

# Installation Manual for KBC Shopware

This manual describes the installation and usage of the KBC extension for Shopware.

**Release Date:** Mon, 07 Aug 2023 14:01:06 +0200  
**Version:** 3.0.153

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

E-Mail: [info@sellxed.com](mailto: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

<b>1</b>	<b>Introduction .....</b>	<b>6</b>
1.1	Procedure of the Installation .....	6
1.2	System Requirements .....	7
<b>2</b>	<b>Configuration KBC - Backend .....</b>	<b>8</b>
2.1	Tab: Global Transaction Parameters .....	8
2.1.1	Default operation code and default data capture procedure .....	8
2.1.2	Payment Retry .....	8
2.1.3	Processing for Individual Transactions .....	8
2.2	Tab: Global Security Parameters .....	8
2.2.1	Hashing Method .....	9
2.3	Tab: Payment Page .....	9
2.4	Tab: Data and Origin Verification .....	9
2.4.1	Checks for e-commerce & Alias Gateway .....	10
2.4.2	Checks for KBC DirectLink and KBC Batch (Automatic) .....	10
2.5	Tab: Transaction Feedback .....	10
2.5.1	HTTP Redirection in the Browser .....	10
2.5.2	Direct HTTP-Server-to-Server-Request .....	10
2.5.3	Dynamic e-Commerce Parameters .....	12
2.5.4	Dynamic Parameters .....	12
2.5.5	Security for Request Parameters .....	12
2.5.6	HTTP Request for Status Changes .....	13
2.5.7	Dynamic Parameters .....	13
2.6	(Dynamic-)Template .....	14
2.7	Hidden Authorisation, Ajax Authorization (FlexCheckout - Hosted Tokenization Page), External Checkout & Alias Manager .....	14
2.7.1	Preconditions .....	14
2.7.2	Setting Up an API-User .....	16
2.7.3	Usage of External Checkouts .....	16
<b>3</b>	<b>Module Installation and Update in the Shopware Shop .....</b>	<b>18</b>
3.1	Installation .....	18
3.2	Updates and Upgrades .....	18
3.2.1	Update Checklist .....	18
3.2.2	Update Instructions .....	19
3.2.3	Updated Plugin from the KBCCw to the CwbKBCCw Version. ....	20
<b>4</b>	<b>Module Configuration in the Shopware Shop .....</b>	<b>21</b>
4.1	Configuration of the Main Module .....	21

4.8	Defining the URL for the Transaction Feedback .....	24
4.3	Configuration of the Payment Method .....	21
4.4	Assigning the Payment Method to the Shipping Costs .....	22
4.5	Direct Capturing of Transactions .....	23
4.6	Uncertain Status .....	24
4.6.1	Setting the order state .....	24
4.7	Optional: Validation .....	24
4.8.1	Usage of the Integrated Multishop Functionality of Shopware .....	24
4.8.2	Usage of several independent Shopware shops .....	25
<b>5</b>	<b>Settings / Configuration of Payment Methods .....</b>	<b>26</b>
5.1	General Information About the Payment Methods .....	26
5.2	Information on Payment Status .....	26
5.2.1	Order status "pending" / imminent payment (or similar) .....	26
5.2.2	Order status "cancelled" .....	26
5.3	Usage Hidden Authorisation / Alias Gateway .....	27
5.4	Credit Card .....	27
5.5	PayPal .....	28
5.6	Bank Transfer .....	28
5.7	Usage of the Ajax Authorization (FlexCheckout) .....	29
<b>6</b>	<b>The Module in Action .....</b>	<b>30</b>
6.1	Capturing / Canceling of Orders .....	30
6.1.1	Capturing Orders .....	30
6.1.2	Cancellation .....	30
6.2	Useful Transaction Information on the Order .....	31
6.3	Refunds .....	32
6.4	Usage of the Alias Managers / Token Solution .....	33
6.5	Dynamic Template .....	34
6.5.1	Dynamic Template Exceptions Shopware 5 .....	36
<b>7</b>	<b>Getting the Credit Card Data via Shopware REST API .....</b>	<b>37</b>
<b>8</b>	<b>Set-up a cron job to activate the timed operations .....</b>	<b>38</b>
<b>9</b>	<b>Testing .....</b>	<b>39</b>
9.1	Test Data .....	39
<b>10</b>	<b>Errors and their Solutions .....</b>	<b>40</b>
10.1	The Referrer URL appears in my Analytics Tool .....	40

<b>11</b>	<b>Compatibility with Third-Party Plugins</b> .....	<b>41</b>
11.1	Birthday and gender in Shopware .....	41
<b>12</b>	<b>Error Logging</b> .....	<b>42</b>
12.1	Log Levels .....	42
12.2	Log Location .....	42
<b>13</b>	<b>Advanced Information</b> .....	<b>43</b>
13.1	Transaction Object .....	43

# 1 Introduction

This manual explains the installation, configuration and usage of the payment module for Shopware and KBC. It is important that you strictly follow the described steps. Only by doing so, a secure usage in compliance with all security regulations can be guaranteed.

Before beginning with the installation, please make sure that you are in possession of all necessary data:

- PSPID & Password. You receive these credentials from KBC after conclusion of contract. Our support staff is happy to help you acquiring a KBC contract if you do not yet have one.
- Shopware payment module from [sellxed.com/shop](https://sellxed.com/shop)
- FTP access to your server
- Login credentials for your 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 Procedure of the Installation

1. Configuration of the administration interface of KBC.
2. Upload of the module
3. Configuration of the [basic settings](#) of the [payment module](#)
4. Configuration of the **payment methods**
5. Carrying out a test purchase by means of the attached [test data](#) at the end of this document
6. If the test was successful, copy all data from your test account into the KBC live 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 KBC might not get through to the shop.

## 1.2 System Requirements

In general, the plugin has the same system requirements as Shopware. 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 KBC - Backend

For the configuration, log into the KBC backend:

- Test Environment: <https://secure.paypage.be/Ncol/Test/BackOffice/>
- Live Environment: <https://secure.paypage.be/Ncol/Prod/BackOffice/>

The settings carried out on the following pages are a precondition for the [configuration of the main module](#).

The configuration of the settings is done under **Configuration > Technical Information**. In the following chapter the chapter titles correspond to the tabs in the back office of KBC. For some parts screenshots will be added for a more precise illustration. Should a specific setting not be mentioned then it has no importance for the configuration of the payment module and can be ignored by you as such.

### 2.1 Tab: Global Transaction Parameters

Under the tab Global Transaction Parameters you define the basic settings for the processing of transactions.

#### 2.1.1 Default operation code and default data capture procedure

We recommend choosing **Sale** and **"Automatic data capture by our system at the end of the day"**. All incoming payments will be directly processed and debited. If you are using transaction management and deferred capturing we strongly advise to use these settings. In case you have saved different settings in the module, then the module's settings will overwrite those of the KBC.

#### 2.1.2 Payment Retry

We recommend entering **10**.

#### 2.1.3 Processing for Individual Transactions

We recommend the option **"Online but switch to offline when the online acquiring system is unavailable"**.

### 2.2 Tab: Global Security Parameters

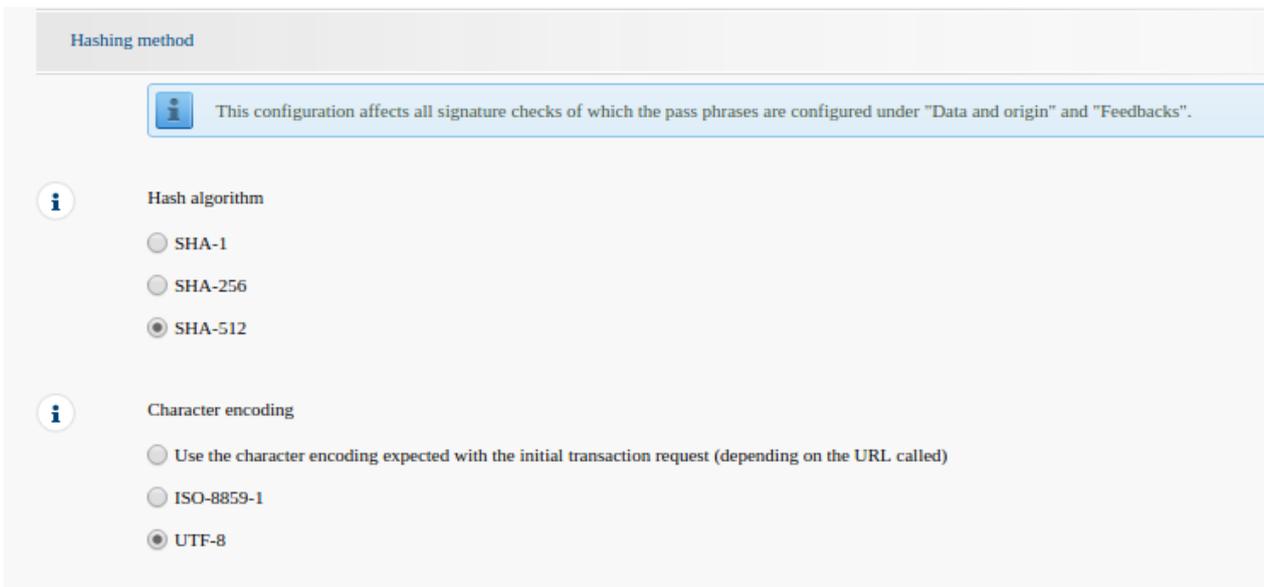
Here you define the security parameters for the interaction between your shop and the gateway of KBC.

## 2.2.1 Hashing Method

The sellxed payment modules support all hashing methods. In order for the module to function, it is crucial that the settings saved with KBC correspond to those made in the backend of your shop.

In order to guarantee a smooth processing of payments, we recommend the following hashing methods:

- Each parameter, followed by the pass phrase. It may be that this parameter is not visible in your shop. In this case it is transmitted automatically.
- Hash Algorithm = SHA 512
- Character Encoding = UTF-8



**Figure 2.1:** Configuration of the SHA-Hash.

These settings must be identical to the settings in the main module (cf. [Configuration of the main module](#)). Please note that the character encoding in our modules is set to UTF-8 per default and cannot be changed.

## 2.3 Tab: Payment Page

The checkbox under **Cancel Button** should be ticked. The field under Back Button Redirection can be left empty.

## 2.4 Tab: Data and Origin Verification

The tab Data and Origin Verification defines security elements concerning the transmission of data and the loading of the payment page.

### 2.4.1 Checks for e-commerce & Alias Gateway

The first box under "URL of the merchant page" can remain empty.

The box "SHA-IN pass phrase" is a security element of the payment verifying its authenticity. Therefore it is important that you use the same pass phrase in the back end and in the shop.

#### Customweb SHA-String Generator

Please note that only ASCII characters may be used for SHA pass phrases. Please use our [SHA pass phrase generator](#) so that you do not enter invalid characters.

### 2.4.2 Checks for KBC DirectLink and KBC Batch (Automatic)

DirectLink allows you to use the Hidden Authorisation (Alias Gateway) as well as the direct debiting of transactions within your shop. If you have activated DirectLink in your account and wish to use this feature, enter **the same** SHA-IN pass phrase in the box "SHA-IN pass phrase" as above.

Further information on the configuration of DirectLink can be found under [Configuration Hidden Authorisation](#).

## 2.5 Tab: Transaction Feedback

In the tab Transaction Feedback the response of KBC to the shop is controlled.

### 2.5.1 HTTP Redirection in the Browser

The parameters in the transaction feedback determine the processing in the shop after payment has been completed. However, the four fields for "HTTP Redirection in the Browser" can remain **empty** because they will be configured through the settings of our module.

**Un-tick** the option "I would like KBC to display a short text to the customer on the secure payment page if a redirection to my website is detected immediately after the payment process."

### 2.5.2 Direct HTTP-Server-to-Server-Request

- The "Timing of the Request" must be set to **Online but switch to a deferred request when the online requests fail**. As "Request Method" you must select **"POST"**.
- In the section **General** make sure that the Option: "I would like KBC to display a "processing" message to the customer during payment processing." is **un-ticked** too.
- Copy here the URL that is listed in the module backend under: **Configuration > KBC > Setup**.

**URL on which the merchant wishes to receive a deferred HTTP request, should the status of a transaction change offline.**

For the **Transaction Feedback** please enter the URL that is generated in the main module. More information can be found under the section [Transaction Feedback](#)

## All orders have the status 'Pending'

Should you notice during the tests that orders are not registered in the shop and that no confirmation email is sent, this might be due to a failure of the transaction feedback. Make sure that your firewall is not blocking any request coming from KBC.

**Direct HTTP server-to-server request**

**i** Timing of the request

No request.  
 Always deferred (not immediately after the payment).  
 Always online (immediately after the payment to allow customisation of the response seen by the customer).  
 Online but switch to a deferred request when the online requests fail.

**URL of the merchant's post-payment page**

If the payment's status is "accepted", "on hold" or "uncertain".

SEE MANUAL FOR THIS URL // URL GEMÄSS MANUAL

If the payment's status is "cancelled by the client" or "too many rejections by the acquirer".

SEE MANUAL FOR THIS URL // URL GEMÄSS MANUAL

Request method

POST  
 GET

**Figure 2.1:** Configuration of the feedback URL for the HTTP-Feedback.

## Deferred Requests und Time Out

KBC sends the notification requests with a time out of 20 sec. Should your shop not respond to the HTTP Request within this time, a time out error will be sent. If you set the option to **Online but switch to a deferred request when the online requests fail** the request will be retrieved four times in ten minute intervals.

Please note that, depending on your KBC subscription, it is possible that the only option available is **Always online (immediately after the payment to allow customization of the response seen by the customer)**. In this case, always select this option.

### 2.5.3 Dynamic e-Commerce Parameters

With this option you control which parameters are sent in the response from KBC. In order for payments to effectively be evaluated in your shop, all necessary parameters must be sent. It is possible that you have other options available to select. Just make sure that you move every parameter from **available** to **selected** except for the following:

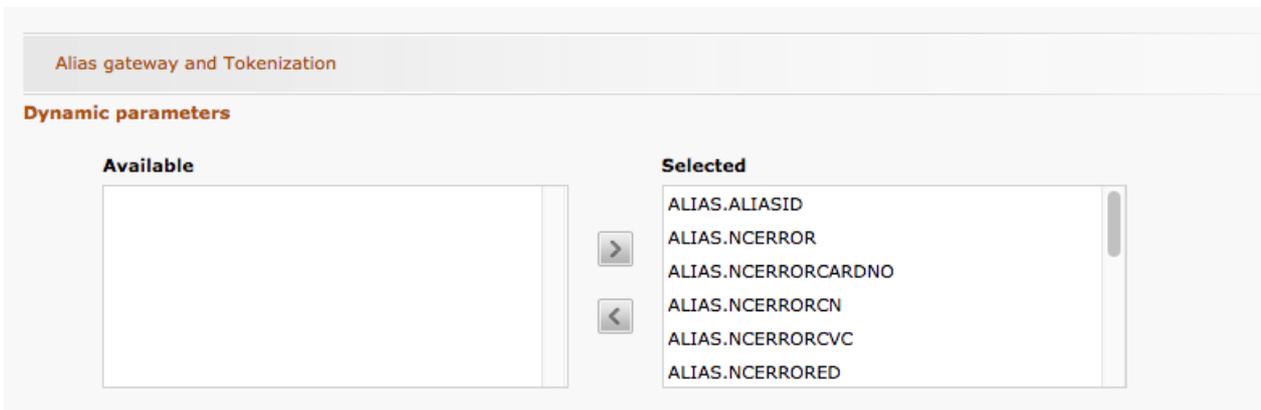
- PM
- CN
- SUBBRAND
- AAVADDRESS
- AAVCHECK
- AAVMAIL
- AAVNAME
- AAVPHONE
- AAVZIP



**Figure 2.1:** Configuration of the Dynamic E-Commerce Parameters. Your selection may look differently depending on your subscription with KBC.

### 2.5.4 Dynamic Parameters

With this option you control which parameters are sent in the response from KBC. In order for payments to effectively be evaluated in your shop, all parameters must be sent. It is possible that you have other options available to select. Just make sure that you move **every** parameters from **available** to **selected**.



**Figure 2.1:** Configuration of the Dynamic Parameters. Your selection may look differently depending on your subscription with KBC.

## 2.5.5 Security for Request Parameters

The box "SHA-OUT pass phrase" is an additional safety mechanism for payments. Here too must the entered pass phrase be identical to the one in your shop. However, please make sure that your SHA-OUT pass phrase differs from the SHA-IN pass phrase. .

### Customweb SHA-String Generator

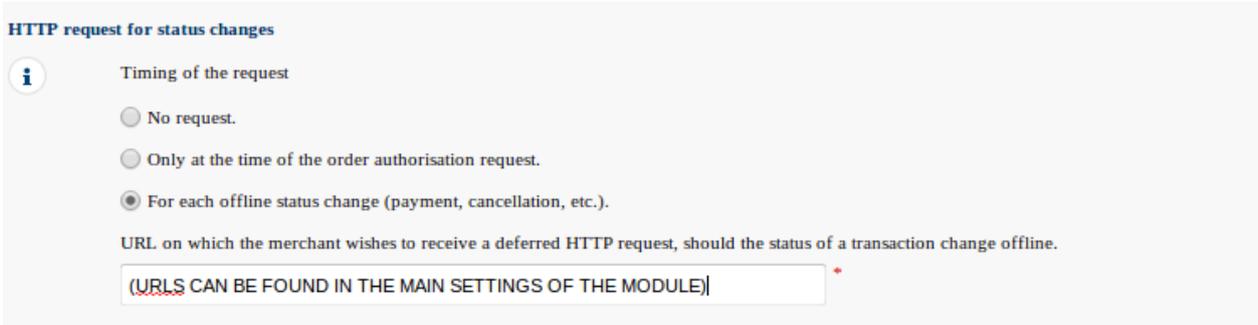
Please note that only ASCII characters may be used for SHA pass phrases. Please use our [SHA pass phrase generator](#) so that you do not enter invalid characters.

## 2.5.6 HTTP Request for Status Changes

You have the option to be informed about status changes by KBC directly. This feature is especially useful when a payment has an uncertain status because the acquirer system could not be reached. The payment status will be updated automatically. Furthermore this feature allows you to capture payments inside KBC and have synchronize them with your shop

In order to use this feature you will have to select the option **For each offline status change (payment, cancellation, etc.)**. Afterwards just copy the URL into the field that is displayed in the Setup of your Main configuration: **Configuration > KBC > Setup**.

Enter the URL shown there into the field 'URL on which the merchant wishes to receive a deferred HTTP request, should the status of a transaction change offline.'



**HTTP request for status changes**

**i** Timing of the request

- No request.
- Only at the time of the order authorisation request.
- For each offline status change (payment, cancellation, etc.).

URL on which the merchant wishes to receive a deferred HTTP request, should the status of a transaction change offline.

**Figure 2.1:** Enter the URL from the Setup of the Main Configuration.

## 2.5.7 Dynamic Parameters

Move all parameters from **available** to **selected**.

## 2.6 (Dynamic-)Template

You can now find the option template via **Configuration > Template**. The Dynamic Template enables you to design the payment mask for the credit card data so it matches your web shop. Should the field for the dynamic template not be visible, please contact KBC. It may be that you have to activate additional options in your account.

For the usage of the dynamic template, please save the following settings:

- Enable JavaScript on template: **'No'**
- Our modules do not allow "usage of static template". So click 'No'
- Allow usage of dynamic template: **'Yes'**
- In case you are using the dynamic template, enter the host name, i.e. `https://Your-Shop.com` in the lower box. The upper one must remain empty.

**Figure 2.1:** Configuration of the Dynamic Template.

### Dynamic Template

Please note that you require an SSL certificate for the usage of the dynamic template. Otherwise the CSS-Resources will not be reloaded in certain browsers which might lead to layout issues.

The activation of the dynamic template in the shop is explained [here](#).

## 2.7 Hidden Authorisation, Ajax Authorization (FlexCheckout - Hosted Tokenization Page), External Checkout & Alias Manager

## 2.7.1 Preconditions

For the usage of the Hidden Authorization, Ajax Authorization (FlexCheckout), Alias Manager or External Checkout during the payment processing, certain adjustments are necessary in the backend of KBC. Please check first if the options are activated in your account under **Account > Your Options**.

Other options

Available options **Premium** Default options

ID	Option	Activation	Monthly fee	Transaction/Unit cost	Status	
AFTP	AFTP Module	0 CHF	0 CHF	0 CHF	Active	Deactivate
D3D	D3D (Direct Link 3D)	0 CHF	0 CHF	0 CHF	Active	Deactivate
ELVDE	ELV (Direct Debits Germany)	0 CHF	0 CHF	0 CHF	Active	Deactivate <a href="#">Configuration</a>
FAQQ	Foreign acquirers	0 CHF	0 CHF	0 CHF	Active	Deactivate
INV	Invoice appearance	0 CHF	0 CHF	0 CHF	Active	Deactivate
NPMA2	Foreign payment methods (fees per payment method activated) - CH	0 CHF	0 CHF	0 CHF	Active	Deactivate
OPC	One Page Checkout	0 EUR	0 EUR	0 EUR	Active	Deactivate
RECO1	Reconciliation Module	0 CHF	0 CHF	0 CHF	Active	Deactivate
RECX	Alias Manager	0 CHF	0 CHF	0 CHF	Active	Deactivate <a href="#">Configuration</a>
SCH	Scheduled Payments	0 CHF	0 CHF	0 CHF	Available	Activate
SUB	Subscription manager	0 CHF	0 CHF	0 CHF	Active	Deactivate
USR4	User Manager up to 20 users	0 CHF	0 CHF	0 CHF	Active	Deactivate <a href="#">Configuration</a>
USR5	User Manager up to 50 users	0 CHF	0 CHF	0 CHF	Available	Activate
USR6	User Manager up to 100 users	0 CHF	0 CHF	0 CHF	Available	Activate
USR7	User Manager up to 200 users	0 CHF	0 CHF	0 CHF	Available	Activate

(1) CAP1 : Includes transaction fee for 3-D Secure transactions

**Figure 2.1:** Make sure to activate the OPC and DPR options.

If an option is not available please contact KBC.

### Options and their Meanings:

- For the usage of the Hidden Authorization you require the option **OnePageCheckout (OPC)**.
- For the usage of the Ajax Authorization (FlexCheckout) you require the option **OnePageCheckout (OPC)** and **DirectLink (DPR)**.
- If you additionally want to use the Alias Manager, you also need the option **Alias Manager (RECX)**.

In case these options are not be available, please contact KBC directly.

## 2.7.2 Setting Up an API-User

If you want to use the features of the transaction management or the hidden authorization (Alias Gateway) you will need to set up an API user as described hereafter:

Go to **Configuration > User Administration** and click on 'New User'. Choose a **User ID** and insert a user name of your choice and your e-mail address. Select **Administrator without user manager** as a profile. It is furthermore important that you select the option **Special user for API (no access to the administration)**. If you do not see this option, you will need to let KBC activate the option DirectLink for your account.

### Password of the API-User

The User ID and the API-user password may only contain letters and numbers as well as special characters. The following special characters are not allowed: & (Ampersand), " (double quotes), ' (single quotes)

Note down the access data. They will have to be inserted in the respective place in the [configuration of the main module](#).

## 2.7.3 Usage of External Checkouts

For the usage of external checkouts there are additional options required. External checkouts, such as MasterPass are payment methods, which offer the possibility to transfer addresses automatically.

The usage of the external checkouts asks for the option Direct Link and for the activation of payment method in your account.

For the performance of the payment method, you have to make sure that under Technical Information > Transactionfeedback in the section Direct Link all parameters selected are shifted

The screenshot shows a web interface for configuring 'Directlink'. At the top, there is a tab labeled 'Directlink'. Below it, the section 'Dynamic parameters' is visible. A blue information box contains the text: 'This exchange box allows you to select or deselect the parameters you wish to receive as feedback for a transaction and such and cannot be removed.' Below this, there are two columns: 'Available' and 'Selected'. The 'Selected' column contains a list of parameters: AAVADDRESS, AAVCHECK, AAVMAIL, AAVNAME, AAVPHONE, and AAVZIP. Between the two columns are two arrow buttons: a right-pointing arrow and a left-pointing arrow. At the bottom left of the interface, there is a 'SAVE' button.

**Figure 2.1:** Response Parameter for Direct Link.

## 3 Module Installation and Update in the Shopware 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](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.

### 3.2.3 Updated Plugin from the KBCCw to the CwbKBCCw Version.

We plan to release this plugin in the shopware store in the near future. To make this possible we had to apply some changes to our code. Most important a prefix for our PHP classes is required and the plugin folder had to be renamed. To update to this new version simply overwriting the existing files is not possible. This section describes the steps required to update the module in detail.

#### Do not do updates directly in the live environment

Please test the update procedure first in your development system. So you can ensure the update is working correctly.

#### Multiple Plugins

If you are using multiple plugins from us. You need to update all of them to the new version.

1. Activate the maintenance mode of the shop.
2. Make a backup for your database and your files in your shop. **Without this you will not be able to revert to the old version, if anything goes wrong.**
3. Upload the new plugin files.
4. Deactivate the existing plugin. **Do not deinstall the old plugin yet.**
5. Install and activate the new plugin.
6. If you have custom CSS or template files for the plugin, adapt the changes to the new plugin.
7. If you are using the Shopware API endpoint of this plugin, update the URL to new endpoint.
8. Verify the main and method settings.
9. Deinstall the old plugin.
10. Delete the files of the old plugin. The following files/folders must be removed, if they exist:
  - engine/Shopware/Plugins/Local/Frontend/KBCCw
  - engine/Library/Customweb
  - engine/Library/Crypt
  - engine/Library/File
  - engine/Library/Math
  - engine/Library/Mobile
  - engine/Library/Net
  - engine/Library/PhoneNumber
  - engine/Library/System
  - engine/Library/loader.php
  - engine/Library/openssl.cnf
11. Make a test transaction to verify everything is working correctly.
12. Deactivate the maintenance mode of the shop.

## 4 Module Configuration in the Shopware 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 order to install the main module search for KBC under **Settings > Plugin Manager**. Click on the + symbol in order to begin the installation process. Afterwards, click the "edit" symbol and configure the main module.

Enter the individual options which you partly may have already used for the configuration of the administration interface of KBC. Should an option not be clear to you, you can click on the question mark and will find further information on the settings.

### 4.2 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 KBC configuration. The URL to be entered can be found in your Shopware Shop under: **Configuration > KBC > Setup > Setup**

### 4.3 Configuration of the Payment Method

After having configured the main module, you will find all payment methods under **Settings > Payment Methods**. Activate each payment method you wish to offer in your shop. You can save individual settings for each payment method (and for each shop in case of a multi-shop set-up) and thereby optimally adapt the payment methods to your existing processes. The most central options are described in more detail further below.

Hauptshop Deutsch

**Autorisierungsmethode:** Hidden Authorization (Authorization Interface) ?

**Authorized Status:** Komplett bezahlt ?

**Uncertain Status:** Der Kredit wurde vorlaeufig akzeptiert. ?

**Cancelled Status:** Vorgang wurde abgebrochen. ?

**Captured Status:** Don't change order status ?

**Kundenadresse:** Keine Adresse senden ?

**Capturing:** Directly after order ?

**Minimaler Gesamtbetrag:** ?

**Maximaler Gesamtbetrag:** ?

**New Order Status:** Offen ?

**Validation:** Nach dem Wählen der Zahlungsart ?

Figure 4.1: Configuration of the Payment Method

## 4.4 Assigning the Payment Method to the Shipping Costs

In order for the transactions and the shipping costs to be matched correctly to the payment methods you need to assign the payment methods to the shipping methods. Go to settings > shipping costs. Open **all** the shipping costs and add a payment method.

**Versandkosten**

Konfiguration

Name: Standard Versand

Beschreibung: Beschreibung

Tracking-URL:

Kommentar:

Sortierung: 0

Aktiv:

Shop: Alle Shops

Kundengruppe: Alle Gruppen

Versandkosten-Berechnung nach: Gewicht

Versandkosten frei ab: Nie

Versandart-Typ: Standard Versandart

Zahlungsart-Aufschlag: Als eigene Warenkorb-Position

Steuersatz: Höchster Steuersatz

Verfügbare Zahlungsmethoden:

Verfügbar	Ausgewählt
SEPA	Debit
	On delivery
	Invoice
	Prepayment
	Test Payment Method

Abbrechen Speichern

**Figure 4.1:** Assigning the Payment Method to the Shipping Method.

### Attributing Payment Methods to the Shipping Method

Depending on the amount of delivered payment methods, it may be that certain payment methods can't be attributed to a specific shipping method as not all payment methods can be displayed. This is a known bug of Shopware. Merely the first 30 payment methods are displayed. Please delete all payment methods that you don't need and then all other payments will be displayed.

## 4.5 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 KBC and the module

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

## 4.6 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.

### 4.6.1 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.

## 4.7 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 KBC during the authorisation process. The payment method is displayed in any case

## 4.8 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 KBC configuration. The URL to be entered can be found in your Shopware Shop under: **Configuration > KBC > Setup**

### 4.8.1 Usage of the Integrated Multishop Functionality of Shopware

The payment module supports the multishop feature of Shopware. No further modifications are necessary. The module automatically recognizes the shop the order belongs to. In order for the multishop functionality to work, it is, however, necessary that the individual sub-shops within Shopware have been configured correctly.

### 4.8.2 Usage of several independent Shopware shops

In case you wish to connect several independent shops to one contract with one contract with KBC, you require a sellxed payment module for each shop as well as the corresponding multi-shop module which you will find directly by the payment module.

1. Extract the multi-shop module and open the file "router.php"
2. Enter a shop ID of your choice in the box "Shop ID" within the main configuration of the module (e.g. shop\_a and shop\_b)
3. Enter the following line into the corresponding place within the file router.php of the multi-shop module (for each shop shop\_a and shop\_b):  
**`$redirect['shop_a'] = 'https://www.your-shop.com/shopware.php/sViewport,KBCCwProcess/sAction,process';`**
4. Enter the URL below (instead of the URL mentioned above) under the point "Feedbacks" in the backend of KBC: **`http[s]://yourshop.com/router.php`**
5. Copy the file router.php into the main directory of your shop

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

#### 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.

### 5.3 Usage Hidden Authorisation / Alias Gateway

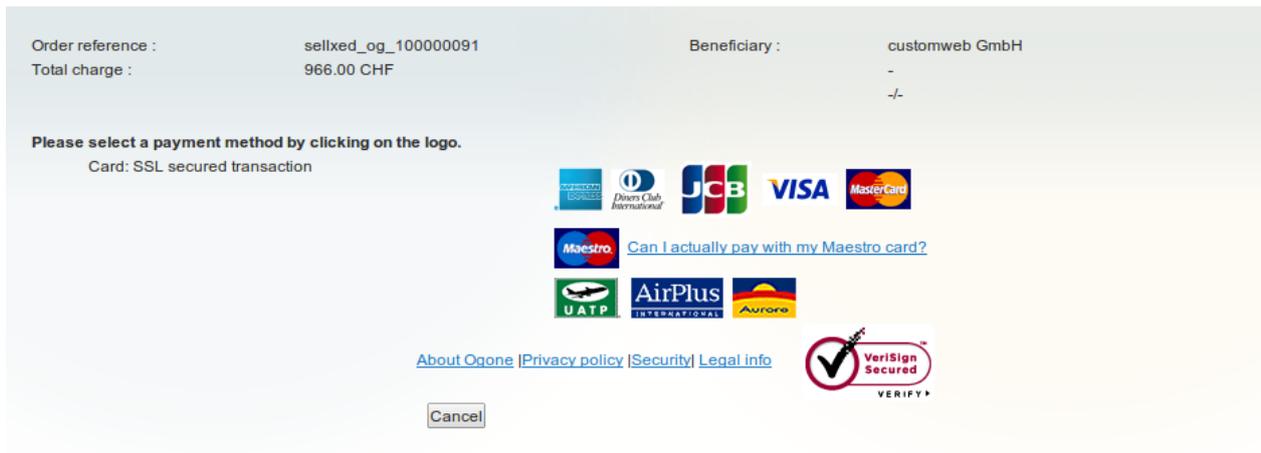
You have the possibility of processing credit cards without redirection to the payment page of KBC.

In order to be able to use this feature, you must first carry out the steps listed under [Hidden Authorisation / Alias Gateway](#) and create an API-User.

You can then activate the authorisation method 'Hidden Authorisation' for the payment methods of your choice. This procedure requires you as a merchant to fulfill the PCI-SAQ-A-EP. At no time is any credit card data saved on your server. Contact your Acquirer for more information about the SAQ and the requirements.

### 5.4 Credit Card

For the integration you have the choice between the pre-selection in the shop; i.e. the customer chooses the card type already in the shop; or a generic integration in which case the customer only sees the option 'credit card' as a payment method and then chooses the card type on the payment page KBC.



**Figure 5.1:** No pre-selection of the card type - the type will then be chosen on the payment page.

## Pre-Selection of the Payment Method

We recommend choosing the option pre-selection. This allows you to configure individual settings for each payment method and add payment fees to the various methods.

## 5.5 PayPal

You can process PayPal directly via KBC. In order to do so, however, some adjustments to your PayPal account are required. Please carry out the following steps:

1. Activate the payment module in your shop
2. Make sure that the payment method is activated in your KBC account
3. The API adjustments must be carried out in your PayPal account. The various steps are explained in the "PayPal Express Checkout Integration Guide". This document can be found in your PSP account under: Support > Integration and User Guides > Technical Guides > KBC PayPal Express Checkout Integration Guide .

## 5.6 Bank Transfer

In order to use the payment method Bank Transfer, make sure that you configured the settings for the [HTTP status changes](#) as described above. Payments done via the payment method bank transfer will first be marked as pending. As soon as the client pays the invoice, KBC sends us the notification for the status change and we will update the state of the payment method.

## 5.7 Usage of the Ajax Authorization (FlexCheckout)

You have the possibility of processing credit cards without redirection to the payment page of KBC.

In order to be able to use this feature, you must first carry out the steps listed under [Hidden Authorisation, FlexCheckout \(Hosted Tokenization Page\), External Checkout und Alias Manager](#) and create an API-User.

You can then activate the authorisation method 'Ajax Authorization (FlexCheckout)' for the payment methods of your choice. At no time is any credit card data saved on your server.

## 6 The Module in Action

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

### 6.1 Capturing / Canceling of Orders

#### 6.1.1 Capturing Orders

##### Please note

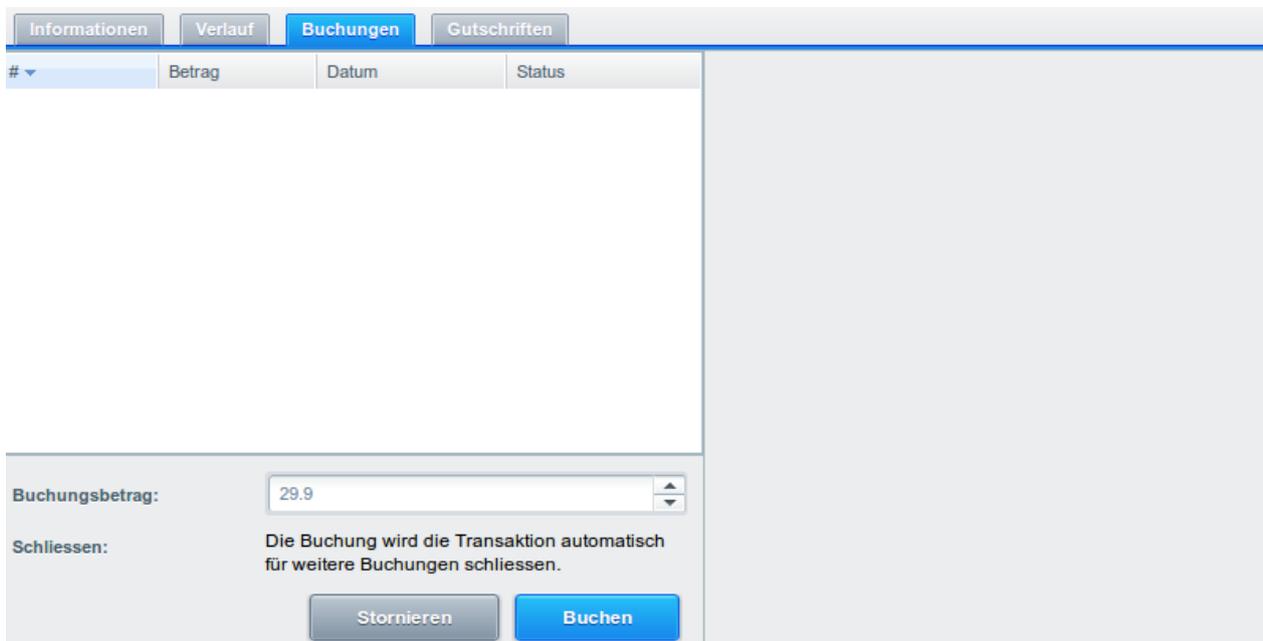
The transaction management between your shop and KBC is not synchronized. If you capture payments with KBC, the status in the shop will not be updated and a second capturing in the shop is not possible.

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

Open the order you wish to capture and switch to the tab **Transactions**. Here you will find the tab **Captures**. Enter the amount you wish to debit from the customer's card. By clicking **Capture**, a direct capturing of the order occurs with KBC.

#### 6.1.2 Cancellation

By clicking the button **Cancel**, you delete the initial authorization on the customer's card. The pre-authorized amount is released from the customer's card.



The screenshot shows the 'Buchungen' (Bookings) tab in the KBC module interface. At the top, there are four tabs: 'Informationen', 'Verlauf', 'Buchungen' (selected), and 'Gutschriften'. Below the tabs is a table with columns: '#', 'Betrag', 'Datum', and 'Status'. The table is currently empty. Below the table, there is a form for capturing a booking. It includes a label 'Buchungsbetrag:' followed by a text input field containing '29.9'. Below this is a label 'Schliessen:' followed by a text box containing the message: 'Die Buchung wird die Transaktion automatisch für weitere Buchungen schliessen.' At the bottom of the form are two buttons: 'Stornieren' (grey) and 'Buchung' (blue).

Figure 6.1: Capturing/Cancellation of orders in Shopware

## 6.2 Useful Transaction Information on the Order

In each order, processed via our module, you can find an overview of the most important information about the transaction as well as a transaction history.

In order to get to the transaction information, open the order and switch to the tab **Transactions**. Here you will find all authorization attempts and the related information.

x
Bestellungs-Details: 20006

Übersicht
Details
Kommunikation
Positionen
Dokumente
Status History
Transaktionen

#	Datum	Zahlungsmethode	Ist Autorisiert	Betrag
19	20.11.2013 11:06:11	Visa	aktiv	29,90

Seite 1 von 1
Anzeige Eintrag 1 - 1 von 1

Informationen
Verlauf
Buchungen
Gutschriften

Autorisierungsbetrag:	29.9	Transaktion verbucht:	Ja
Währung:	EUR	Transaktion bezahlt:	Ja
Zahlungsmethode:	VISA	Kartenummer:	xxxx xxxx xxxx 0111
Zahlungsid:	80C26IAU017rUAj3v7fQAn6fQnGb	Kartenablaufdatum:	10/24
Verbuchter Betrag:	29.9	Card Type:	VISA
Transaktion autorisiert:	Ja	3D Secure:	Erfolgreich
Transaktion ist unsicher:	Nein		

Figure 6.1: Transaction Information in Shopware.

## 6.3 Refunds

You can refund already captured transactions and automatically transmit them to KBC. In order to do so, open the order. Switch to the tab **Transactions**. A new window will open up in which you must then switch to the tab **Refunds** in order to refund orders.

To refund a line item you need to enter the quantity you wish to refund. By clicking **Refund**, the refund is automatically transmitted to KBC. It will then be refunded via the payment method the customer chose for the transaction.

Transaktionsdetails: 23
X

Information
Verlauf
Verbuchungen
Gutschrift

#	Betrag	Datum	Status

Name	SKU	Typ	MwSt.	Menge	Total (exkl. MwSt.)	Total (inkl. MwSt.)
ELASTIC MÜTZE	SW10153	product	19.00 %	1	12.56	14.95
Standard Versand	shipping	shipping	19.00 %	1	3.28	3.9
<b>Total gutgeschriebener Betrag</b>						<b>18.85</b>

Transaktion für weitere Rückvergütungen schliessen
 Rückerstattung

**Figure 6.1:** Refunding Orders in Shopware.

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

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

## 6.4 Usage of the Alias Managers / Token Solution

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

## Zahlungs-Formular

**Saved cards:**  ▼  
 You may choose one of the cards you paid before on this site.

**Karteninhaber\***   
 Bitte geben Sie den Karteninhaber auf Ihrer Kreditkarte ein.

**Kartennummer\***   
 Bitte geben Sie hier die Kartennummer Ihrer Kreditkarte ein.

**Karten Ablaufdatum\***  ▼  ▼  
 Wählen Sie das Ablaufdatum Ihrer Karte aus.

**CVC Code\***   
 Bitte geben Sie hier den CVC Ihrer Karte ein. Sie finden diesen auf der Rückseite Ihrer Kreditkarte.

Figure 6.1: Alias Manager Usage in Shopware

### Please note!

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

## 6.5 Dynamic Template

We provide a standard skin for the usage of the dynamic template for all sellxed modules. You can activate the option by setting the option "Dynamic Template" to "Use Shop Template" in the main module. Please make sure that all settings specified under [Template](#) have been carried out in the backend of KBC.

The Dynamic Template allows you to load the Payment Page of KBC in the layout of your shop. The customer, therefore, does not notice that the customer has been forwarded to the website of KBC. The template dynamically adjusts to your shop template. This, furthermore, allows you to easily integrate Mobile Skins (cf. Screenshots).

### The Dynamic Template requires an SSL Certificate

In order for the Dynamic Template to work correctly with all browsers, you require an SSL certificate in your shop.

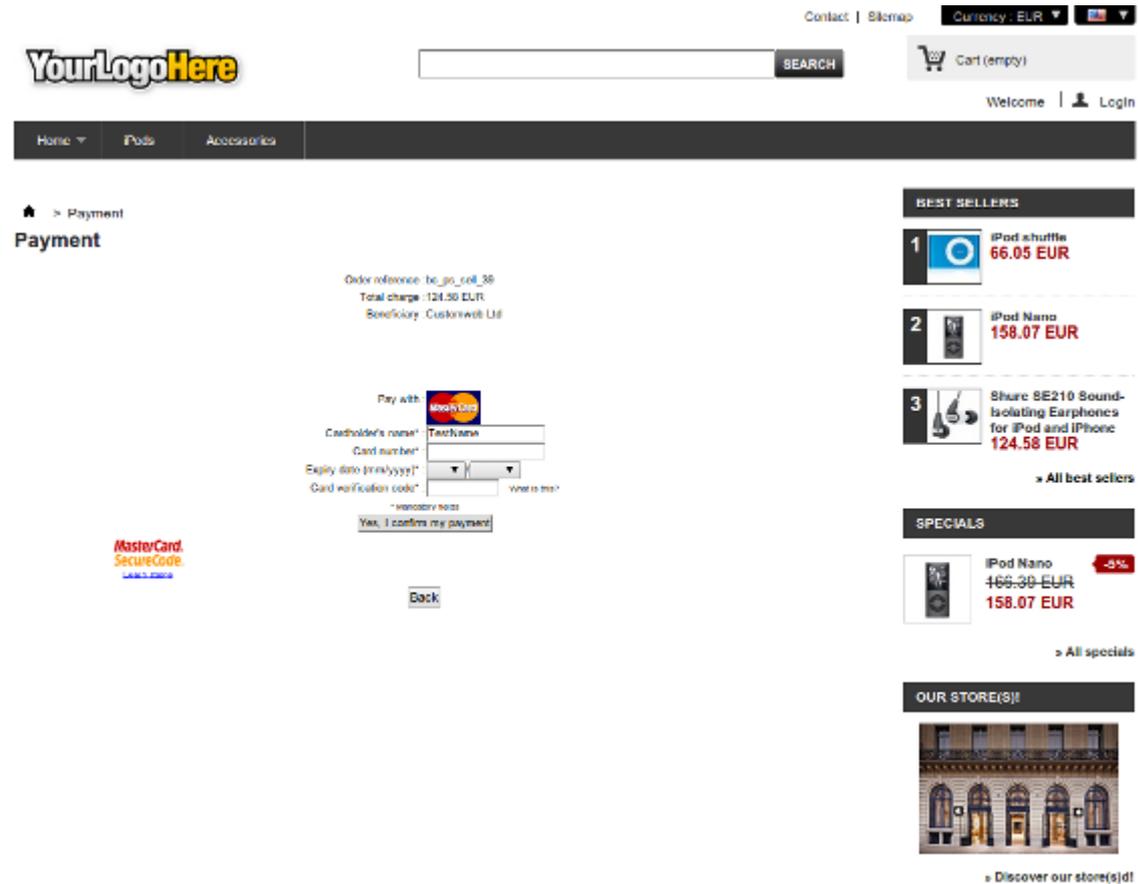


Figure 6.1: Example: The skin adapts to the layout of your shop.

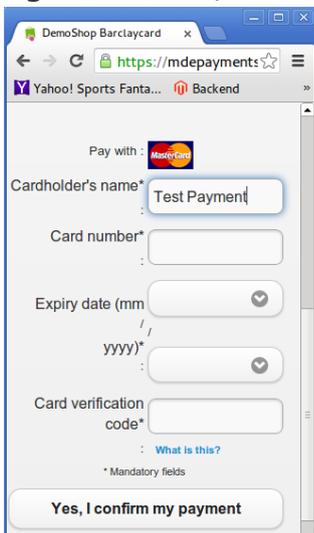


Figure 6.1: Example: Dynamic Template with mobile skin

### 6.5.1 Dynamic Template Exceptions Shopware 5

Shopware Version 5 incorporates icons through an additional font. Hence, icons may not be displayed correctly on the payment page. To prevent this, please make the following changes to your server configuration:

Insert the .htaccess files on your server into the modification below. The file's location depends on the folder structure of your theme. For standard templates, the relevant files can be found in the following locations:

- /themes/Frontend/Responsive/frontend/\_public/vendors/fonts/open-sans-fontface/.htaccess
- /themes/Frontend/Responsive/frontend/\_public/src/fonts/.htaccess

```
<IfModule mod_headers.c> Header set Access-Control-Allow-Origin "*" </IfModule>
```

## 7 Getting the Credit Card Data via Shopware REST API

When connecting ERP systems to Shopware it may be that further information is required via Shopware API. Our module provides you with a lot of payment information also via Shopware REST API.

Basic Information about the usage of the REST API of Shopware can be found in the [Shopware Developers Guide](#).

The service can be called up under <https://www.shop-url.com/api/CwbkbccwTransactions>. The call of the above URL leads to the listing of all transactions. By adding the Transaction ID you can filter by the corresponding Transaction ID. The Transaction ID can be found in the order overview (Customer > Payments > KBC )

Additionally, you can also access the transaction via the Payment ID (transmitted through KBC). To do so, the parameter 'usePaymentId=1' has to be added as a parameter to the URL.

## 8 Set-up 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 Shopware Cron engine. Especially the update function allows you to automatically retrieve additional information or changes of your order directly via the API of KBC. For payment methods which use a delayed authorisation process this is mandatory. Please note it could be that in order to use the update feature it may be necessary that KBC activates additional options in your account.

The module uses the standard cron engine of Shopware. More information regarding the set up can be found here: [https://community.shopware.com/Cronjobs\\_detail\\_1102.html](https://community.shopware.com/Cronjobs_detail_1102.html).

## 9 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.

### 9.1 Test Data

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

#### Credit / Debit Card

Card number	3750 0000 000 0007	American Express
Expiry Date	12/2020	
CVC	1234	
Card number	5399 9999 9999 9999	Mastercard
Expiry Date	12/2020	
CVC	123	
Card number	4111 1111 1111 1111	VISA (Without 3-D Secure Check)
Expiry Date	12/2020	
CVC	123	
Card number	4000 0000 0000 0002	VISA (With 3-D Secure Check)
Expiry Date	12/2020	
CVC	123	

#### MasterCard

Card number	5399 9999 9999 9999	Mastercard
Expiry Date	12/2020	
CVC	123	

#### Visa

Card number	4000 0000 0000 0002	VISA (With 3-D Secure Check)
Expiry Date	12/2020	
CVC	123	
Card number	4111 1111 1111 1111	VISA (Without 3-D Secure Check)
Expiry Date	12/2020	
CVC	123	

## 10 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>

### 10.1 The Referrer URL appears in my Analytics Tool

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

# 11 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.

## 11.1 Birthday and gender in Shopware

For certain payment service providers it is necessary to check the birthday and the gender of a customer. Shopware does not check this by default. To enable these checks go to "Configuration > Basic Settings > Frontend > Login / Registration" and check the following parameters:

- Show Birthday Field
- Birthday is required
- Show "I am" select field

## 12 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.

### 12.1 Log Levels

You can configure the log level in the KBC settings.

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

### 12.2 Log Location

The log file is stored in the Shopware log directory. Please make sure the path exists and it is writable by the webserver. (Default Path: {shopRootDirectory}/logs or {shopRootDirectory}/var/log)

## 13 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.

### 13.1 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 Shopware initialized in your script, this is required for the database connection.

```
require __DIR__ . '/autoload.php';
use Shopware\Kernel;
$environment = 'production';
$kernel = new Kernel($environment, false);
$kernel->boot();
```

Register the module classes.

```
$pluginPath = 'engine/Shopware/Plugins/Local/Frontend/CwbKBCCw';
Shopware()->Loader()->registerNamespace('Customweb', 'Customweb/');
Shopware()->Loader()->registerNamespace('CwbKBCCw_Helpers', $pluginPath .
'/Helpers/');
Shopware()->Loader()->registerNamespace('CwbKBCCw_Entities', $pluginPath
. '/Entities/');
Shopware()->Loader()->registerNamespace('CwbKBCCw_Components',
$pluginPath . '/Components/');
Shopware()->Loader()->registerNamespace('Shopware\Components\Api',
$pluginPath . '/Components/Api/');
```

```
require_once $pluginPath . '/Library/loader.php';
Customweb_Core_Util_Class::registerClassLoader(function ($className){
    return Shopware()->Loader()->loadClass($className);
});
```

To load old transactions register a legacy class loader.

```
Customweb_Core_Util_Class::registerClassLoader(function($className){
    if(strpos($className, 'KBCCw') === 0){
        $fileName = substr($className, strpos($className, '
if(file_exists('engine/Shopware/Plugins/Local/Fronte
```

```
                require_once 'engine/Shopware/Plugins/Local/  
                return true;  
            }  
        }  
        return false;  
    });
```

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

Load the transaction by Id:

```
$transactionById = CwbKBCCw_Helpers_Util::loadTransaction  
($transactionId);  
$transactionObject = $transactionById->getTransactionObject();
```