Installation Manual for BetterPayment JTL

This manual describes the installation and usage of the BetterPayment extension for JTL.

Release Date: Mon, 07 Aug 2023 14:01:06 +0200
Version: 2.0.273

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

####conditional####

Table of Contents

1 Introduction

This manual describes the installation, configuration and use of the payment module forJTL and BetterPayment.

Before you begin with the installation, make sure you have all the necessary data:

- You will receive a document fromBetterPayment containing all the relevant access data which you will need for the SetUp.
- You need theBetterPayment payment module from www.sellxed.com/shop.

In case you don't have a contract with BetterPayment yet, we will gladly provide you with one.

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 Procedure

This document provides you with all the necessary information regarding the installation of the module. You should follow the steps of the checklist carefully during the startup procedure. Only by doing so, the safe use in compliance with all the safety regulations can be ensured.

1. Configuration of the Sandbox and creating the technical accounts.
2. Configuration of the payment methods and the basic settings.
3. Carrying out a test order by means of the enclosed Test data 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 BetterPayment might not get through to the shop.

1.2 System Requirements

In general, the plugin has the same system requirements as JTL. 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.

2 Configuration BetterPayment

2.1 Test and Live Backend to Administrate the Test Transactions

You will find the administration and the display of transactions depending on which mode (live or test mode) via the following links:

- Test-Platform:https://testapi.betterpayment.de/backend.
- Live-Platform:https://api.betterpayment.de/backend.

2.2 Configuration of the Main Module

Navigate to the main module in your shop. Please fill the following data into the corresponding sections. You will receive all of the data in a document in advance directly from BetterPayment:

- API Key
- Shop ID (can be chosen freely, this only serves as an identification of the transaction)
- Incoming Key
- Outgoing Key

Save the data according to live or test mode.

2.3 Activating the payment methods

Navigate to the payment methods in you shop. You can activate the payment methods by pressing a button. Chose between **Payment Page or Iframe** in the authorization method.

Furthermore you can decide whether you want to activate the risk check in the payment methods. The transaction will then be examined based on additional criteria. The risk check is only available for SEPA Direct Debits, Giropay and Sofort. The risk check will be conducted automatically when paying by invoice.

2.4 Transactions Administration

You will find an overview of the transactions, if you log in to the back end. (see top) Here you will find an overview of the transactions made.

You can credit the transactions directly here. But make sure the status is not synchronized with the shop. If you credit a transaction here, you can't credit it at the shop again.
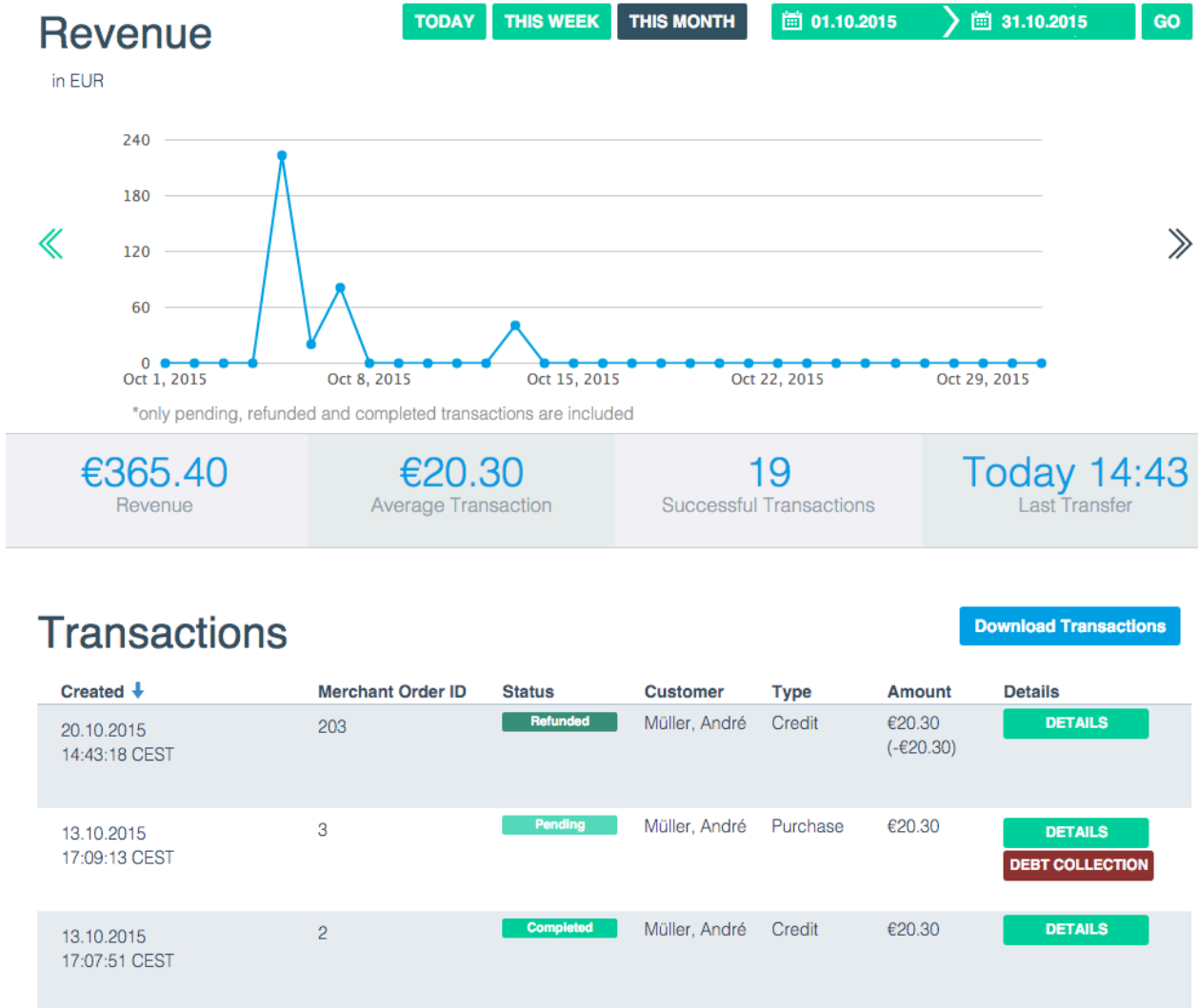
Figure 2.1: transaction overview at the back end of BetterPayment (icon).

2.5 Testing

Next you merely need to activate the payment method and run a test.

When you have activated and configured the payment methods you will run a test by means of the published test data. If all the tests were successful, switch the operating mode in the main module to **Live**.

2.6 Updating the Risk Check Settings

If BetterPayment changes the risk check settings on a payment method, the currently saved configuration in your shop must be synchronized. This may be done in the backend form "Synchronize Risk Settings" by pressing the button "reset". During the next checkout, the new settings will be applied.

3 Module Installation and Update in the JTL 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 JTL 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 Configuration of the Main Module

In a first step the plugin must be installed. The installation of the plugin is carried out under: **Plugins > Plugin administration > BetterPayment** . By clicking on **install**, the plugin is installed in your shop. You can find the settings for the main module under **Plugins > BetterPayment** .

In order to configure the main module, go to **Plugins > BetterPayment** . Change to the tab **Configurations**. Enter the data and settings you have already saved in the back-office of BetterPayment. In case of questions, you can find further information by moving the cursor to the question mark.

In the tab **Information** you can find important information on the plugin such as the version number, release date and callback URL. This data is important when contacting our support.

You have the option of synchronising the order status with the transactions with BetterPayment. This means that when an item is sent or cancelled, you have the possibility of automatically also debiting the payment with BetterPayment. Further information can be found in the section The Module in Action.

4.2 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 **Payment Processing > Payment Methods** . Each payment method is listed individually. Install the payment methods you wish to offer you customers in your shop. 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.

In order for the payment method to be displayed in the shop, you must link them with a shipping method. Open the shipping methods under **Payment Processing > Shipping Methods**. By clicking

edit, you can modify the shipping methods. Here you can link the payment methods you wish to offer and also have the option of setting individual payment fees.

4.2.1 Configuration of payment methods in the WaWi

In order for the JTL WaWi to map the payment methods from the shop, make sure that you add them inside the WaWi Einstellungen > Zahlungsarten verwalten). Make sure that they are spelled exactly as written in the colum "Anmerkung".

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 BetterPayment and the module

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

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 BetterPayment 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 BetterPayment 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 BetterPayment** 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 BetterPayment' 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 BetterPayment are being blocked. In this case check the settings of your firewall and ask the Hoster to activate the IPs and User Agents of BetterPayment.

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 BetterPayment module.

6.1 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 BetterPayment.

Figure 6.1: Transaction Information in the JTL backend

The payment information is also transmitted to your JTL WaWi Client.

Figure 6.1: Transaction Information in JTL WaWi

6.2 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:

6.2.1 JTL Order Confirmation (E-Mail)

Please add the following code block into your JTL email template, if you want to send the payment details of the processor in the mail confirmation sent to the customer:

1. **HTML Mailtemplate**

   {if !empty($Bestellung->cwPaymentInformation)} {$Bestellung->cwPaymentInformation} {/if}

2. **PLAIN Mailtemplate**

   {if !empty($Bestellung->cwPaymentInformation)} {$Bestellung->cwPaymentInformation|replace:"\n"|strip_tags:false} {/if}

6.2.2 JTL Invoice (PDF)

Due to technical limitations, it is currently not possible to display the "payment information" here.

6.2.3 JTL-Backend (Transaction details)

You can view the transaction details in JTL under **Plugins > Plugins > BetterPayment > Transactions**.

6.2.4 JTL Success-Page

Due to technical limitations, it is currently not possible to display the "payment information" here.

6.3 Transaction Management within the JTL Shop

The transaction management can be carried out from within the shop.

6.3.1 Shipment

JTL does not support order status. The WAWI allows the processing of the order (e.g. shipments etc.)
The module does generate a payment receipt based on the transaction. The shipment should be created normally based on the payment receipt.

6.3.2 Capturing of Orders

You can directly debit orders in the backend of your shop. In order to do so, switch to the tab **Transactions**. You can now search for the order number and then click on **Show** in order to find further information about the transaction. If you wish to capture an order, click 'capture'. Here you can modify the amount and the items. By clicking **Capture**, the action will be transmitted to BetterPayment.

In case you do not want to capture all items of the order, you can also close the transaction for further captures.

Figure 6.1: Capturing of Orders in JTL backend

Partial Capturing

Please find out if the capturing of partial amounts is supported by your BetterPayment contract. If this is not the case, it might happen that the transaction is closed for further transactions after a partial capture.

6.3.3 Cancel Orders

In order to cancel an order, open the corresponding transaction. By clicking **Cancel** , a cancellation of the payment occurs with BetterPayment. The reserved amount on the customer's card will be released automatically.

Figure 6.1: Cancelling of Orders in the JTL backend

6.4 Refunds

You can also create refunds for already debited transactions and automatically transmit these to BetterPayment. In order to do so, open the invoice of the already debited order (as described above). By clicking on **Refund** a dialog box for refunds will open up. Insert the amount you wish to refund and then click on **Refund**. The transaction will now be transmitted to BetterPayment.

Figure 6.1: Refunds Directly from Within the JTL backend

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

Executing a refund will not affect the order.

6.5 Setting up Cron Job

In order to activate the time-controlled functions of the plugin (such as update service, deleting pending orders, etc.) set up a time-controlled request in your hosting on the URL, which you will find via Plugins > Plugins >BetterPayment in the tab "Extended Information". Invoking it regularly will trigger the actions.

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

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

Test card

Card number	4111 1111 1111 1111
Expiry Date	03/2027
CID	123

8 Errors and their Solutions

You can find detailed information under http://www.sellxed.com/en/faq. 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 name of the payment method is not shown correctly in the email.

Depending on the name of the payment method and due to the naming convention, the name of the payment method in the email can be slightly strange. You can avoid this by adapting the email template. Go to Content > Email Templates.

In the order email you can replace the following '{$Bestellung->cZahlungsartName}' with '{$Bestellung->Zahlungsart->caption}'. From then on, the name you set in the payment configuration will be shown.

8.2 The Referrer URL appears in my Analytics Tool

When a customer and the notification are redirected via Header Redirection, the BetterPayment 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 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.

9.1 Birthday and gender in JTL

For certain payment service providers it is necessary to check the birthday and the gender of a customer. JTL does not check this by default. These checks can be enabled under "Einstellungen > Kunden / Formulare > Anrede abfragen" and "Einstellungen > Kunden / Formulare > Anrede abfragen" respectively.

10 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.

10.1 Log Levels

You can configure the log level in the BetterPayment settings. The logs are stored according to this setting and are not affected by the shopwide log level settings.

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

10.2 Log Location

The log messages are visible in the JTL backend under the menu item **Admin > Systemlog**.

11 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.

11.1 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 JTL initialized in your script, this is required for the database connection.

```
require_once("includes/globalinclude.php");
```

Include the module main file and modules Transaction class.

```
require_once 'includes/plugins/betterpaymentcw/version/100/init.php';
require_once 'includes/plugins/betterpaymentcw/version/100/classes
/BetterPaymentCw/Entity/Transaction.php';
```

Now you can load the transaction and then extract the transactionObject.
Load the transaction by Id:

```
$transactionById = BetterPaymentCw_Entity_Transaction::loadById
($transactionId);
$transactionObject = $transactionById->getTransactionObject();
```