

# Installation Manual for Dimoco Shopware

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

**Release Date:** Mon, 07 Aug 2023 14:01:06 +0200  
**Version:** 2.0.164

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>5</b>
1.1	Process of the Installation .....	5
1.2	System Requirements .....	6
<b>2</b>	<b>Configuration of the Payment Module and the Payment Methods .....</b>	<b>7</b>
2.1	Live and Test Credentials .....	8
2.1.1	Live Credentials .....	8
2.1.2	Test Credentials .....	8
2.2	Payment Method Specific Configurations .....	9
2.2.1	Credit Card 3D Secure .....	9
2.2.2	SEPA Direct Debits Mandates .....	9
2.3	Integrated Payment Form (Ajax Authorization) .....	9
<b>3</b>	<b>Module Installation and Update in the Shopware Shop .....</b>	<b>10</b>
3.1	Installation .....	10
3.2	Updates and Upgrades .....	10
3.2.1	Update Checklist .....	10
3.2.2	Update Instructions .....	11
3.2.3	Updated Plugin from the DimocoCw to the CwbDimocoCw Version. ....	12
<b>4</b>	<b>Module Configuration in the Shopware Shop .....</b>	<b>13</b>
4.1	Configuration of the Main Module .....	13
4.2	Configuration of the Payment Method .....	13
4.3	Assigning the Payment Method to the Shipping Costs .....	14
4.4	Direct Capturing of Transactions .....	15
4.5	Uncertain Status .....	16
4.5.1	Setting the order state .....	16
4.6	Optional: Validation .....	16
4.6.1	Usage of the Integrated Multishop Functionality of Shopware .....	17
<b>5</b>	<b>Settings / Configuration of Payment Methods .....</b>	<b>18</b>
5.1	General Information About the Payment Methods .....	18
5.2	Information on Payment Status .....	18
5.2.1	Order status "pending" / imminent payment (or similar) .....	18
5.2.2	Order status "cancelled" .....	18
<b>6</b>	<b>The Module in Action .....</b>	<b>19</b>
6.1	Capturing / Canceling of Orders .....	19

6.1.1	Capturing Orders .....	19
6.1.2	Cancellation .....	19
6.2	Useful Transaction Information on the Order .....	20
6.3	Refunds .....	21
6.4	Usage of the Alias Managers / Token Solution .....	22
<b>7</b>	<b>Getting the Credit Card Data via Shopware REST API .....</b>	<b>23</b>
<b>8</b>	<b>Set-up a cron job to activate the timed operations .....</b>	<b>24</b>
<b>9</b>	<b>Testing .....</b>	<b>25</b>
9.1	Test Data .....	25
<b>10</b>	<b>Errors and their Solutions .....</b>	<b>29</b>
10.1	The Referrer URL appears in my Analytics Tool .....	29
<b>11</b>	<b>Compatibility with Third-Party Plugins .....</b>	<b>30</b>
11.1	Birthday and gender in Shopware .....	30
<b>12</b>	<b>Error Logging .....</b>	<b>31</b>
12.1	Log Levels .....	31
12.2	Log Location .....	31
<b>13</b>	<b>Advanced Information .....</b>	<b>32</b>
13.1	Transaction Object .....	32

# 1 Introduction

This manual leads you through the installation, configuration and usage of the payment module for Shopware and DIMOCO Payment Services (Dimoco).

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

- Your Dimoco API Username and Password
- Access to your merchant account of Dimoco - On this platform you can download the access information.
- Shopware payment module from [sellxed.com/shop](https://sellxed.com/shop).
- Access to your server.

In case you don't yet have a contract with Dimoco, we please ask you to contact their support. In case you do not have a contract with Dimoco yet, we kindly ask you to contact Dimoco Sales ([payment-merchants@dimoco.eu](mailto:payment-merchants@dimoco.eu)).

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

## 1.1 Process of the Installation

In this document you will find all important information for the installation of the module. It is important that you strictly follow the checklist. Only by doing so, can a secure usage in correspondence with all security regulations be guaranteed.

1. Configuration of the [basic settings](#) of the payment module
2. Configuration of the [payment methods](#)
3. Carrying out of a test purchase by means of the attached [test data](#) at the end of this document
4. Get in touch with your Technical Account Manager at Dimoco. This person will help you set up your account correctly for the successful handling of transactions.
5. If your test was successful, do not forget to switch the operating mode to live mode. Once you have switched to live, contact your Technical Account Manager to ensure that the live transactions can now be processed.

## 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 Dimoco 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 of the Payment Module and the Payment Methods

In order to configure and set up the [main module](#) in Shopware please double check that you received the following information:

- API Username (provided by Dimoco on setup email)
- API Password (provided by Dimoco on setup email)
- API Key (From your Connector's Settings)
- Shared Secret (From your Connector's Settings)

All the required information is provided directly from Dimoco or can be seen within the tab 'Connectors' at your Dimoco merchant account.

The screenshot shows the Dimoco backend dashboard. On the left is a green sidebar with navigation options: Gateway, Dashboard, Statistics, Connectors, and Transactions. The main content area is titled 'Dashboard' and features a 'Last Transactions' section with a 'Show all Transaction' button. Below this is a table with the following data:

Date	TransactionId	Type	Amount	Status	Connector	
2018-01-17 12:43	14500039	debit	65.00 EUR	error	testshop Creditcard with PCI	Details
2018-01-17 10:36	14500038	debit	80.00 EUR	success	testshop Creditcard with PCI	Details
2018-01-17 10:33	14500037	debit	65.00 EUR	success	testshop Creditcard with PCI	Details
2018-01-16 15:44	14500036	debit	80.00 EUR	error	testshop Creditcard with PCI	Details
2018-01-16 07:58	14500035	debit	280.00 EUR	success	testshop Creditcard with PCI	Details
2018-01-15 15:38	14500034	debit	65.00 EUR	success	testshop Creditcard with PCI	Details
2018-01-15 13:20	14500033	debit	65.00 EUR	success	testshop Creditcard with PCI	Details
2018-01-15 13:03	14500032	debit	65.00 EUR	success	testshop Creditcard with PCI	Details

Figure 2.1: Overview of the backend (My Payments) of Dimoco.

Enter the required credentials directly into the [main module](#) to allow payments over Dimoco. If you want to have additional information regarding how to use the backend of Dimoco we ask you kindly to directly call the customer support of Dimoco.

## 2.1 Live and Test Credentials

Please note that the configuration is different for live mode than for test mode as described in the following sections.

### 2.1.1 Live Credentials

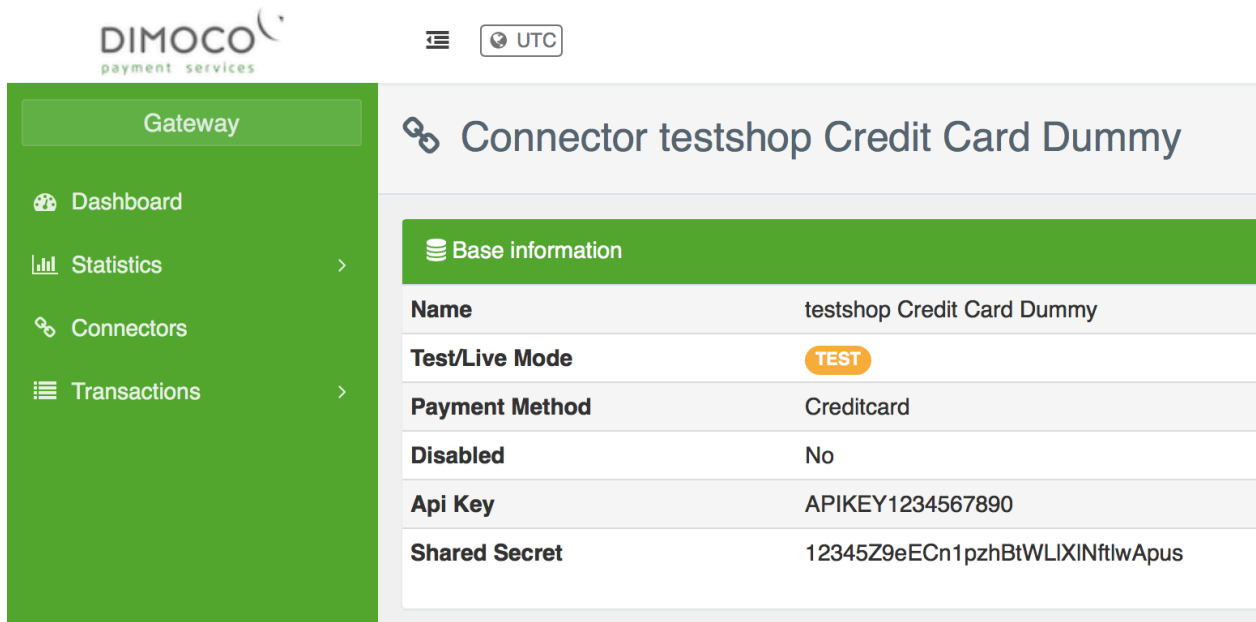
Enter the API Username and the API Password into the main configuration of the plugin. In your Dimoco account under "Connectors" you will find a connector for every payment method which has been set up by Dimoco.

Inside these connectors you will find the parameters API Key and Shared Secret. For each connector enter the API Key and Shared Secret into the configuration of the corresponding [payment method configuration](#) in your shops backend.

### 2.1.2 Test Credentials

A general test connector can be set for all payment methods. Specifically for credit cards and SEPA, it is possible to set your own test connector. If this is not set, the general test connector will be used. These connectors are used to test different payment flows.





**Figure 2.1:** The access credentials within the connector at Dimoco.

The connectors for live and test mode will be set up by Dimoco. If you wish to have additional connectors setup please contact Dimoco directly.

## 2.2 Payment Method Specific Configurations

### 2.2.1 Credit Card 3D Secure

You have the option to activate or deactivate 3D Secure authorisation within the payment method Credit Card. You can change the 3D Secure behaviour by navigating to the payment method settings and changing your 3D Secure settings to Off, Optional or Mandatory.

### 2.2.2 SEPA Direct Debits Mandates

If you have activated the payment method SEPA Direct Debits, you will need to make sure to configure the 'Mandate Creditor ID' and the 'Mandate Merchant Name'.

## 2.3 Integrated Payment Form (Ajax Authorization)

If you wish to use the embedded form (Ajax) you may activate the payment method called Credit Card (Ajax) instead. You will have to enter a 'Public Integration Key' which will be provided to you by Dimoco. You can also change the layout of the CVV element using CSS.

## 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 DimocoCw to the CwbDimocoCw 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/DimocoCw
  - 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 Dimoco 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 Dimoco. 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 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

<b>Autorisierungsmethode:</b>	Hidden Authorization (Authorization Interface) <span style="float: right;">▼ ?</span>
<b>Authorized Status:</b>	Komplett bezahlt <span style="float: right;">▼ ?</span>
<b>Uncertain Status:</b>	Der Kredit wurde vorlaeufig akzeptiert. <span style="float: right;">▼ ?</span>
<b>Cancelled Status:</b>	Vorgang wurde abgebrochen. <span style="float: right;">▼ ?</span>
<b>Captured Status:</b>	Don't change order status <span style="float: right;">▼ ?</span>
<b>Kundenadresse:</b>	Keine Adresse senden <span style="float: right;">▼ ?</span>
<b>Capturing:</b>	Directly after order <span style="float: right;">▼ ?</span>
<b>Minimaler Gesamtbetrag:</b>	<input type="text"/> <span style="float: right;">?</span>
<b>Maximaler Gesamtbetrag:</b>	<input type="text"/> <span style="float: right;">?</span>
<b>New Order Status:</b>	Offen <span style="float: right;">▼ ?</span>
<b>Validation:</b>	Nach dem Wählen der Zahlungsart <span style="float: right;">▼ ?</span>

Figure 4.1: Configuration of the Payment Method

### 4.3 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

---

Versandkosten | **Zahlart Auswahl** | Länder Auswahl | Kategorien sperren | Erweiterte Einstellungen

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.

## 4.4 Direct Capturing of Transactions

The option "Capture" allows you to specify if you wish to debit payments directly or if you first

wish to authorise them and then debit the payment at a later point.

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

## Different settings between Dimoco and the module

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

## 4.5 Uncertain Status

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

### 4.5.1 Setting the order state

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

## 4.6 Optional: Validation

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

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

You have the choice between these options:

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



#### **4.6.1 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.

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

#### 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 Dimoco module.

### 6.1 Capturing / Canceling of Orders

#### 6.1.1 Capturing Orders

##### Please note

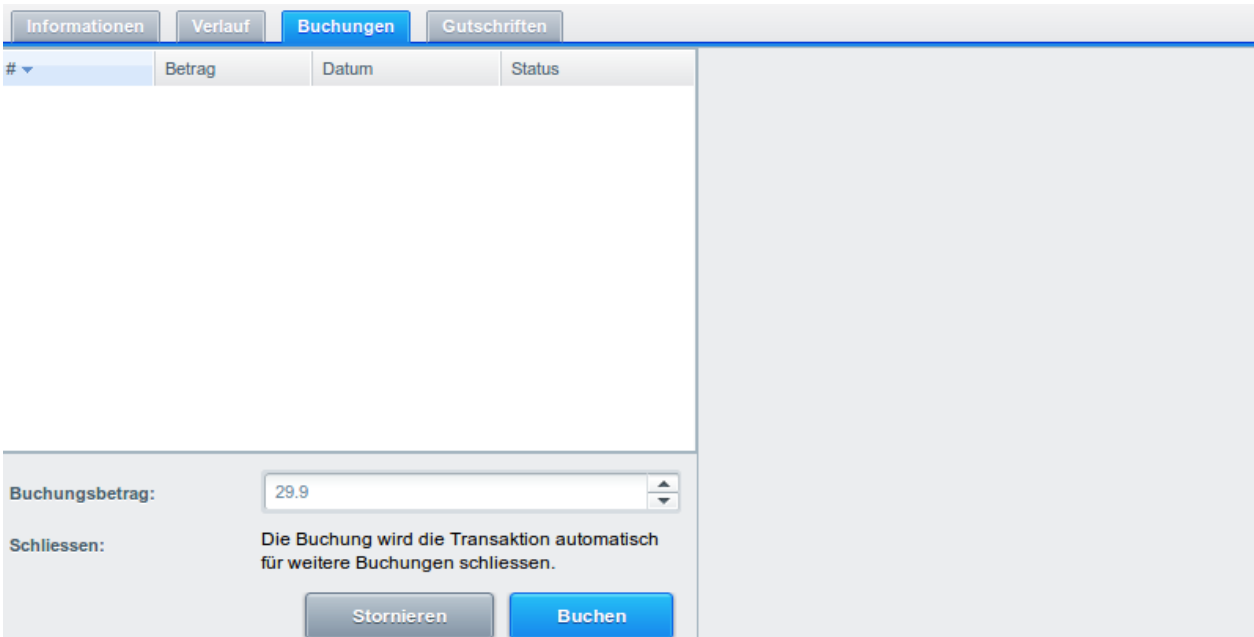
The transaction management between your shop and Dimoco is not synchronized. If you capture payments with Dimoco, the status in the shop will not be updated and a second capturing in the shop is not possible.

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 Dimoco.

#### 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.



#	Betrag	Datum	Status

**Buchungsbetrag:**

**Schliessen:** Die Buchung wird die Transaktion automatisch für weitere Buchungen schliessen.

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.

Bestellungs-Details: 20006
x

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 Dimoco. 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 Dimoco. 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 Dimoco 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

<b>Saved cards:</b>	<div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;">xxxx xxxx xxxx 0111 <span style="float: right;">▼</span></div> <p>You may choose one of the cards you paid before on this site.</p>
<b>Karteninhaber*</b>	<div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;">Test Buyer </div> <p>Bitte geben Sie den Karteninhaber auf Ihrer Kreditkarte ein.</p>
<b>Kartennummer*</b>	<p>xxxx xxxx xxxx 0111</p> <p>Bitte geben Sie hier die Kartennummer Ihrer Kreditkarte ein.</p>
<b>Karten Ablaufdatum*</b>	<div style="display: flex; gap: 10px;"> <div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;">03 <span style="float: right;">▼</span></div> <div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;">2015 <span style="float: right;">▼</span></div> </div> <p>Wählen Sie das Ablaufdatum Ihrer Karte aus.</p>
<b>CVC Code*</b>	<div style="border: 1px solid #ccc; height: 20px; margin-bottom: 5px;"></div> <p>Bitte geben Sie hier den CVC Ihrer Karte ein. Sie finden diesen auf der Rückseite Ihrer Kreditkarte.</p>

**Figure 6.1:** Alias Manager Usage in Shopware

#### Please note!

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

## 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/CwbdimococwTransactions>. 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 > Dimoco )

Additionally, you can also access the transaction via the Payment ID (transmitted through Dimoco). 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 Dimoco. 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 Dimoco 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 Card

Card number	4111 1111 1111 1111	Visa Success
Expiry Date	12/2020	
CVC	123	
Card number	4242 4242 4242 4242	Visa Failure
Expiry Date	12/2020	
CVC	123	
Card number	5555 5555 5555 4444	MasterCard Success
Expiry Date	12/2020	
CVC	123	
MasterCard (Failure)	5105 1051 0510 5100	MasterCard Failure
Expiry Date	12/2020	
CVC	123	

#### SOFORT

Instructions	Set the last name of the customer to the given value to simulate an outcome.	Simulation control
ignore	Do not simulate any behaviour	
success	The transaction succeeds immediately	
failed	The transaction fails immediately	
redirectSuccess	A redirection occurs, and then the transaction succeeds	
redirectError	A redirection occurs, and then the transaction fails	
pending	The transaction stays pending	
html	Submit a given form, then the transaction succeeds	
htmlError	Submit a given form, then the transaction fails	

### giropay

	Set the last name of the customer to the given value to simulate an outcome.	Simulation control
Instructions		
ignore	Do not simulate any behaviour	
success	The transaction succeeds immediately	
failed	The transaction fails immediately	
redirectSuccess	A redirection occurs, and then the transaction succeeds	
redirectError	A redirection occurs, and then the transaction fails	
pending	The transaction stays pending	
html	Submit a given form, then the transaction succeeds	
htmlError	Submit a given form, then the transaction fails	

### EPS

	Set the last name of the customer to the given value to simulate an outcome.	Simulation control
Instructions		
ignore	Do not simulate any behaviour	
success	The transaction succeeds immediately	
failed	The transaction fails immediately	
redirectSuccess	A redirection occurs, and then the transaction succeeds	
redirectError	A redirection occurs, and then the transaction fails	
pending	The transaction stays pending	
html	Submit a given form, then the transaction succeeds	
htmlError	Submit a given form, then the transaction fails	

### paysafecard

	Set the last name of the customer to the given value to simulate an outcome.	Simulation control
Instructions		
ignore	Do not simulate any behaviour	
success	The transaction succeeds immediately	
failed	The transaction fails immediately	
redirectSuccess	A redirection occurs, and then the transaction succeeds	

	A redirection occurs, and then the
redirectError	transaction fails
pending	The transaction stays pending
html	Submit a given form, then the transaction succeeds
htmlError	Submit a given form, then the transaction fails

**Przelewy24**

	Set the last name of the customer to the given value to simulate an outcome.	Simulation control
Instructions	Do not simulate any behaviour	
ignore	The transaction succeeds immediately	
success	The transaction fails immediately	
failed	A redirection occurs, and then the transaction succeeds	
redirectSuccess	A redirection occurs, and then the transaction fails	
redirectError	The transaction stays pending	
pending	Submit a given form, then the transaction succeeds	
html	Submit a given form, then the transaction fails	
htmlError		

**IDEAL**

	Set the last name of the customer to the given value to simulate an outcome.	Simulation control
Instructions	Do not simulate any behaviour	
ignore	The transaction succeeds immediately	
success	The transaction fails immediately	
failed	A redirection occurs, and then the transaction succeeds	
redirectSuccess	A redirection occurs, and then the transaction fails	
redirectError	The transaction stays pending	
pending	Submit a given form, then the transaction succeeds	
html	Submit a given form, then the transaction fails	
htmlError		

**Sepa Direct Debits**

Simulation control

	Set the last name of the customer to the given value to simulate an
Instructions	outcome.
ignore	Do not simulate any behaviour
success	The transaction succeeds
failed	immediately
redirectSuccess	The transaction fails immediately
redirectError	A redirection occurs, and then the transaction succeeds
pending	A redirection occurs, and then the transaction fails
html	The transaction stays pending
htmlError	Submit a given form, then the transaction succeeds
	Submit a given form, then the transaction fails

## 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 Dimoco 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 Dimoco 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/CwbDimocoCw';
Shopware()->Loader()->registerNamespace('Customweb', 'Customweb/');
Shopware()->Loader()->registerNamespace('CwbDimocoCw_Helpers',
$pluginPath . '/Helpers/');
Shopware()->Loader()->registerNamespace('CwbDimocoCw_Entities',
$pluginPath . '/Entities/');
Shopware()->Loader()->registerNamespace('CwbDimocoCw_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, 'DimocoCw') === 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:

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