 0009	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	1234	
JCB	JCB J/Secure	JCB 3D
Card number	3528 4501 3100 3315	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
3-D Secure Password	12345	
JCB		JCB
Card number	3088 0000 0000 0009	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
Maestro		Maestro
Card number	5000 0000 0000 0009	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	

Carte Bleue		Carte Bleue
Card number	4973 0100 0000 0004	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
Diners Club		Diners Club
Card number	30 0000 0000 0004	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
Discover		Discover Card
Card number	6011 1111 1111 1117	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
MasterCard		
Mastercard		Mastercard
Card number	5500 0000 0000 0004	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
MasterCard	MasterCard SecureCode	Mastercard 3D
Card number	5453 0100 0008 0200	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
3-D Secure Password	12345	
Visa		
VISA	CH	Visa CH
Card number	4111 1111 1111 1111	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
VISA	DE	Visa DE
Card number	4111 1310 1011 1111	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
VISA	AT	Visa AT
Card number	4111 1210 1111 1111	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	

VISA	Verified by Visa	Visa 3D
Card number	4012 0010 3714 1112	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
3-D Secure Password	12345	
American Express		
American Express		American Express
Card number	340 0000 0000 0009	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	1234	
Diners Club		
Diners Club		Diners Club
Card number	30 0000 0000 0004	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
Discover Card		
Discover		Discover Card
Card number	6011 1111 1111 1117	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
Carte Bleue		
Carte Bleue		Carte Bleue
Card number	4973 0100 0000 0004	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
JCB		
JCB		JCB
Card number	3088 0000 0000 0009	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
JCB	JCB J/Secure	JCB 3D
Card number	3528 4501 3100 3315	
Expiry Date:	12/2020	
Name	Hans Muster	
CVC:	123	
3-D Secure Password	12345	

Maestro

Maestro

Maestro

Card number

5000 0000 0000 0009

Expiry Date:

12/2020

Name

Hans Muster

CVC:

123

9 Errors and their Solutions

You can find detailed information under <http://www.sellxed.com/en/fag>. Should you not be able to solve your problem with the provided information, please contact us directly under: <http://www.sellxed.com/en/support>

9.1 Page Not Found Error prior to the redirection to Payment Page

If you are using WPML it can occur that the Pages for the display of the payment methods or for the redirection to PAYONE end in a Page Not Found Error.

In this case you have to duplicate our Pages in the corresponding languages. In order to do so please follow these steps:

1. Go to the main menu and select **Pages > All Pages**. There you should find a Page called PAYONE Checkout.
2. Open the configurations.
3. On the right hand side under languages you can now duplicate the content of this page for all of your installed languages. When you have successfully saved the changes this page will be active for all of your languages and the Error should have disappeared.

9.2 The Referrer URL appears in my Analytics Tool

When a customer and the notification are redirected via Header Redirection, the PAYONE Referrer URL might appear in your Analytics Tool thus hiding the original traffic source. However, most Analytic Tools are able to minimize this problem.

In case you are using Google Analytics as reporting tool, this step by step guide may help you to exclude the URLs: [under bullet point 4](#).

10 Compatibility with Third-Party Plugins

The plugins listed below are compatible with our payment modules and allow you to handle certain tasks in an easier way.

10.1 Fee's and discount's within WooCommerce

To configure a PAYONE payment gateways based fee and discount you will need the following 3rd-Party plugin.

- [Checkout Fees for WooCommerce](#)

10.2 Multilingual WooCommerce sites

To run multilingual WooCommerce sites you will need the following 3rd-Party plugin.

- [WPML for WooCommerce](#)

10.3 Birthday and gender in WooCommerce

For certain payment service providers it is necessary to check the birthday and the gender of a customer. WooCommerce does not check this by default.

How to enable gender and birthday checks in your shops checkout

1. Either add custom checkout field yourself or download [Flexible Checkout Fields](#) to do so.
2. Modify the order context getters to return the value of your custom checkout field from the order / session (or wherever the previous step saves the data).

Order Context Getters

- AbstractOrderContext
- getBillingDateOfBirth()
- getBillingGender()

These functions can be found in "wp-plugins/woocommerce_payonecw/classes/PayoneCw/AbstractOrderContext.php".

11 Error Logging

The module will log different unexpected errors or information depending on the configured level. If there is any issue with the module, this log can help identify the cause.

11.1 Log Levels

You can configure the log level in the PAYONE settings.

- Error: Logs unexpected errors only. (Default)
- Info: Logs extended information.
- Debug: Logs information helpful for debugging.

11.2 Log Location

The log file is stored in **wp-content/uploads/wc-logs**. Please make sure the path exists and it is writable by the webserver.

12 Advanced Information

This section of the manual is for advanced usage of the module. The content is for advanced users with special requirements. Everything in this section is optional and not required for the daily usage of the module.

12.1 Transaction Object

This section describes how to extract information from a transaction, if you need it for further processing. E.g. you require more information of the transaction for further processing an order in your ERP system.

The code snippets in this section assume your script resides in the root folder of the shop with the default shop folder structure.

Have Wordpress initialized in your script, this is required for the database connection.

```
require_once( dirname(__FILE__) . '/wp-load.php' );
```

Include the modules Util class.

```
require_once ABSPATH.'wp-content/plugins/woocommerce_payonecw/classes/PayoneCw/Util.php';
```

Now you can load the transaction and then extract the transactionObject.

Load the transaction by Id:

```
$transactionById = PayoneCw_Util::getTransactionById($transactionId);
$transactionObject = $transactionById->getTransactionObject();
```

Load the transaction by the external Id:

```
$transactionByExternal = PayoneCw_Util::getTransactionByTransactionNumber($externalId);
$transactionObject = $transactionByExternal->getTransactionObject();
```

Load transactions by Post ID, this is the internal WooCommerce order id:

```
$transactionsByPostId = PayoneCw_Util::getTransactionsByPostId($orderId);
foreach($transactionsByPostId as $transaction){
    $transactionObject = $transaction->getTransactionObject();
    //Do something with each object
}
```