

# Installation Manual for Innocard OpenCart

This manual describes the installation and usage of the Innocard extension for OpenCart.

**Release Date:** Wed, 12 Jun 2019 13:07:23 +0200  
**Version:** 4.0.80

customweb GmbH  
General-Guisan-Strasse 47  
CH-8400 Winterthur

E-Mail: [info@customweb.ch](mailto:info@customweb.ch)

Phone:

CH: +41 (0)44 533 15 70

DE: +49 (0)180 3 729 636

UK: +44 (0)2036 086 268

US: +1 (0)650 618 3944

© copyright by customweb GmbH Wed, 12 Jun 2019 13:07:23 +0200

# Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>5</b>
1.1	Installation Process .....	5
<b>2</b>	<b>Configuration .....</b>	<b>6</b>
2.1	Main Configuration .....	6
2.1.1	Settings > Shops > Your Shop (Innocard Backend) .....	6
2.1.2	Production Certificate .....	6
2.1.3	Payment Notifcation .....	6
2.2	Payment Method Configuration .....	7
2.3	Testing .....	7
<b>3</b>	<b>Module Installation and Update in the OpenCart Shop .....</b>	<b>8</b>
3.1	Installation .....	8
3.2	Updates and Upgrades .....	8
3.2.1	Update Checklist .....	8
3.2.2	Update Instructions .....	9
<b>4</b>	<b>Module Configuration in the OpenCart Shop .....</b>	<b>10</b>
<b>5</b>	<b>OpenCart 1.5.x / 2.0 / 3.0 Installation - Additions .....</b>	<b>11</b>
5.1	Configuration of the Main Module .....	11
5.2	Opencart 2.0 / 3.0 - Refresh Modifications .....	11
5.3	OpenCart 1.0 Specifications .....	11
5.3.1	Capture Delay .....	11
5.4	Defining the URL for the Transaction Feedback .....	12
5.5	Configuration of the Payment Module .....	12
5.6	Shop ID .....	12
5.7	Order Prefix .....	12
5.8	Selecting the Authorisation Method .....	13
5.9	Direct Capturing of Transactions .....	13
5.10	Uncertain Status Status .....	13
5.11	Optional: Validation .....	13
5.12	Multi-Shop Set-Up .....	14
<b>6</b>	<b>Settings / Configuration of Payment Methods .....</b>	<b>16</b>
6.1	General Information About the Payment Methods .....	16

6.2	Information on Payment Status .....	16
6.2.1	Order status "pending" / imminent payment (or similar) .....	16
6.2.2	Order status "cancelled" .....	16
<b>7</b>	<b>The Module in Action .....</b>	<b>17</b>
7.1	Useful Transaction Information on the Order .....	17
7.2	Usage of the Alias Managers / Token Solution .....	17
7.3	Capturing / Cancelling of Orders .....	18
7.3.1	Capturing Orders .....	18
7.3.2	Cancel Orders .....	19
7.4	Refunding Orders .....	19
7.5	Set-up a cron job to activate the timed operations .....	20
<b>8</b>	<b>Testing .....</b>	<b>21</b>
8.1	Test Data .....	21
<b>9</b>	<b>Errors and their Solutions .....</b>	<b>23</b>
9.1	The Referrer URL appears in my Analytics Tool .....	23
<b>10</b>	<b>Compatibility with Third-Party Plugins .....</b>	<b>24</b>
10.1	Birthday and gender in OpenCart .....	24
<b>11</b>	<b>Error Logging .....</b>	<b>25</b>
11.1	Log Levels .....	25
11.2	Log Location .....	25
<b>12</b>	<b>Advanced Information .....</b>	<b>26</b>
12.1	Transaction Object .....	26

# 1 Introduction

This manual describes how to install, configure and use the payment module for OpenCart and Innocard.

Before getting started with the Installation make sure to have all necessary data at hand:

- Innocard Shop ID
- Innocard Test and Live Certificate
- OpenCart Payment module from [sellxed.com/shop](https://sellxed.com/shop)
- Login data for your server and shop

## 1.1 Installation Process

This document contains all information necessary to install the module. When installing the module, strictly follow the instructions. This ensures the safe use of the module in compliance with all security regulations.

1. Entering the [Main Configuration](#) into the main module
2. Configuration of the [payment methods](#)
3. Conducting a test order with the [Test Data](#) provided at the end of this document
4. Creating the production certificate in the Innocard-backend.

### 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 informations 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; otherwise the payment feedback of Innocard might not get through to the shop.

## 2 Configuration

### 2.1 Main Configuration

For the module configuration, go to the setting page in the [Main Module](#) of your shop and enter the following data for the Live Tests in the corresponding fields.

#### 2.1.1 Settings > Shops > Your Shop (Innocard Backend)

All information necessary for the installation can be found in the Innocard Backend (<https://secure.innopay.ch/vads-merchant/>) under Settings > Shops. Here you can select the shop you want to install.

Transfer the data into the [Main Module](#):

- Shop ID
- Test certificate

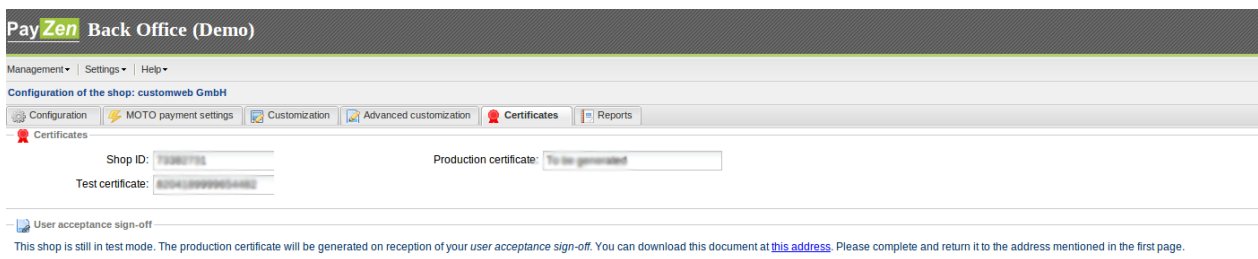


Figure 2.1: Payzen Settings

#### 2.1.2 Production Certificate

As soon as the test payments were successful, set the **Operation Mode** to Live and request the activation of your account with Innocard. After the activation you can enter the **Certification Key** under Settings > Shops > in the main module.

#### 2.1.3 Payment Notification

For the **Instant Payment Notification** please enter the URL that is generated in the main module in the backend of Innocard. More information can be found under the section [Transaction Feedback](#). Enter the notification URL in the backend of Innocard under Settings > Notification Rules > 'IPN bei Erstellung einer Rate' into both TEST and LIVE fields and check 'Automatic Retry in case of failure'.

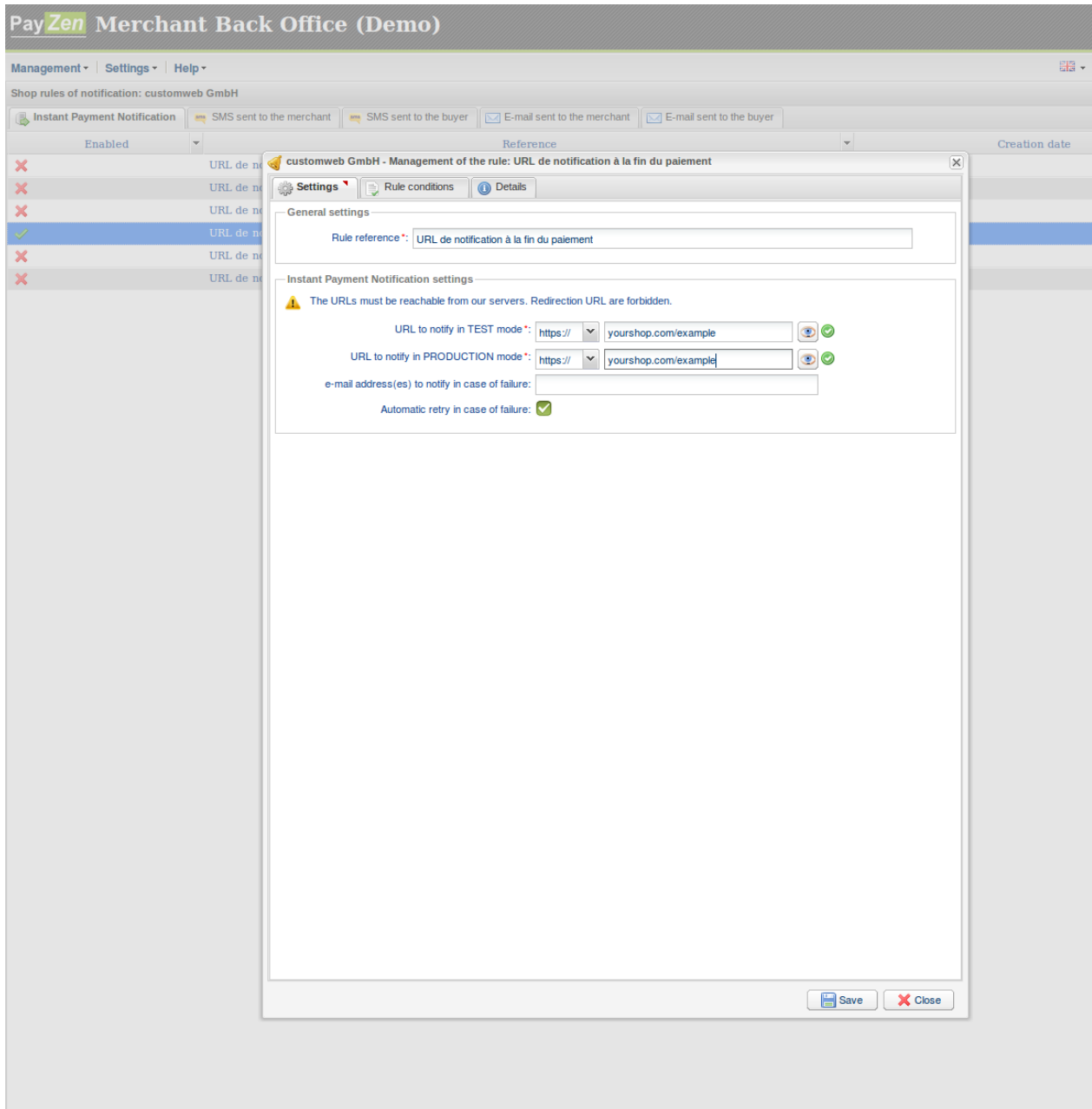


Figure 2.1: Entering the Notification URL

## 2.2 Payment Method Configuration

All other setting options in the main module are not relevant for the product Checkout Page and can be ignored. The next step is the [payment method](#) configuration.

## 2.3 Testing

The credit card data for the testing is shown on the Payment Page. By clicking on it the card number is transferred to the mask.

## 3 Module Installation and Update in the OpenCart 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 download 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 shops 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 thats 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 back up 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. They can be found on the plugin page in the shop under the section change log on the lower end of the page. If there are no special remarks, you can proceed by just overwriting the files in your system

## 4 Module Configuration in the OpenCart 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 back-ups!

Please create a back-up 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.

## 5 OpenCart 1.5.x / 2.0 / 3.0 Installation - Additions

All our modules are compatible with OpenCart 2.0. If you use OpenCart 2.0 upload all the files in the folder files\_2.x to the root folder of your shopping cart. In case you are using OpenCart 1.5.x as basis you should use the folder files\_1.x

In order to guarantee smooth operations and the usage of all features please make sure that you follow the instructions below.

### 5.1 Configuration of the Main Module

You will find the settings for the main module under "**Extension > Modules > Innocard Base Module**". Install the module by clicking **Install**.

By clicking **Edit** you can configure the main module. Enter all data in the corresponding fields. The required data has either already been saved in the back-end of Innocard has been sent to you by Innocard. Each option is, furthermore, explained in short info texts in the shop.

### 5.2 Opencart 2.0 / 3.0 - Refresh Modifications

In case you are using OpenCart 2.0 you will have to refresh the modifications. Go to **Module > Modifications**. and click on the refresh button in the top right corner. Make sure that the folder system > modification is writable.

### 5.3 OpenCart 1.0 Specifications

In case you are using OpenCart 1.0 you should make sure that you use VQMOD in order to be able to use the whole set of functionalities. More Information about the project and the installation can be found under: <https://github.com/vqmod/vqmod/wiki/Installing-vQmod-on-OpenCart>

#### 5.3.1 Capture Delay

You have the option to set a Capture Delay with the configuration. The Capture Delay defines, how many days after the authorization the transaction should be captured. We recommend to leave the number at 0 (standard setting).

In case you want to 0 we strongly recommend setting up a Cron Job so you can stay informed about the transaction status at all times.

In any case, it is helpful to set up the Cron Job to be informed about the Capture Status of the transaction. Instructions on how to set up a Cron Job can be found in this manual under [Cron Job](#).

## 5.4 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 Innocard configuration. The URL to be entered can be found in your OpenCart Shop under: `___setupForm___` > **Setup**

## 5.5 Configuration of the Payment Module

After having successfully configured the main module, you can find the settings for the individual payment methods in your shop under **Extensions > Payments** . Each payment method is listed individually. Install the payment methods you wish to offer to your customers. You can carry out individual settings for each payment method and thereby optimally adapt the payment methods to your existing processes. The most central options are described in more detail further below.

By clicking on **Install** the payment method is activated in your shop. Click **Edit** in order to modify the configuration of the payment method.

## 5.6 Shop ID

In the case of a multi-store set up (cf. [chapter: Multi-Store](#)), the parameter Shop ID identifies the shop's request and, therefore, redirects the feedback of Innocard based on the shop ID.

## 5.7 Order Prefix

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

## 5.8 Selecting the Authorisation Method

You can choose between two options of authorizing credit card payments. By selecting one of the two under the option 'Authorisation Methods', you define how you want to process the individual payment methods.

Further information on the various authorisation methods can be found on our [sellxed website](#).

### Payment Processing

Please pay attention to the fact that you require the activation of further options with Innocard in order to use some of the authorisation methods.

It may be that certain authorization methods are not available for every payment method.

Alongside the introduction of PCI 3 there are additional certification requirements in case you are using the authorization method Hidden or AJAX. Please contact Innocard for further information or switch to another authorization method.

## 5.9 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 the debit the payment at a later point.

Depending on your acquiring contract, a reservation is only guaranteed for a specific amount 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 Innocard and the module

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

## 5.10 Uncertain Status Status

You can specifically label orders for which for example no 3D-Secure verification nor any other security mechanism was carried out. This allows you to manually control the order before shipment.

## 5.11 Optional: Validation

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

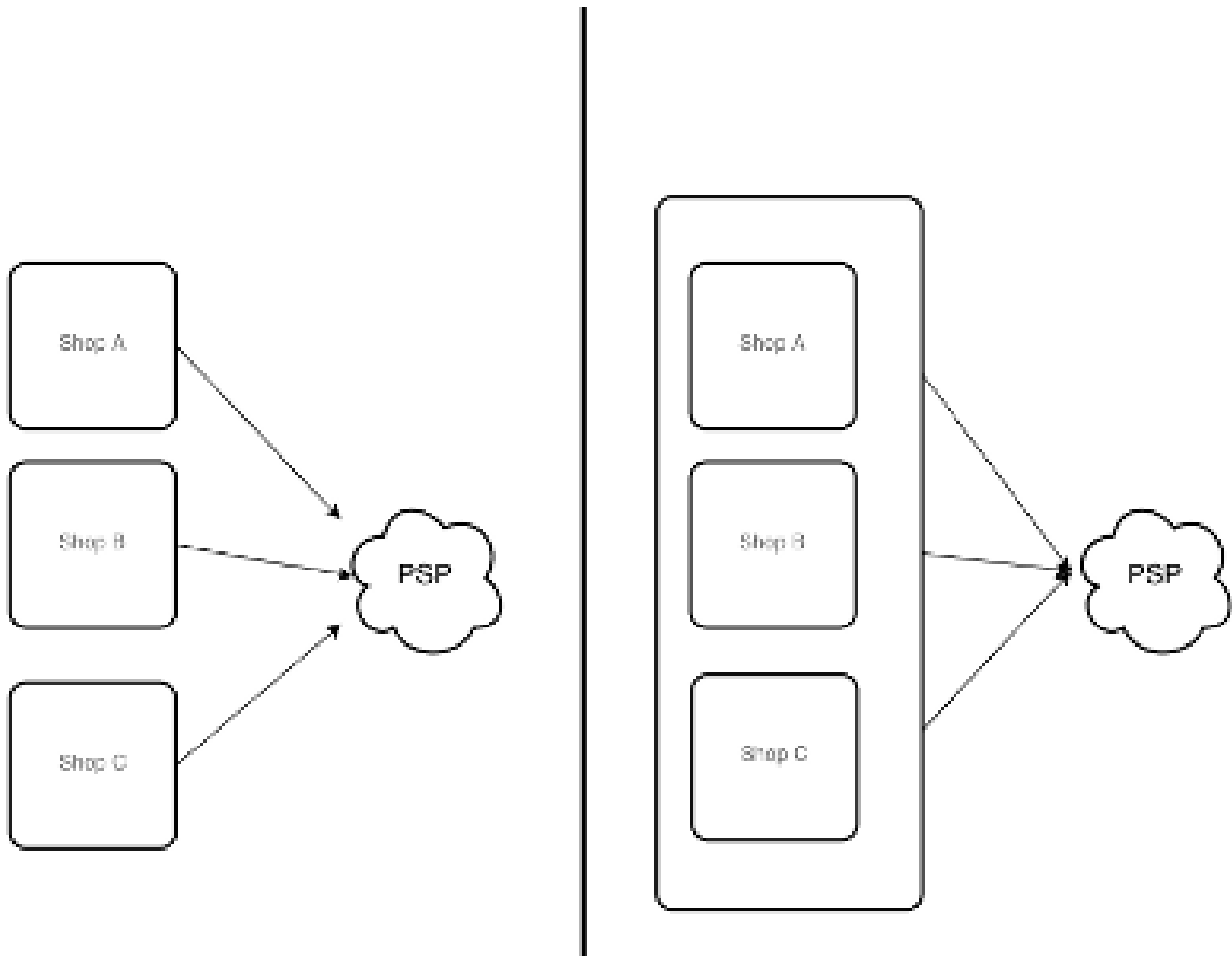
With the option 'Validation' you can define the moment when the payment method should be made visible to the customer during the checkout process. This setting is relevant for modules

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. 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 he or she 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 Innocard during the authorisation process. The payment method is displayed in any case

## 5.12 Multi-Shop Set-Up

The payment module is designed for the usage of one Innocard contract for several shops. It must be differentiated, however, between the case of an integrated multi-shop functionality and independent shops:



**Figure 5.1:** Multi-shop set-up

## Licensing

In the case of independent shops, you require a payment module license for each shop. For scale prices, please contact us directly.

## 6 Settings / Configuration of Payment Methods

### 6.1 General Information About the Payment Methods

All sellxed payment modules are delivered with 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 Innocard 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.

### 6.2 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 Innocard** or any similar pending status which is implemented by the module.

#### 6.2.1 Order status "pending" / imminent payment (or similar)

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

#### 6.2.2 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.



## 7 The Module in Action

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

### 7.1 Useful Transaction Information on the Order

You can find an overview over all the most important information of a transaction in all orders that have been processed with the sellxed module. Among others, this information allows for the definite attribution of the orders to their corresponding transaction, seen in the back-end of Innocard.

Orders	
<b>Order Details</b>	Order ID: #2
<b>Payment Details</b>	Invoice No.: <a href="#">[ Generate ]</a>
<b>Shipping Details</b>	Store Name: Your Store
<b>Products</b>	Store Url: <a href="http://demo.sellxed.com/">http://demo.sellxed.com/</a>
<b>History</b>	Customer: <a href="#">Sascha krüsi</a>
	Customer Group: Default
	E-Mail: info@customweb.ch
	Telephone: 33123456798
	Total: 78.90€
	Order Status: Pending
	IP Address: 160.85.155.47
	User Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Ubuntu Chromium/30.0.1599.114 Chrome/30.0.1599.114 Safari/537.36
	Accept Language: de,en-US;q=0.8,en;q=0.6
	Date Added: 13/11/2013
	Date Modified: 13/11/2013

Figure 7.1: Transaction Information in OpenCart.

### 7.2 Usage of the Alias Managers / Token Solution

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

Use Stored Card

Card Holder Name\*   
Please enter here the card holder name on the card.

Card Number\*   
Please enter here the number on your card.

Card Expiration\*    
Select the date on which your card expires.

CVC Code\*   
Please enter here the CVC code from your card. You find the code on the back of the card.

**Figure 7.1:** Alias Manager

### Alias Manager Options

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

## 7.3 Capturing / Cancelling of Orders

### 7.3.1 Capturing Orders

In order to capture orders, open the transaction manager under Sales > Innocard Transactions. Here you will find an overview over all transactions. Search for the order in the field with the order numbers. By clicking "view" you will open the transaction of the order.

#### 7.3.1.1 Capturing Complete Orders or Partial Capturing

By clicking the button "Capture Transaction" a new window opens up. You can now capture every item individually in case you do not wish to send all items at once. The amount of possible partial captures depends on your contract. Please contact Innocard directly in order to clarify questions.

### Capturing of Orders in the Back-End of Innocard

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

In case you do not wish to capture all items of an order, you can close the transaction by clicking the box.

**PARTIAL CAPTURE**

With the following form you can perform a partial capture.

Name	SKU	Type	Tax Rate	Quantity	Total Amount (excl. Tax)	Total Amount (incl. Tax)
iPhone	product 11	product	0 %	<input type="text" value="1"/>	<input type="text" value="74.24"/>	<input type="text" value="74.24"/>
Flat Shipping Rate	shipping	shipping	0 %	<input type="text" value="1"/>	<input type="text" value="3.68"/>	<input type="text" value="3.68"/>
Total Capture Amount:						77.91 EUR

Close transaction for further captures

[Capture](#)

Figure 7.1: Capturing of Orders

**Partial Capturing**

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

### 7.3.2 Cancel Orders

By clicking "Cancel Transaction", the transaction is **cancelled** and the reserved amount on your customer's card is released automatically.

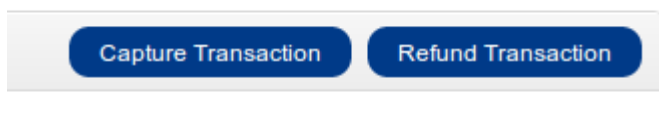


Figure 7.1: Capture or Cancel in OpenCart.

## 7.4 Refunding Orders

In order to refund orders, open the transaction information (cf. above).

You can refund individual items or any amount of your choice by modifying the total amount or the item quantity.

## PARTIAL REFUND

With the following form you can perform a partial refund.

Name	SKU	Type	Tax Rate	Quantity	Total Amount (excl. Tax)	Total Amount (incl. Tax)
iPhone	product 11	product	0 %	<input type="text" value="1"/>	<input type="text" value="74.24"/>	<input type="text" value="74.24"/>
Flat Shipping Rate	shipping	shipping	0 %	<input type="text" value="1"/>	<input type="text" value="3.68"/>	<input type="text" value="3.68"/>
Total Refund Amount:						77.91 EUR

Close transaction for further refunds

Refund

**Figure 7.1:** Refunds in OpenCart for Innocard.

### Maximal Refund

With our sellxed modules it is not possible to refund more than 100% of the originally authorised amount.

## 7.5 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 OpenCart Cron engine. Especially the update function allows you to automatically retrieve additional information or changes of your order directly via the API of Innocard. Please note it could be that in order to use the update feature it may be necessary that Innocard activates additional options in your account.

In order to use the timed operations, please schedule a cron job in your server to the following controller:

<https://www.your-shop.com/index.php?route=innocardcw/cron/cron>

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

## 8 Testing

Before switching from test to live mode it is important that you test the module extensively. You can find all relevant test credit card data under: <http://www.sellxed.com/de/testen>.

### Testing

Do not forget to switch the operating mode from test to live after having successfully tested the module.

### 8.1 Test Data

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

#### Credit Card

Card number	4917610000000000
Expiry Date	12/2021
CVC	123

#### Visa

Card number	4970100000000000
Expiry Date	12/2021
CVC	111

Card number	4970100000000001
Expiry Date	12/2021
CVC	111

Card number	4970100000000009
Expiry Date	12/2021
CVC	111

#### MasterCard

Card number	5970100300000000
Expiry Date	12/2021
CVC	111

Card number	5970100300000001
Expiry Date	12/2021
CVC	111

Card number	5970100300000009
Expiry Date	12/2021
CVC	111

#### American Express

Card number	378282246310005
Expiry Date	12/2020

CVC

123

## 9 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>

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

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

## 10 Compatibility with Third-Party Plugins

The plugins listed below are compatible with our payment modules and allow you to handle certain tasks in a easier way.

### 10.1 Birthday and gender in OpenCart

For certain payment service providers it is necessary to check the birthday an the gender of a customer. OpenCart does not check this by default.

#### How to enable gender and birthday checks in your shops checkout

1. Add two new custom fields to your checkout via your shops backend under "Customers > Custom Fields"
2. Modify the order context getters to return the value of your custom checkout field from the order / session (or wherever the previous step saves the data).

##### Order Context Getters

- AbstractOrderContext
- getBillingDateOfBirth()
- getBillingGender()

These functions can be found in "system/library/cw/Innocard /AbstractOrderContext.php".



## 11 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.

### 11.1 Log Levels

You can configure the log level in the Innocard settings.

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

### 11.2 Log Location

The log file is stored in the default log folder of OpenCart. The path is configured in the config.php of your shop system. (Default Path: {shopRootDirectory}/system/logs or {shopRootDirectory}/system/storage/logs)

## 12 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.

### 12.1 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.

In your script initialize the base of OpenCart.

#### Opencart 1.x

```
require_once('config.php');

require_once(DIR_SYSTEM . 'startup.php');

require_once(DIR_SYSTEM . 'library/customer.php');
require_once(DIR_SYSTEM . 'library/affiliate.php');
require_once(DIR_SYSTEM . 'library/currency.php');
require_once(DIR_SYSTEM . 'library/tax.php');
require_once(DIR_SYSTEM . 'library/weight.php');
require_once(DIR_SYSTEM . 'library/length.php');
require_once(DIR_SYSTEM . 'library/cart.php');

$registry = new Registry();
$loader = new Loader($registry);
$registry->set('load', $loader);
$config = new Config();
$registry->set('config', $config);
$db = new DB(DB_DRIVER, DB_HOSTNAME, DB_USERNAME, DB_PASSWORD,
DB_DATABASE);
$registry->set('db', $db);
```

#### Opencart 2.x

```
require_once('config.php');
require_once(DIR_SYSTEM . 'startup.php');
// Registry
$registry = new Registry();

// Config
```

```
$config = new Config();
$config->load('default');
$config->load('catalog');
$registry->set('config', $config);
$loader = new Loader($registry);
$registry->set('load', $loader);
$registry->set('db', new DB($config->get('db_engine'), $config->get(
('db_hostname'), $config->get('db_username'), $config->get(
('db_password'), $config->get('db_database'), $config->get('db_port'))));
```

### Opencart 3.x

```
require_once('config.php');
require_once(DIR_SYSTEM . 'startup.php');
// Registry
$registry = new Registry();

// Config
$config = new Config();
$config->load('default');
$config->load('catalog');
$registry->set('config', $config);
$loader = new Loader($registry);
$registry->set('load', $loader);
$registry->set('db', new DB($config->get('db_type'), $config->get(
('db_hostname'), $config->get('db_username'), $config->get(
('db_password'), $config->get('db_database'), $config->get('db_port'))));
```

Include the module specific files and set registry.

```
require_once DIR_SYSTEM.'library/cw/init.php';
require_once DIR_SYSTEM.'library/cw/InnocardCw/Util.php';
require_once DIR_SYSTEM.'library/cw/InnocardCw/Entity/Transaction.php';
InnocardCw_Util::setRegistry($registry);
```

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

Load the transaction by Id:

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

Load transactions by Order ID:

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

