

Installation Manual for EPRO Modified Shop

This manual describes the installation and usage of the EPRO extension for Modified Shop.

Release Date: Mon, 07 Aug 2023 14:01:06 +0200
Version: 1.0.22

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
1.2	System Requirements	6
1.3	Configuration of the Main Module	6
1.3.1	Configuration of the API Key	7
1.3.2	Configuration of the Notification URL	7
1.3.3	Configuration of the Scheme	7
1.4	Configuration of the Payment Methods	7
2	Module Installation and Update in the Modified Shop Shop	8
2.1	Installation	8
2.2	Updates and Upgrades	8
2.2.1	Update Checklist	8
2.2.2	Update Instructions	9
3	Module Configuration in the Modified Shop Shop	10
3.1	Configuration of the Main Module	10
3.1.1	Create Pending Orders	10
3.1.2	Database Encoding	11
3.1.3	Order Prefix	11
3.1.4	Order Prefix	11
3.2	Direct Capturing of Transactions	11
3.3	Uncertain Status	11
3.3.1	Setting the order state	11
3.4	Optional: Validation	11
4	Settings / Configuration of Payment Methods	13
4.1	General Information About the Payment Methods	13
4.2	Information on Payment Status	13
4.2.1	Order status "pending" / imminent payment (or similar)	13
4.2.2	Order status "cancelled"	13
4.2.3	Code Adjustments for the MasterPass Integration (xt:commerce SP 2.1)	13
4.2.4	Further information	14
5	The Module in Action	15
5.1	Useful Transaction Information on the Order	15
5.1.1	Refunding Orders	16
5.2	Usage of the Alias Managers / Token Solution	17

5.3	Setup a Cron Job to Activate the Timed Operations	18
6	Testing	19
6.1	Test Data	19
7	Errors and their Solutions	20
7.1	Payment methods are not displayed in the checkout	20
7.2	The Referrer URL appears in my Analytics Tool	20
8	Error Logging	21
8.1	Log Levels	21
8.2	Log Location	21
9	Advanced Information	22
9.1	Transaction Object	22

1 Introduction

This manual describes how to install, configure and use the payment module for Modified Shop and EPRO.

Before you begin with the installation make sure you have all of the necessary data. You will need:

- EPRO API Key (Live/Test)
- Modified Shop Payment Module from sellxed.com/shop
- Entry 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. Strictly follow the steps on the check list when setting up the module. This is the only way to ensure a safe usage in compliance with all security regulations.

1. Before beginning with the installation you will need your API Key which you will receive from your contact.
2. Configuration of the [Main Module](#)
3. Configuration of the accepted [payment methods](#) and the respective authorization methods
4. Conducting a test order

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 EPRO might not get through to the shop.

1.2 System Requirements

In general, the plugin has the same system requirements as Modified Shop. 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.

1.3 Configuration of the Main Module

For the commissioning of the main module you will need to enter your API Key which you receive directly from EPRO.

1.3.1 Configuration of the API Key

For the configuration of the main module you will need to enter your API Key which you receive from your EPRO contact into the backend of your shop system.

Once your tests are done, contact EPRO to activate your account for live transactions. Testcard numbers can be found in the [Test section](#) of this manual.

1.3.2 Configuration of the Notification URL

For a correct handling of the **Transaction Feedback** to your shop, you may please send the notification URL to the responsible person at EPRO via e-mail. The notification URL is listed in the setup form of the payment plugin in your shop.

1.3.3 Configuration of the Scheme

You can specify a scheme for the transaction ID, for multi-store setups this can be useful. You can also define the description text. There are various placeholders which you can change such as the order number of the shop or the internal transaction IDs for which schemes are used.

1.4 Configuration of the Payment Methods

Further information regarding the configuration of the payment methods and the implementation of the different settings can be found [here](#). Once you have activated and configured the payment methods you can run a test. Testcard numbers can be found in the [Test section](#) of this manual.

2 Module Installation and Update in the Modified Shop Shop

2.1 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.

2.2 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.

2.2.1 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.

2.2.2 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.

3 Module Configuration in the Modified Shop 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.

Admin Folder

Please change the Admin Folder back to "admin" if modified during the installation. Alternatively you can use a **not recommended** and **not update proof**solution, and modify our "hard coded" path strings. Please make sure to rename the "admin" folder of the module to exactly the same name on your webserver.

3.1 Configuration of the Main Module

You will find the settings for the module under **"Modules > (Export)-Modules > EPRO Base Module"**. Install the module and click on "Open Method Configuration". A new window should open up. Enter the individual options such as described above in the configuration of the administration interface of EPRO. Should one of the options not be clear to you, you can find further information by moving the cursor to the question mark. The most central settings are explained in more detail in this manual.

3.1.1 Create Pending Orders

You have the possibility of creating pending orders. In this case a temporary order will be created with the status "pending payment". This allows you to reserve the item in your shop and to transmit the identical order number as in the shop to EPRO. In the case when a customer cancels his or her purchase, the order remains in your shop - except if you activate the option "Remove cancelled orders". This makes sure that cancelled orders are deleted regularly. If you do not wish

to create pending orders, simply keep the default setting of xt:Commerce. However, this also means that the reference number (payment ID) with EPRO doesn't correspond to the order number in the shop. You are, however, free to use both options.

3.1.2 Database Encoding

If you haven't selected UTF-8 as a default for your database, this might lead to encoding problems with parameters with special characters. In this case you must select the option that data is UTF-8 encoded before being sent to EPRO.

3.1.3 Order Prefix

With the option order prefix you can define your own order scheme for the transmission to EPRO. This option is especially useful in a multi-store set up so that you can identify which shop a transaction in the back-end of EPRO belongs to. The tag "{id}" is automatically replaced by consecutive order numbers.

3.1.4 Order Prefix

With the option order prefix you can define your own order scheme for the transmission to EPRO. This option helps you identify to which shop a transaction the backend of EPRO is related to. The tag "{id}" will automatically be replaced by the consecutive order number.

3.2 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 EPRO and the module

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

3.3 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.

3.3.1 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.

3.4 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 EPRO during the authorisation process. The payment method is displayed in any case

4 Settings / Configuration of Payment Methods

4.1 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 EPRO 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.

4.2 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 EPRO** or any similar pending status which is implemented by the module.

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

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

4.2.2 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.

4.2.3 Code Adjustments for the MasterPass Integration (xt:commerce SP 2.1)

In order for the button "Pay with MasterPass" to appear in the shopping cart when using external checkout with address transmission, the following adjustments in the code have to be made. If you prefer to simply use MasterPass as a regular payment method without address transmission, no changes are necessary.

The files containing the necessary code adjustments can be found in the zip files in /changes/*.txt. Please paste the codes contained in these files into the location described below.

4.2.3.1 Adjusting shopping_cart.php

Please copy the code snippet from the file /changes/shopping_cart.php.txt in front of the following code:

```
$smarty->assign('language', $_SESSION['language']); (ca. line 140)
```

4.2.3.2 Adjusting templates/[current-active-template]/module/shopping_cart.html

Please copy the code found in /changes/shopping_cart.html.txt in front of the **tag-`{/if}`**:

4.2.4 Further information

MasterPass with External Checkout cannot be used in combination with vouchers. If the customer wants to redeem a voucher, she has to go through standard checkout.

5 The Module in Action

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

Sprache:	german
Zahlungsweise:	XXXXXXXXXX_XXXXXXXX
Kreditkartentyp:	Mastercard (28_23)
Kreditkarteninhaber:	
Kreditkartennummer:	
Sicherheitscode (CVV):	
Kreditkarte läuft ab am:	

Figure 5.1: Transaction Information

5.1 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 EPRO.

Transaktionsnummer

Transaktions ID	12
Transaktionsnummer	9_12
Bestell ID	9
Erstellt am	2013-11-26 18:03:28
Erneuert am	2013-11-26 18:03:59
Kunden ID	1
Zahlungsid	25691316
Zahlungs-Klasse	prepaid_visa
Autorisierungsbetrag	125
Währung	EUR
Zahlungsmethode	Visa
Zahlungsid	25691316
Verbuchter Betrag	125
Transaktion autorisiert	Ja
Transaktion ist unsicher 	Nein
Transaktion verbucht	Ja
Transaktion bezahlt	Ja
Akzeptanz	test123
Kartennummer	XXXXXXXXXXXX0003
Kartenablaufdatum	06/16
Händler Referenz	9_12

Figure 5.1: Transaction Information

5.1.1 Refunding Orders

In order to refund orders, open the transaction information (cf. above). There you will find information on the transaction. Further below you will find an input field in which you can enter the amount you wish to refund. By clicking on "refund", the refund request will be transmitted to EPRO. You no longer need to log into the backend of EPRO.

Zurück

Teilgutschrift

Name	SKU	Typ	MwSt.	Menge	Total (exkl. MwSt.)	
Testartikel	ABC123	product	8 %	1	20.54	2
Selbstabholung (Selbstabholung der Ware in unserer Geschäftsstelle.)	shipping	shipping	0 %	1	0.00	0
					Total gutgeschriebener Betrag:	

Transaktion für weitere Rückvergütungen schliessen

Figure 5.1: refund button

Executing a refund will not change the status of the order.

5.2 Usage of the Alias Managers / Token Solution

With the Alias Manager, your customers can securely save their credit cards with EPRO 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.

ZAHLUNGSWEISE

Bitte wählen Sie die gewünschte Zahlungsweise aus.

Visa

Karten Nr: 9451 1231 0000 0111 Ablaufdatum: 12/2015 CVV: 123

Use previous stored credit
card:

xxxx xxxx xxxx 0111 ▼

Figure 5.1: Alias Manager

Please pay attention!

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

5.3 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.), please activate in a regular request to the file `eprocw_cron.php`. The regular call triggers the appropriate actions.

Here we suggest you use a Cron Engine like for example [EasyCron](#). That way you can This allows you to open the file (URL) with an external service.

6 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.

6.1 Test Data

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

Credit / Debit Card

Test Card Number

Card number 4012 8888 8888 1881

Expiry Date 09/2022

CVC 333

7 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>

7.1 Payment methods are not displayed in the checkout

In case the payment methods are not shown in the checkout even though they have been enabled, then please change in the language file in the respective language folder (/lang/LANG /LANG.php) the following entry (on line nr. 50): `define('LANGUAGE_CURRENCY', 'EUR');`

Set here the default language currency code according to the default language settings in your shop.

7.2 The Referrer URL appears in my Analytics Tool

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

8 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.

8.1 Log Levels

You can configure the log level in the EPRO settings.

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

8.2 Log Location

The log messages are visible in the Modified Shop backend under "**Modules > (Export)-Modules > EPRO Base Module > Log Messages**".

9 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.

9.1 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 Modified Shop initialized in your script, this is required for the database connection.

```
include ('includes/application_top.php');
```

Include the module main file and module classes.

```
require_once DIR_FS_DOCUMENT_ROOT.'admin/includes/EPROCw/init.php';
require_once DIR_FS_DOCUMENT_ROOT.'admin/includes/EPROCw/classes/EPROCw
/Entity/Util.php';
```

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

Load the transaction by Id:

```
$transactionById = EPROCw_Entity_Util::findTransactionByTransactionId
($transactionId);
$transactionObject = $transactionById->getTransactionObject();
```

Load the transaction by external Id:

```
$transactionById = EPROCw_Entity_Util::
findTransactionEntityByTransactionExternalId($externalId);
$transactionObject = $transactionById->getTransactionObject();
```

Load transactions by Order ID:

```
$transactionsByOrderId = EPROCw_Entity_Util::
findTransactionsEntityByOrderId($orderId);
foreach($transactionsByOrderId as $transaction){
    $transactionObject = $transaction->getTransactionObject();
    //Do something with each object
}
```