

Installation Manual for Innocard Drupal

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

Release Date: Wed, 12 Jun 2019 13:07:23 +0200
Version: 3.0.144

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

E-Mail: 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

| | | |
|----------|--|-----------|
| 1 | Introduction | 5 |
| 1.1 | Installation Process | 5 |
| 2 | Configuration | 6 |
| 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 |
| 3 | Module Installation and Update in the Drupal Shop | 8 |
| 3.1 | Installation | 8 |
| 3.2 | Updates and Upgrades | 8 |
| 3.2.1 | Update Checklist | 8 |
| 3.2.2 | Update Instructions | 9 |
| 4 | Module Configuration in the Drupal Shop | 10 |
| 4.1 | Basic Module Configuration | 10 |
| 4.1.1 | Capture Delay | 10 |
| 4.2 | Transaction Feedback URL Settings | 10 |
| 4.3 | Payment Method Configuration | 10 |
| 4.4 | Shop ID | 11 |
| 4.5 | Order Prefix | 11 |
| 4.6 | Selecting the Authorisation Method | 12 |
| 4.7 | Direct Capturing of Transactions | 12 |
| 4.8 | Uncertain Status Status | 12 |
| 4.9 | Optional: Validation | 12 |
| 4.10 | Multi-Shop Set-Up | 13 |
| 5 | Settings / Configuration of Payment Methods | 15 |
| 5.1 | General Information About the Payment Methods | 15 |
| 5.2 | Information on Payment Status | 15 |
| 5.2.1 | Order status "pending" / imminent payment (or similar) | 15 |
| 5.2.2 | Order status "cancelled" | 15 |
| 6 | The Module in Action | 16 |

| | | |
|-----------|---|-----------|
| 6.1 | Capturing / Cancelling Orders | 16 |
| 6.1.1 | Capturing Orders | 16 |
| 6.2 | Useful Transaction Information | 17 |
| 6.3 | Refunds | 17 |
| 6.4 | Alias Manager / Token Solution Usage | 18 |
| 6.5 | Cron Job Setup | 19 |
| 7 | Testing | 20 |
| 7.1 | Test Data | 20 |
| 8 | Errors and their Solutions | 22 |
| 8.1 | The Referrer URL appears in my Analytics Tool | 22 |
| 9 | Error Logging | 23 |
| 9.1 | Log Levels | 23 |
| 9.2 | Log Location | 23 |
| 10 | Advanced Information | 24 |
| 10.1 | Transaction Object | 24 |

1 Introduction

This manual describes how to install, configure and use the payment module for Drupal 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
- Drupal Payment module from 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

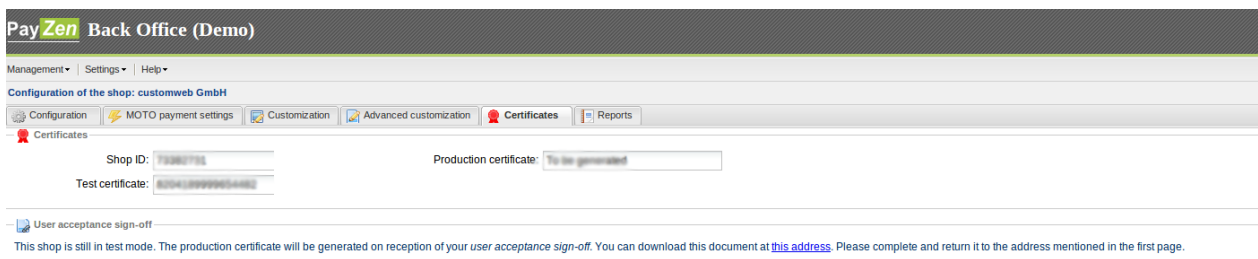


Figure 2.1: Payzen Settings

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'.

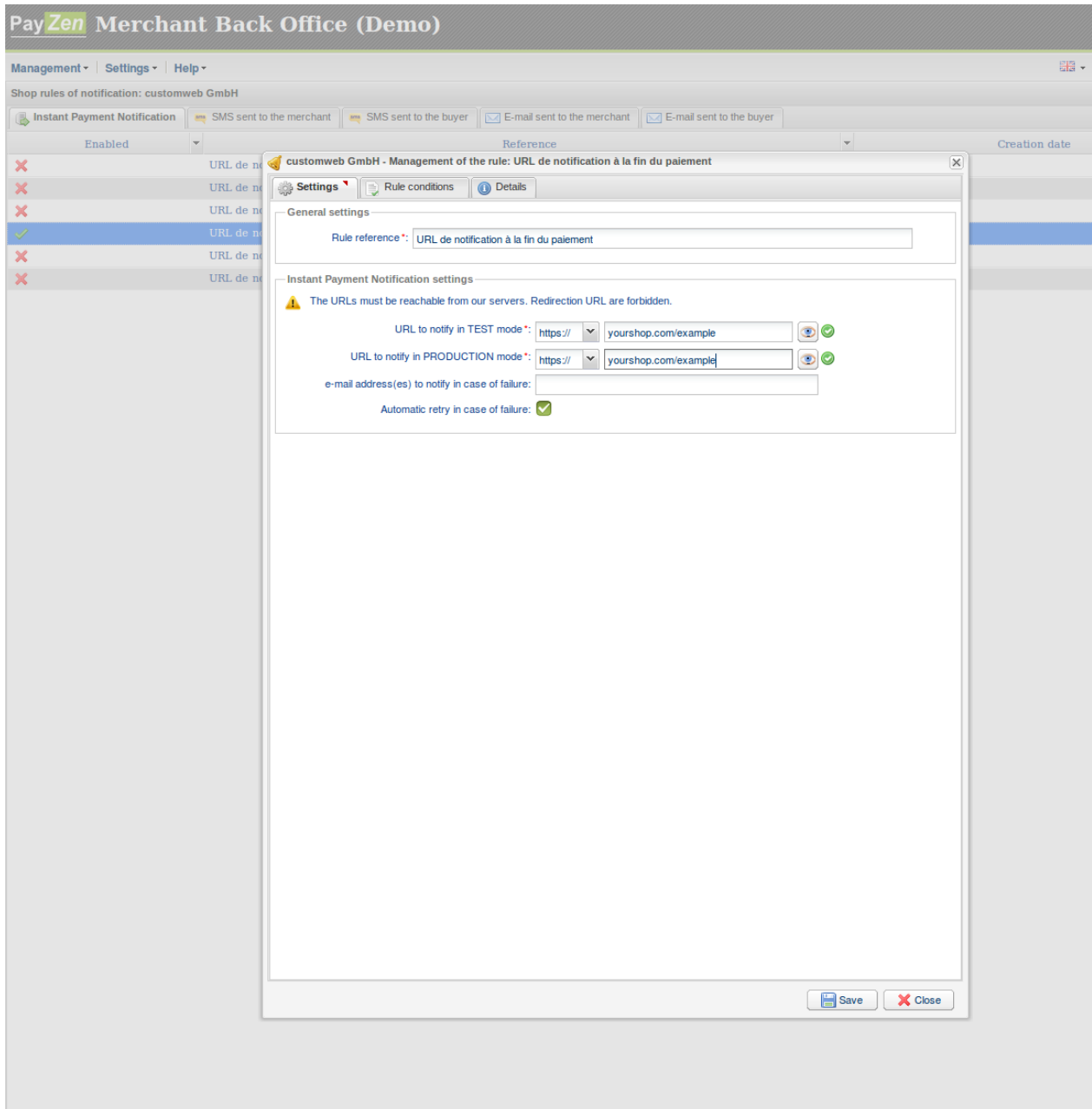


Figure 2.1: Entering the Notification URL

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 Drupal 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

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 Drupal 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.

4.1 Basic Module Configuration

In order to install the basic module, search for Innocard under **Site Settings > Modules**. Activate the module by checking the box and set click on **save configuration**.

Once the installation has been completed, go to **Store Settings > Innocard** to configure the basic settings. Enter the necessary parameters that are explained in the help texts.

4.1.1 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](#).

4.2 Transaction Feedback URL Settings

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 Drupal Shop under: **Store Settings > Innocard > Setup** (--> Tab top right)

4.3 Payment Method Configuration

Once the basic module configuration is completed, you can view the included payment methods under **Store Settings > Payment Methods**. Activate the payment methods that you want to

support in your shop. Individual adjustments can be made for each method to ensure maximum compatibility to your processes. Open the configuration by clicking **edit**. Then, click on the link shown under the menu item **Action** and the payment method configuration will open.

Actions [Show row weights](#)

| Elements | Operations |
|---|---|
| + Enable payment method: Credit Card Parameter: <i>Order</i> : [commerce-order] | <input type="button" value="edit"/> <input type="button" value="delete"/> |
| <input type="button" value="+ Add action"/> <input type="button" value="+ Add loop"/> | |

Figure 4.1: Payment Method Configuration.

4.4 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.

4.5 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.

4.6 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.

4.7 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.

4.8 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.

4.9 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

4.10 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:

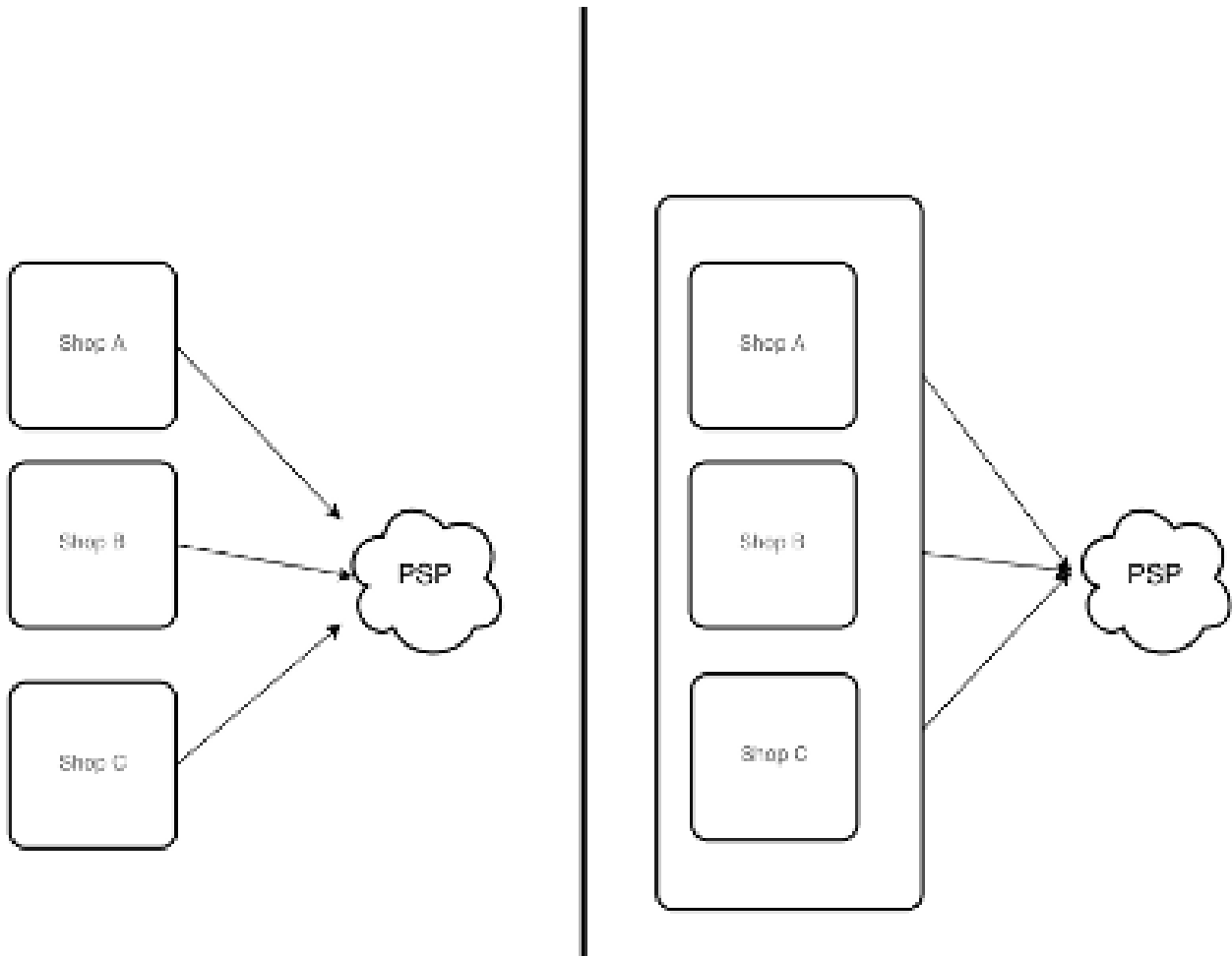


Figure 4.1: Multi-shop set-up

Licensing

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

5 Settings / Configuration of Payment Methods

5.1 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.

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 Innocard** 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 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.

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 over the most important features in the daily usage of the Innocard module.

#####conditional(isFeatureCapturingActive or isFeatureCancellationActive)#####

6.1 Capturing / Cancelling Orders

6.1.1 Capturing Orders

Please note:

The transaction management between Innocard and Drupal is not synchronized. If you capture payments through Innocard the order status in the shop is not adjusted and no second capture in the shop is possible.

In order to capture orders, make sure to set the Capture to **deferred** in the [Payment Method](#) configuration.

Open the order you want to capture and switch to the tab **Payments**. By clicking 'view' additional order information is displayed. To capture the order, click on the button **Capture** . Enter the amount you want to charge the customer. When you click **Capture**, the order is directly captured by Innocard.

| Captures | |
|----------------|-------------------------|
| Transaction ID | 12 |
| Order | 14 |
| Payment method | IDEAL |
| Remote ID | 36741662 |
| Message | |
| Amount | 34,20 € |
| Status | Success |
| Remote status | |
| Created | Mon, 11/10/2014 - 11:19 |

Figure 6.1: Capturing / Cancelling orders with Innocard

#####conditional#####

6.2 Useful Transaction Information

When opening any order processed by a sellxed module, an overview of important information can be accessed, as well as a transaction history which shows all process steps of the order.

To view the transaction information, open the order and change to the tab **Payments** where you will find all information on the payment.

Captures Refunds

| | |
|----------------|-------------------------|
| Transaction ID | 12 |
| Order | 14 |
| Payment method | iDEAL |
| Remote ID | 36741662 |
| Message | |
| Amount | 34,20 € |
| Status | Success |
| Remote status | |
| Created | Mon, 11/10/2014 - 11:19 |

Figure 6.1: Transaction Information in Drupal.

6.3 Refunds

Refunds for already charged transactions can be created directly in the shop and are transmitted to Innocard automatically. To create a refund, open the order, switch to the tab **Payment** and click on 'view'. A new window will open. Click on **Refund** to refund an order.

By means of the input field, you can define the amount you want to refund. By clicking **Refund**, the request is transmitted to Innocard the amount is added to the customer's credit balance.

Back

| Name | SKU | Type | Tax Rate | Quantity | Total Amount (excl. Tax) | Total Amount (incl. Tax) |
|----------------------------------|-------------------------------------|----------|----------|----------|--------------------------|--------------------------|
| Water Bottle 1 | WTR-BLU-OS_31 | product | 20% | 1 | 16.00 | 19.20 |
| Express shipping: 1 business day | Express shipping: 1 business day_33 | shipping | 0% | 1 | 15.00 | 15.00 |
| | | | | | Total Refund Amount | 34.20 EUR |

Close transaction for futher refunds

Refund

Figure 6.1: Refunds with Innocard.

Please keep in mind that the maximum for the refund is the originally charged amount.



6.4 Alias Manager / Token Solution Usage

The Alias Manager allows you to store credit card data securely with Innocard. To use this feature, activate the option "Alias Manager" in the [payment method setting](#). The customer can then choose between already saved credit cards for future purchases and does not need to enter all data repeatedly.

SHIPPING INFORMATION

test
Teststreet 2
8406 Winterthur
Switzerland

PAYMENT

 Credit Card
  iDEAL

Alias

XXXXXXXXXXXX0007 (10/16) ▾

Select your CreditCard

[Continue to next step](#)

or [Go back](#)

Figure 6.1: Alias Manager usage with Innocard.

Please note

In order to use the Alias Manager, the respective option has to be enabled by Innocard. Please contact the Innocard support to do so.

6.5 Cron Job Setup

Cron Jobs are necessary for the time-controlled plug-in functions (e.g. Update Service, deleting pending orders, etc.) and have to be set up in Drupal in order to use these options. Especially the Update function is useful since it allows you to fetch transaction changes by Innocard via the API. Please note that additional options might have to be enabled.

This module uses the standard Drupal Cron. Information on the setup can be found at <https://www.drupal.org/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.

7 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.

7.1 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

8 Errors and their Solutions

You can find detailed information under <http://www.sellxed.com/en/fag>. Should you not be able to solve your problem with the provided information, please contact us directly under: <http://www.sellxed.com/en/support>

8.1 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](#).

9 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.

You need to have at least one module active, which stores the Drupal 'watchdog' messages. We recommend you activate the "Database logging" module, which stores this messages into the database. This module is available per default.

9.1 Log Levels

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

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

9.2 Log Location

The log messages are visible, if the "Database logging" module is active, in the Drupal backend under the menu item **Admin > Reports > Recent log messages**.

10 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.

10.1 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 Drupal initialized in your script, this is required for the database connection.

```
define('DRUPAL_ROOT', getcwd());  
require_once DRUPAL_ROOT . '/includes/bootstrap.inc';  
drupal_bootstrap(DRUPAL_BOOTSTRAP_FULL);
```

Include the modules Util class.

```
module_load_include('php', 'commerce_innocardcw', 'includes/classes  
/InnocardCw/Util');
```

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

Load the transaction by Id:

```
$transactionById = InnocardCw_Util::loadTransactionById($transactionId);  
$transactionObject = $transactionById->getTransactionObject();
```

Load the transaction by the external Id:

```
$transactionByExternal = InnocardCw_Util::loadTransactionByExternalId  
($externalId);  
$transactionObject = $transactionByExternal->getTransactionObject();
```

Load transactions by order Id:

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