

Installation Manual for mPAY24 e-Commerce

This manual describes the installation and usage of the mPAY24 extension for e-Commerce.

Release Date: Mon, 07 Aug 2023 14:01:06 +0200
Version: 2.0.251

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
1.1	Installation Process	5
2	Configuration of the mPAY24 - Backend	6
2.1	Transaction Overview and Capture Settings	6
2.2	Configuration of Main Module Data	6
2.3	Configuration of Payment Methods and Testing	6
2.4	Invoice	6
2.5	Direct Debit	6
3	Module Installation and Update in the e-Commerce Shop	7
3.1	Installation	7
3.2	Updates and Upgrades	7
3.2.1	Update Checklist	7
3.2.2	Update Instructions	8
4	Module Configuration in the e-Commerce Shop	9
4.1	Basic Module Configuration	9
4.2	Payment Method Configuration	9
4.3	Direct Capturing of Transactions	10
4.4	Uncertain Status	10
4.4.1	Setting the order state	10
4.5	Optional: Validation	10
5	Settings / Configuration of Payment Methods	11
5.1	General Information About the Payment Methods	11
5.2	Information on Payment Status	11
5.2.1	Order status "pending" / imminent payment (or similar)	11
5.2.2	Order status "cancelled"	11
6	The Module in Action	12
6.1	Useful Information on Transactions	12
6.2	Alias Manager / Token Solution Usage	12
6.3	Capture / Cancel Orders	13
6.3.1	Capturing and Cancelling Orders	14
6.4	Order Refunds	14

6.5	Transaction History	15
6.6	Using Invoice Details of a Processor	16
6.6.1	e-Commerce Order Confirmation (E-Mail)	16
6.6.2	e-Commerce Invoice (PDF)	16
6.6.3	e-Commerce-Backend (Transaction details)	16
6.6.4	e-Commerce Success-Page	17
6.7	Cron Job Setup	18
7	Testing	20
7.1	Test Data	20
8	Errors and their Solutions	21
8.1	The Referrer URL appears in my Analytics Tool	21
9	Error Logging	22
9.1	Log Levels	22
9.2	Log Location	22
10	Advanced Information	23
10.1	Transaction Object	23

1 Introduction

This manual describes the installation, configuration and usage of the payment module for e-Commerce and mPAY24.

Before starting the installation, make sure to be in possession of all necessary data. You will need:

- mPAY24 Merchant ID and password
- e-Commerce payment module from sellxed.com/shop
- Login data to your server and shop

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

1.1 Installation Process

This document contains all information necessary to install the module. When activating the module you should make sure to follow the steps in this manual strictly. This ensures the safe use of the module in compliance with all security regulations.

1. Configuration of the [Main Module](#) by entering the login data
2. Configuration of the accepted [Payment Methods](#) including the authorization methods
3. Conducting a test order
4. Activating your mPAY24 account

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 mPAY24 might not get through to the shop.

2 Configuration of the mPAY24 - Backend

2.1 Transaction Overview and Capture Settings

You can access an overview of all transaction in the mPAY24 - Backend <https://www.mpay24.com/app/bin/mif/de/login>

In addition you should make sure to set the Capturing of credit card transactions to "Immediately". In order to do this, change the settings in the tab "Clearing" to automatic. This way incoming transactions will be charged immediately. If you do not activate automatic capturing, you need to ensure that all transactions are captured manually in the backend within the time limit.

A few other settings have to be adapted as well. The notification URL are always set by the module automativally. If you entered any URL into the backend in former installations they should be deleted.

2.2 Configuration of Main Module Data

First, enter the data provided by mPAY24 directly into the [Configuration of the Main Module](#).

2.3 Configuration of Payment Methods and Testing

Further information on the configuration of payment methods and the implementations of settings can be found [here](#). After the activation and the configuration of the payment methods you should conduct a test with the [Test Data](#) published in the manual. If all tests were successful you can set the Operation Mode in the Main Module to **Live**.

2.4 Invoice

The Payment Method Invoice is set as a standard feature of the module by mPAY24. You can find it under the name Open Invoice. Make sure to select one of the processors. You can choose between **Klarna** and **BILLPAY**.

2.5 Direct Debit

The Payment Method Direct Debit is set as a standard feature of the module by mPAY24. You can find it under the name Direct Debit. Make sure to select one of the processors. This choice has to be made in compliance with your contracts.

3 Module Installation and Update in the e-Commerce Shop

3.1 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.

3.2 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.

3.2.1 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.

3.2.2 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.

4 Module Configuration in the e-Commerce 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.

4.1 Basic Module Configuration

Activate the plugin by selecting the module E-commerce mPAY24 in the **Plugin** menu.

The settings for the basic module can be found at **mPAY24 > mPAY24**. Fill out the fields, you should have already entered the necessary data in the mPAY24 backend or you received it as part of your correspondence with mPAY24. Each option is explained in more detail in a short info text in the shop.

4.2 Payment Method Configuration

After the basic module has been configured successfully, you can proceed to the payment settings. The payment methods you want to accept in your shop can be adjusted at **Settings > Store > Payments**. Every payment method is listed individually. Activate the payment methods you would like to provide in your shop by ticking the corresponding boxes on the right side. Individual changes can be made for each payment method to adjust it to your processes.

4.3 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 mPAY24 and the module

It may be that settings saved in the payment modules overwrite settings saved in mPAY24.

4.4 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.

4.4.1 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.

4.5 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 mPAY24 during the authorisation process. The payment method is displayed in any case

5 Settings / Configuration of Payment Methods

5.1 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 mPAY24 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.

5.2 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 mPAY24** or any similar pending status which is implemented by the module.

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

Orders with the status 'pending mPAY24' 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 mPAY24 are being blocked. In this case check the settings of your firewall and ask the Hoster to activate the IPs and User Agents of mPAY24.

5.2.2 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.

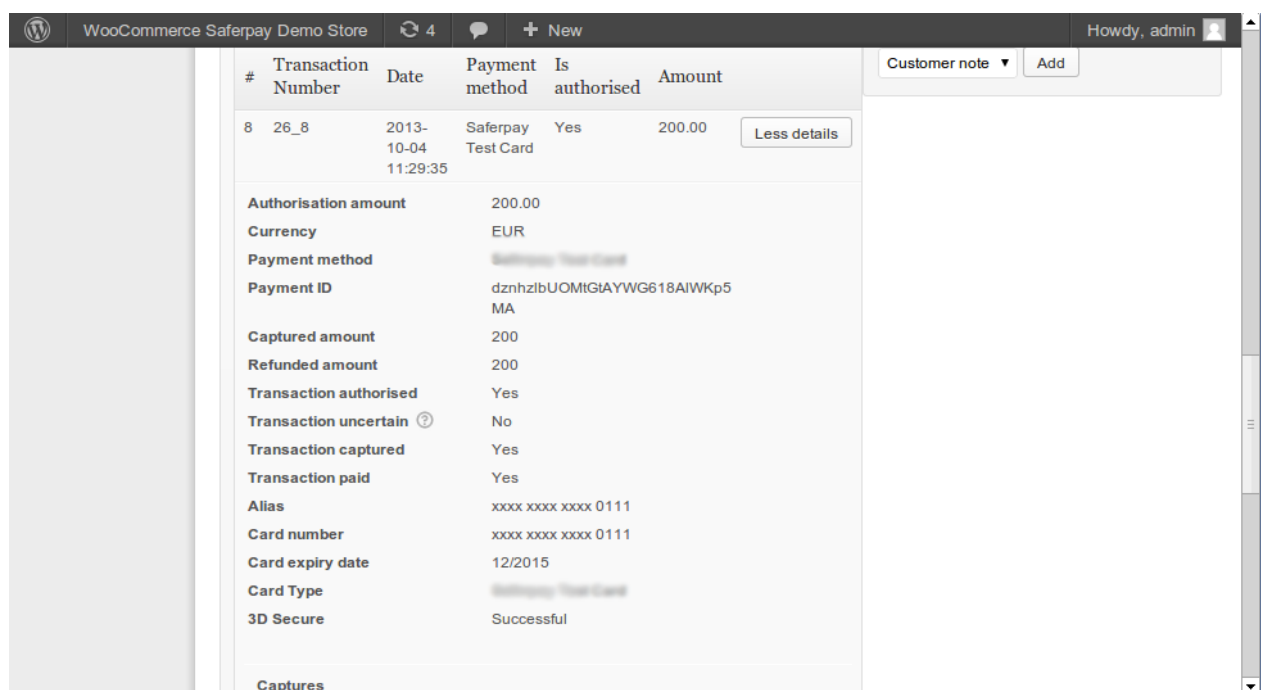
6 The Module in Action

Below you will find an overview of the most important features in the daily usage of the mPAY24 module.

6.1 Useful Information on Transactions

For each order processed with our module you can access an overview of the most important transaction information. Among other things, this allows you to identify each corresponding order to each transaction displayed in the backend.

You can find the orders at Dashboard > Store Sales.



#	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		dznhzlbUOMIGIAYWG618AIWKp5MA					
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 6.1: Transaction information in WordPress E-Commerce.

6.2 Alias Manager / Token Solution Usage

The Alias Manager allows you to store your customers' credit card data safely with mPAY24. You can enable this by activating the option "Alias Manager" in the configuration of the [Payment Method](#). The customer can then select between the saved cards when making purchases in the future and does not have to enter all credit card data again.

The screenshot shows a WooCommerce checkout page for 'WooCommerce Saferpay Demo Store'. The top navigation bar includes links for 'New' and 'Edit Page', and a user greeting 'Howdy, admin'. The checkout summary shows 'SHIPPING' as 'Free Shipping' and an 'ORDER TOTAL' of '€100'. The main form is for a VISA card payment. It includes a dropdown to select a previously used card (currently showing 'xxxx xxxx xxxx 0111'). Below this are four required fields: 'Card holder name*' (filled with 'Test Card'), 'Card number*' (filled with 'xxxx xxxx xxxx 0111'), 'Card expiration date*' (filled with '12' and '2015'), and 'CVC Code*'. Each field has a placeholder text indicating what to enter.

Figure 6.1: Alias Manager in WooCommerce

Alias Manager Options

In order to use the Alias Manager, the option has to be unlocked by mPAY24. Please contact the support service if you wish to activate the option.

6.3 Capture / Cancel Orders

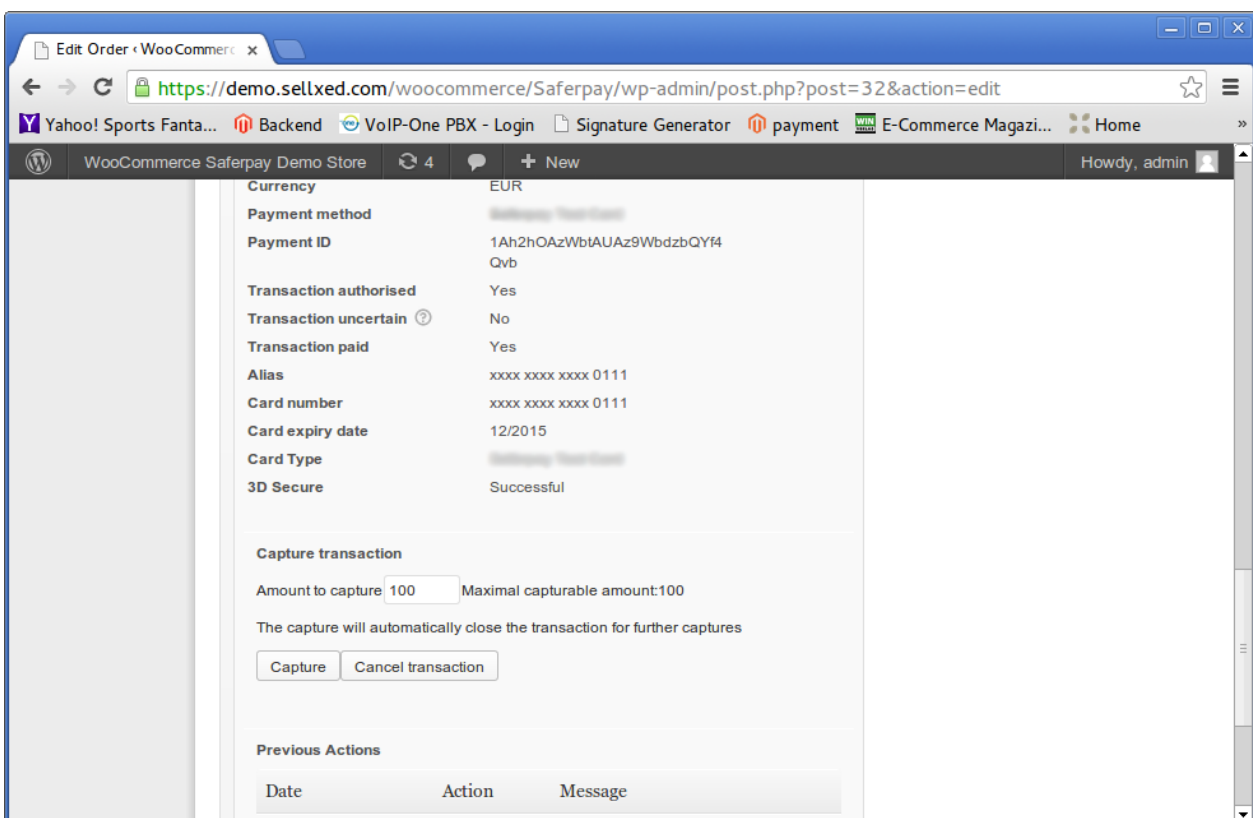
6.3.1 Capturing and Cancelling Orders

In order to capture orders, open the order first (Dashboard > Store Sales) and then open the transaction information. Enter the amount you want to capture into the input field. By clicking on Capture the card is charged directly.

An order can be **cancelled** by clicking "Cancel Transaction" and the transaction as well as the reserved amount is released on the customer's card immediately.

Capturing Orders in the mPAY24 Backend

The transaction management between mPAY24 and your shop is not synchronised. When capturing payments in the mPAY24, the status is not adjusted in the shop and a second capture is not possible.



The screenshot shows the 'Edit Order' page in the Sellxed WooCommerce admin. The browser address bar shows the URL: <https://demo.sellxed.com/woocommerce/Saferpay/wp-admin/post.php?post=32&action=edit>. The page title is 'Edit Order - WooCommerce'. The main content area displays transaction details:

Currency	EUR
Payment method	Saferpay Test Mode
Payment ID	1Ah2hOAzWbIAUAz9WbdzbQYf4Qvb
Transaction authorised	Yes
Transaction uncertain	No
Transaction paid	Yes
Alias	xxxx xxxx xxxx 0111
Card number	xxxx xxxx xxxx 0111
Card expiry date	12/2015
Card Type	Saferpay Test Mode
3D Secure	Successful

Below the details, there is a 'Capture transaction' section:

Amount to capture: Maximal capturable amount: 100

The capture will automatically close the transaction for further captures

Buttons:

At the bottom, there is a 'Previous Actions' table with columns: Date, Action, Message.

Figure 6.1: Cancelling and capturing orders

Partial Capturing

Please inquire, whether multiple captures are supported within your mPAY24 contract. Otherwise, partially captured transactions could be blocked from future captures.

6.4 Order Refunds

In order to refund an order, open the transaction information (see above).

You can refund individual items or an arbitrary amount by adjusting the total amount or the number of items.

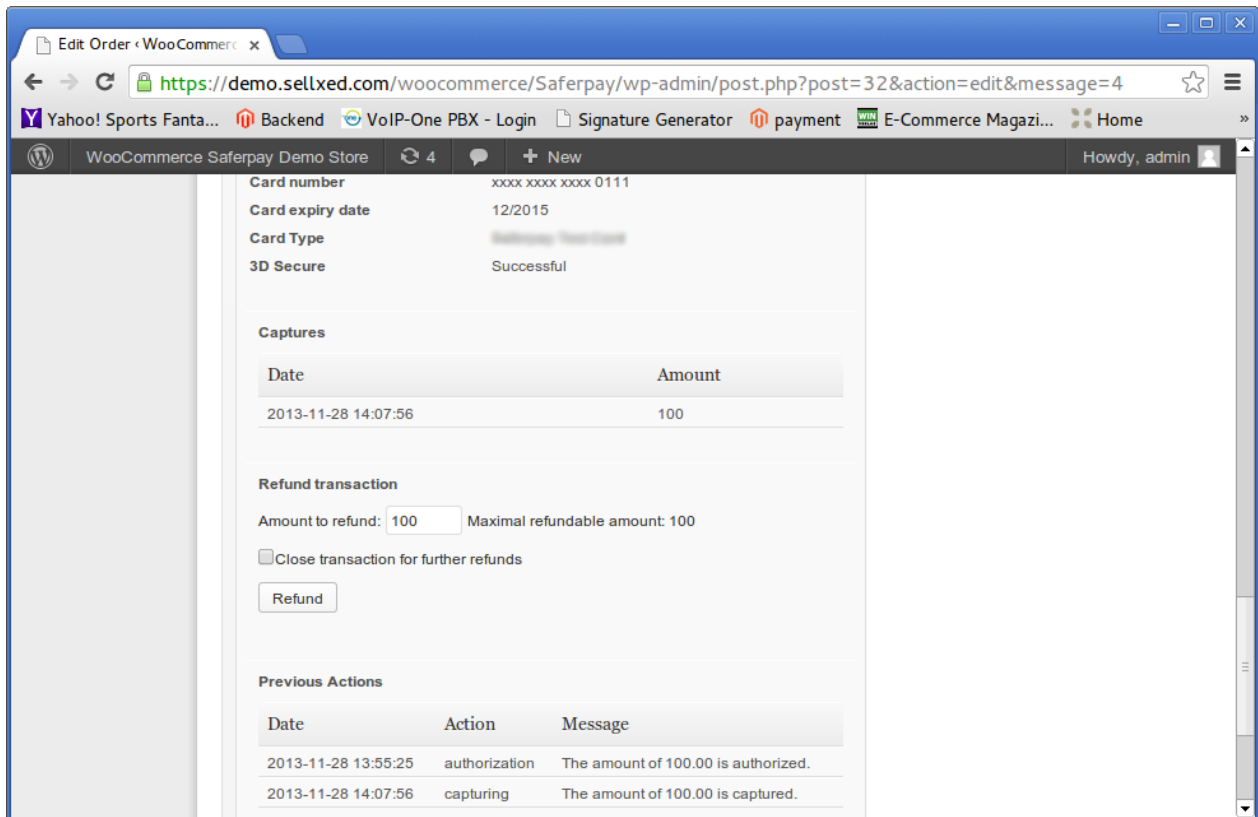


Figure 6.1: Refunds in WooCommerce for mPAY24.

Maximum Refund

Our module restricts the refund amount to 100% of the originally authorised amount.

6.5 Transaction History

Every action within the transaction management is processed through the module. Information on each transaction can be viewed in the corresponding order.

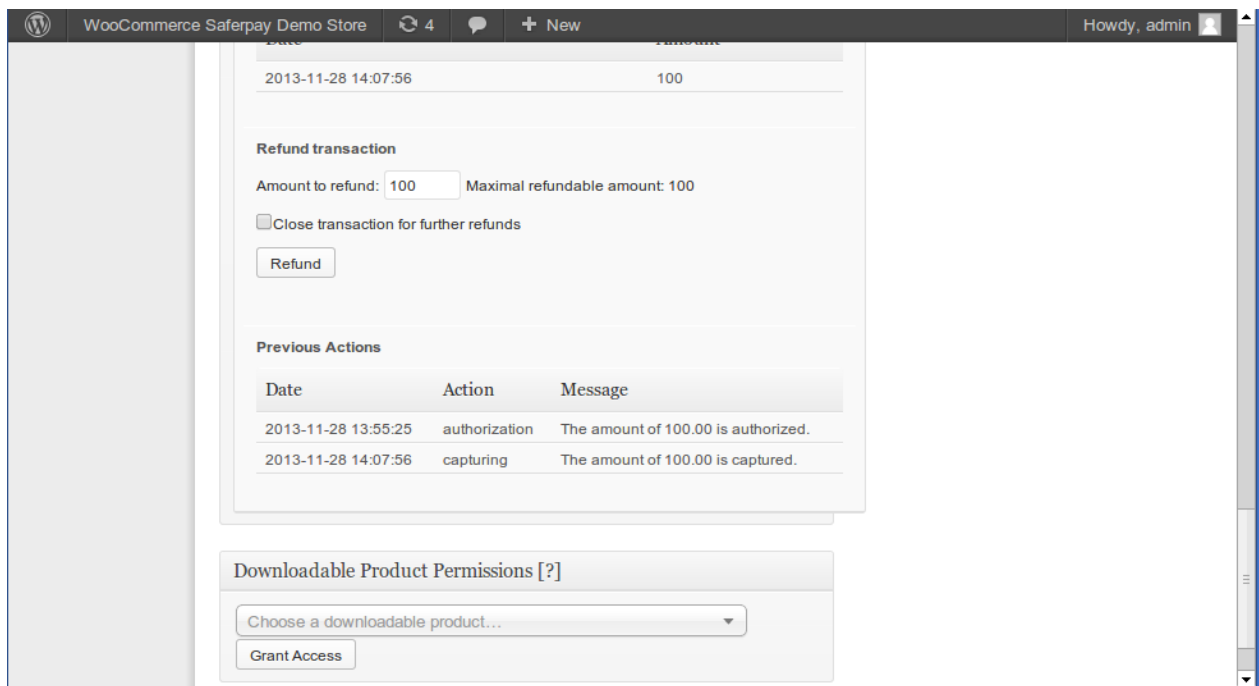


Figure 6.1: Transaction overview in the transaction history.

6.6 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:

6.6.1 e-Commerce Order Confirmation (E-Mail)

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

6.6.2 e-Commerce Invoice (PDF)

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

6.6.3 e-Commerce-Backend (Transaction details)

Customweb Transactions

#	Transaction Number	Date	Payment Method	Authorised	Amount
3	12	2017-03-07 15:12:01	Open Invoice	Yes	10.00

Less 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

Capture

Cancel

Previous actions

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

Figure 6.1: Transaction details within e-Commerce.

6.6.4 e-Commerce 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
Schöneggstrasse 2
8004 Zürich

Payment Information

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

Figure 6.1: Payment information on the e-Commerce "Success-Page".

6.7 Cron Job Setup

In order to activate the time-controlled plugin functions (e.g. update service, deleting pending orders etc.) you have to set up cron jobs in e-Commerce. The update function is especially

relevant - it allows you to fetch subsequent transaction changes at mPAY24 via the API. Please keep in mind that additional options might have to be activated.

The module uses the default e-Commerce cron. Information on the setup can be found [here](#).

7 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.

7.1 Test Data

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

Credit / Debit Card

Card number	4444333322221111
Expiry Date	05/2020
CVC	123

Visa

Card number	4444333322221111
Expiry Date	05/2020
CVC	123

Card number	4444333322221111
Expiry Date	12/2020
CVC	123

MasterCard

Card number	5555444433331111
Expiry Date	05/2020
CVC	123

Card number	5555444433331111
Expiry Date	12/2020
CVC	123

American Express

Card number	378282246310005
Expiry Date	12/2020
CVC	123

8 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>

8.1 The Referrer URL appears in my Analytics Tool

When a customer and the notification are redirected via Header Redirection, the mPAY24 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](#).

9 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.

9.1 Log Levels

You can configure the log level in the mPAY24 settings.

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

9.2 Log Location

The log file is stored in the logs folder of this plugin **wp-content/plugins/plugins/ecommerce_mpay24cw/logs**. Please make sure the path exists and it is writable by the webserver.

10 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.

10.1 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/ecommerce_mpay24cw/classes
/Mpay24Cw/Util.php';
```

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

Load the transaction by Id:

```
$transactionById = Mpay24Cw_Util::getTransactionById($transactionId);
$transactionObject = $transactionById->getTransactionObject();
```

Load the transaction by the external Id:

```
$transactionByExternal = Mpay24Cw_Util::getTransactionByExternalId
($externalId);
$transactionObject = $transactionByExternal->getTransactionObject();
```

Load transactions by the order id:

```
$transactionsByOrder = Mpay24Cw_Util::getTransactionsByOrderId($orderId);
foreach($transactionsByOrder as $transaction){
    $transactionObject = $transaction->getTransactionObject();
    //Do something with each object
}
```