Magento Concardis PayEngine Extension

Contents

1 Installation **4**

1.1 Requirements 4

1.2 Installation process 4

2 Configuration **5**

2.1 PayEngine Backend 5

2.1.1 Payment methods 5

2.1.2 User management 5

2.1.3 Technical Information 5

2.2 Magento Back Office 9

2.2.1 Payment method configuration 11

2.2.1.1 Open Invoice 11

2.2.1.2 InterSolve 11

2.2.1.3 Cc with Alias Manager 11

2.2.1.4 Configuration for Kwixo payment methods 12

2.2.1.5 Paypal with Paypal Seller protection 16

3 Use **17**

3.1 General 17

3.1.1 InterSolve 17

3.1.2 Credit card with stored payment information (Alias Manager) 17

3.1.3 Direct Debit payments 20

3.1.4 Kwixo payment methods 20

3.2 Shipment receipt 21

3.3 Invoicing 22

3.4 Cancellations 22

3.5 Refunds / Credit Memo 23

3.6 Additional Information 24

3.7 Trigger Payments in Magento Backend (PayEngine MOTO eTerminal) 25

3.7.1 Payment using existing PayEngine-Transaction 25

4 Notes **26**

4.1 General 26

4.2 Multistore 26

4.2.1 Example - different domains using the same PayEngine account 26

4.2.2 Example - Store code in url using the the same PayEngine account 27

4.2.3 Example - different domains and different PayEngine accounts 27
4.2.4 Example - different domains using the same PayEngine account 28
4.2.5 Example - Store code in url and different PayEngine accounts 29
5 Workflow 31
5.1 Payment 31

1 Installation

1.1 Requirements

- PHP 5.2+
- An active PayEngine Account
- One of the following Magento versions
    - Community Edition 1.7 - 1.9
    - Enterprise Edition 1.12 - 1.14

1.2 Installation process

Copy the "Extension Key" of the Magento Commerce website

1. Copy the "Extension Key" of the Magento Commerce website
2. Login to your Magento backend
3. Go to "System" -> "Magento Connect" -> "Magento Connect Manager"
4. Login to your "Magento Connect Manager"
5. Select the tab "Extensions"
6. Copy the key of the Magento Commerce website into the field "Paste extension key to install" and click on "Install"
7. Under some circumstances you need to select "Proceed" to confirm the installation.
8. In the black installation console will the message "Package ... installed successfully" be shown. The module is now correctly installed.

2 Configuration

2.1 PayEngine Backend

Login to your PayEngine back office and enter the following settings in the menu.

2.1.1 Payment methods

Depending on the PayEngine subscription you have selected it might be necessary to activate further mandatory options in your PayEngine account. These options are: OPC, DirectLink (New Payment), DirectLink (Maintenance), D3D and Alias Manager (more info about in the chapter Cc with Alias Manager). Please check if these options are activated by default and if this is not the case please contact PayEngine for the activation.

The Magento PayEngine extension might not work in conjunction with other extensions. To avoid malfunction of the PayEngine extension we recommend to disable any other extension for test purposes.

2.1.2 User management

The shop needs a separate API-User to communicate with PayEngine. You can create the API-User via "Configuration"-> "Users" select "New User". Select a USERID (for example: `APIUSER`) and fill in the name and an existing e-mail address. The field "External USERID" can be left empty. Select the profile "Administrator". Select the Access Type "API only". Write down the USERID and password which are shown on your screen.

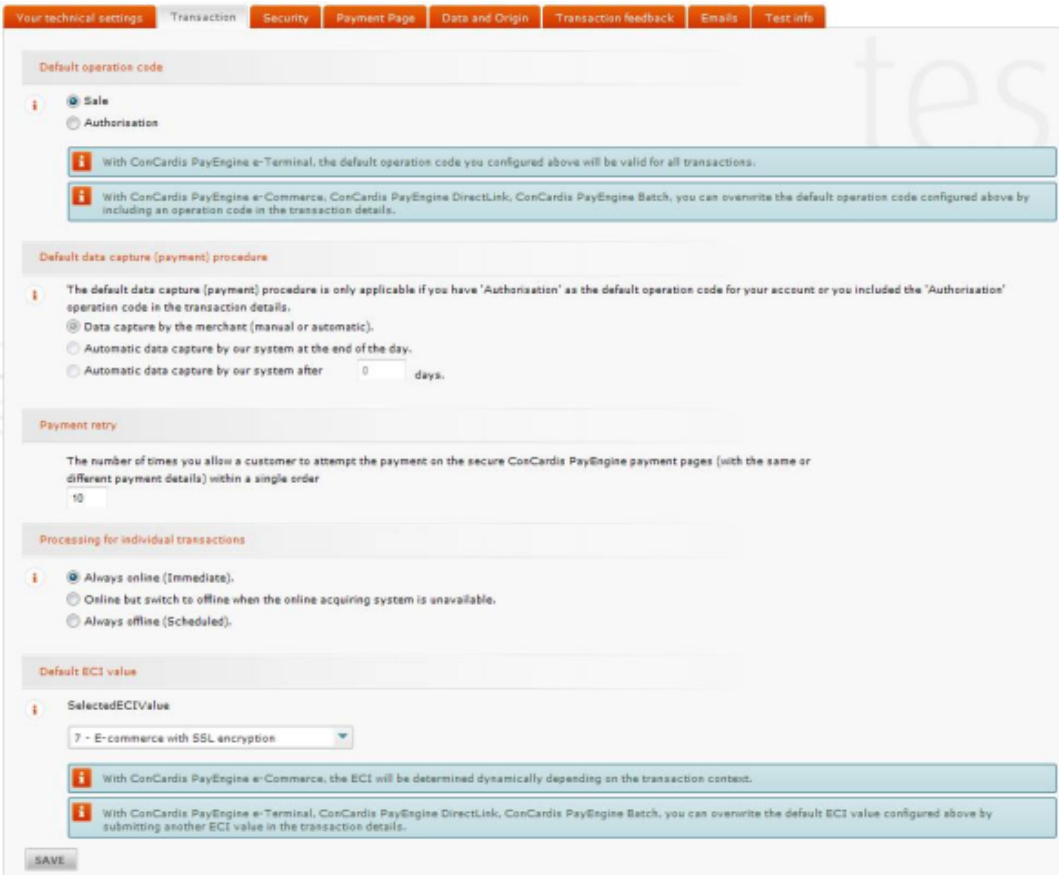
Keep in mind that the USERID and the password may not contain any special characters.

If PayEngine creates a password with a special character, please change the password:

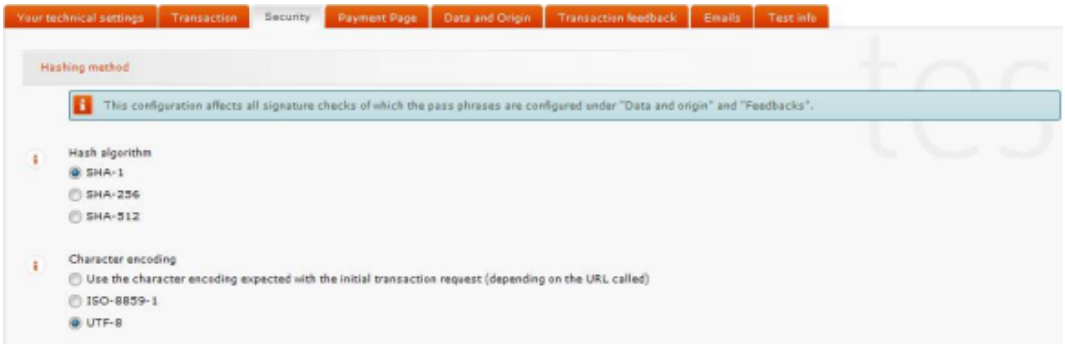
- Select Password in the menu on the left
- Select the USERID of the API-User
- Change the password to a password without any special characters

2.1.3 Technical Information

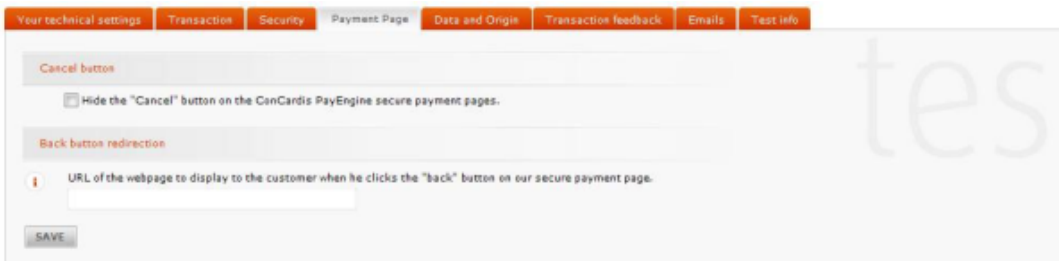
You have to configure the Technical settings in the PayEngine Back office via "Configuration" -> "Technical Information" -> "Global Transactionparameters": Select "Direct sale" or "Authorisation" and change the "Payment retries" value if you want to.



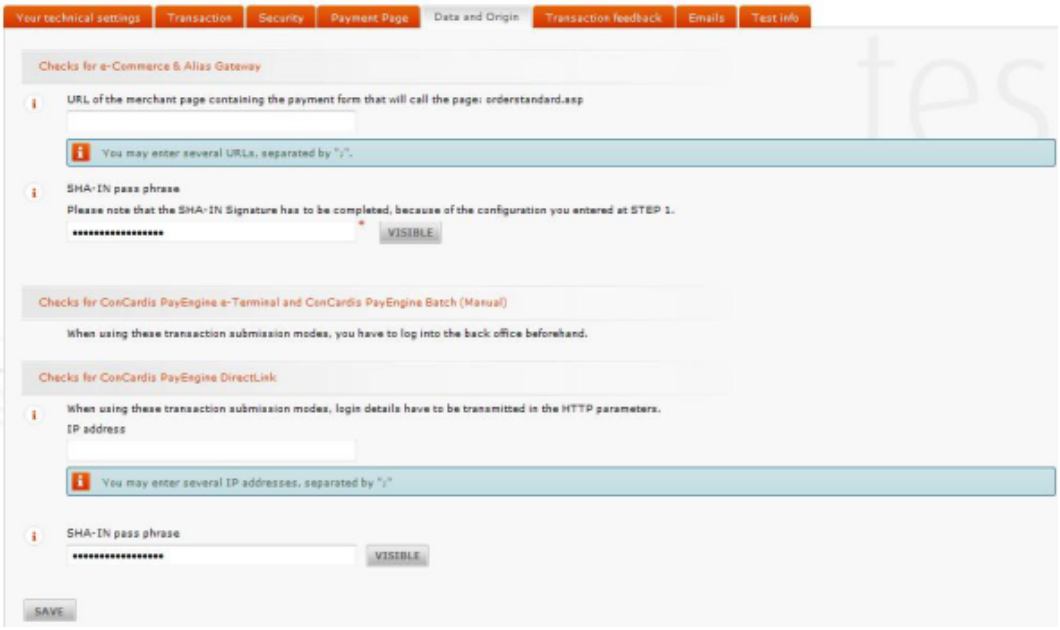
Then go to “Global security parameters” and select the SHA-1 hashing algorithm and UTF-8 character encoding.



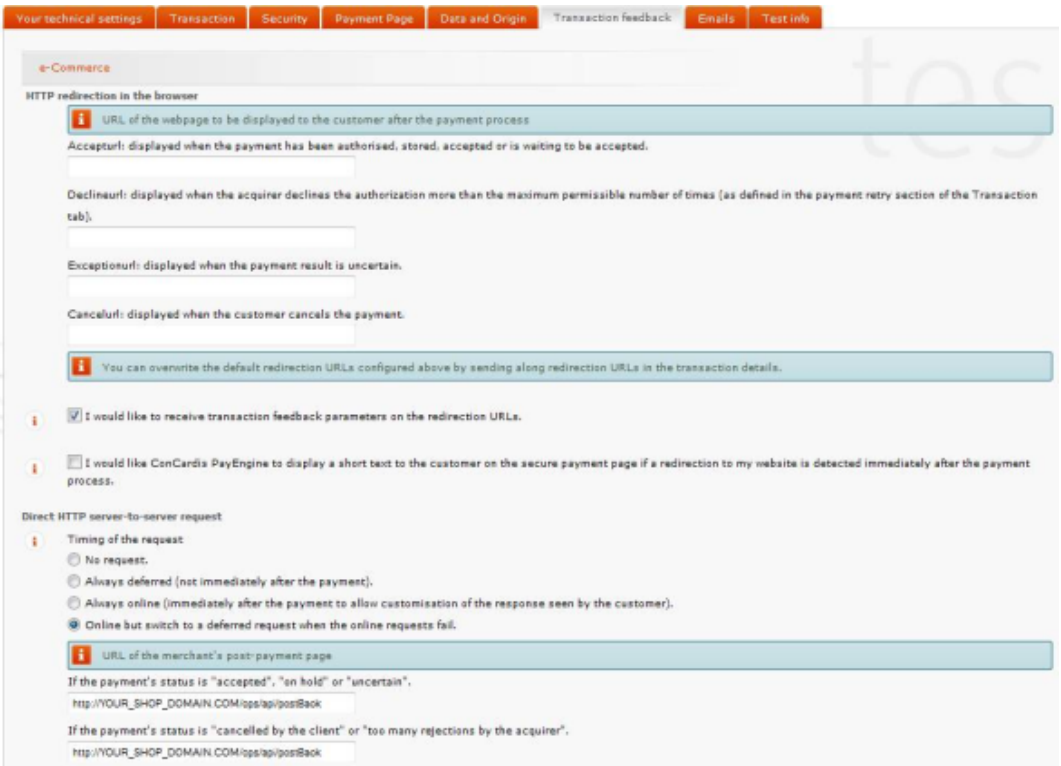
The settings in the “Payment Page layout” tab are optional.

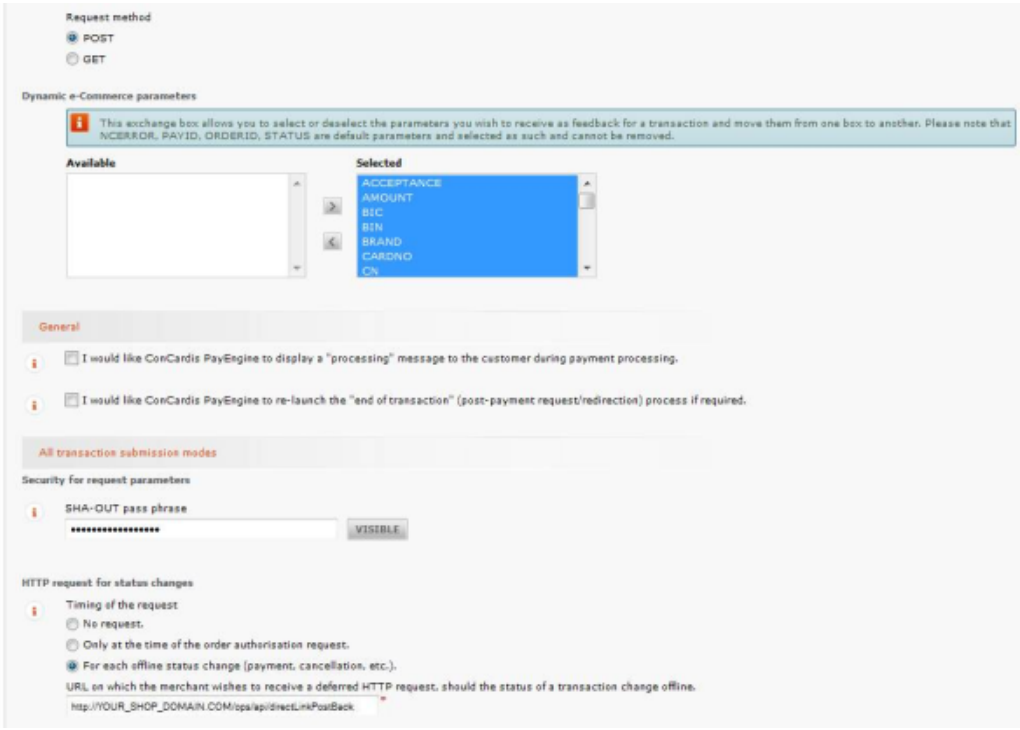


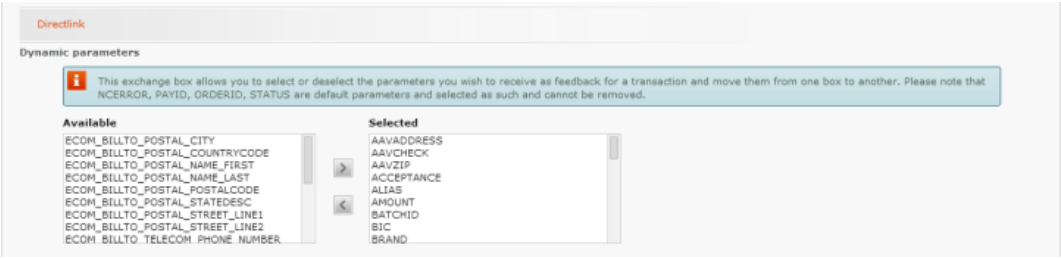
In the tab “Data and origin verification” do you need to fill in the SHA-IN passphrases. **Please keep in mind that you need to fill the SHA-IN pass phrase for e-Commerce _and_ for DirectLink.** SHA-IN and SHA-OUT keys have to be alphanumeric only and both keys should have the same value.



Please take over the following settings:

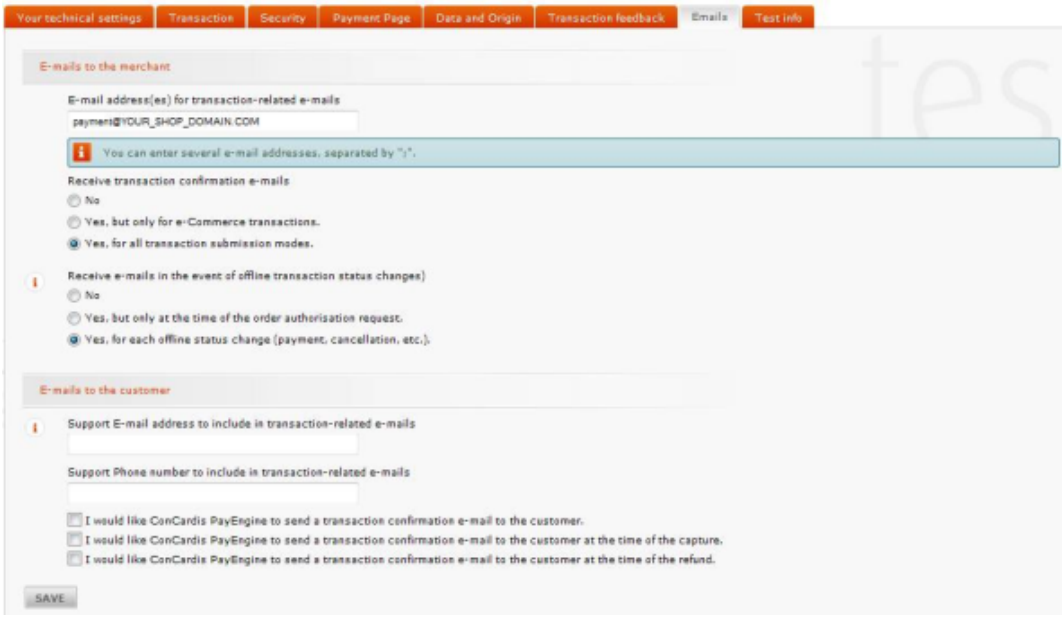




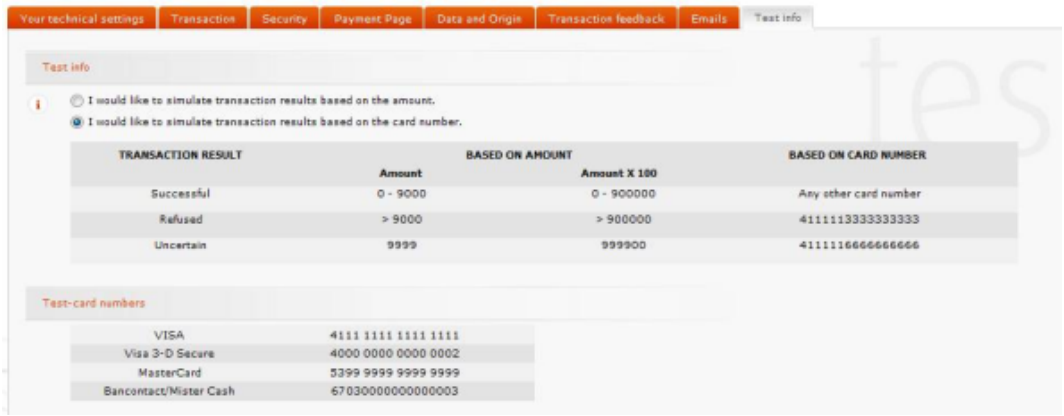


Please don’t forget to configure the SHA-OUT pass phrase. Please note: Refrain from selecting the ECOM_BILLTO or ECOM_SHIPTO parameters since these can easily result in transmission errors.

Please use the following settings in the “Transaction e-mails” tab.



Configure the following setting in the “Test info” tab.



2.2 Magento Back Office

After the installation of the module, you now go to the Magento back office -> System -> Configuration -> Payment services. There you will see the option “PayEngine Account”. Click and fill in the following information:

Setting	Description
PSPID	Fill in the PSPID of your PayEngine account.
SHA-IN Pass phrase	Fill in the SHA-IN passphrase that you have setup in your PayEngine back office -> Technical information -> Data and Origin verification tab.
SHA-OUT Pass phrase	Fill in the SHA-OUT passphrase that you have setup in the PayEngine back office -> Technical information -> Transaction feedback tab.

Gateway URLs	The gateway URLs are already correctly configured. However, when you switch from the TEST environment to the PRODUCTION environment, you need to replace these TEST URLs by PRODUCTION URLs, which are shown in the comment of this configuration fields. Please note that the Direct Query feature must be enabled in the PayEngine's backend in order to use the DirectLink Maintenance API properly.
API User / API Password	Fill in the API details that you have created in the PayEngine Back office.
Payment action	Please keep in mind that this settings needs to be equal to the setting in the PayEngine back office -> Technical information -> Global transaction parameters -> Default operation code.
Payment template	When you select `Magento`, the payment page will be in the same layout as your shop. When you select `PayEngine`, all the fields that follow with reference to the PayEngine template are then mandatory to fill in.
Layout of payment methods	Here you can select the way that the payment methods are shown in the checkout page.
Order reference in case of redirect payments	With this option can be specified if the Orders increment ID or the Quote ID should be transferred as a payment reference for the orders to PayEngine. The Order ID is the number in format 100000001, which is listed in the backend order grid in column "Order #". In case the "Order ID" is selected this number will be transferred during the payment process to PayEngine. Because of compatibility reasons a hash sign is added before the Order ID. If "Quote ID" is selected this number will be transferred to PayEngine. The Quote ID is also the only mode for all inline payment methods like credit card (if activated) and direct debit and the Kwixo payment methods.
Order reference in case of inline payments	With this option can be specified if the Orders increment ID or the Quote ID should be transferred as a payment reference for the orders to PayEngine. The Order increment ID is the number in format 100000001, which is listed in the backend order grid in column "Order #". In case the "Order ID" is selected this number will be transferred during the payment process to PayEngine. Because of compatibility reasons a hash sign is added before the Order's increment ID. If "Quote ID" is selected this number will be transferred to PayEngine. The Quote ID is also the only mode the Kwixo payment methods.
Show Quote ID in the order grid	If activated the Quote ID will be added as a column in the backend order grid.
Submit extra parameters	If activated additional parameters are transmitted to PayEngine in order to make use of the fraud detection or the Paypal Seller protection.

Device ID	Enables the tracking of the customer for fraud detection purposes
Debug	When this option is selected, all requests and answers from PayEngine will be checked. We recommend you to activate this option only when you are in test mode and not in production mode.

Please note: The PayEngine extension might not work correctly, if the Store Codes were added to the URLs (System->Configuration->General->Web)! It is therefore advised not to use this option.

2.2.1 Payment method configuration

After that you should configure your payment methods at System -> Configuration -> Payment Methods.

Please remember, that you should only activate payment methods that are activated in PayEngine backend via "Configuration" -> "Payment methods". For activiations and further changes please contact PayEngine customer service .

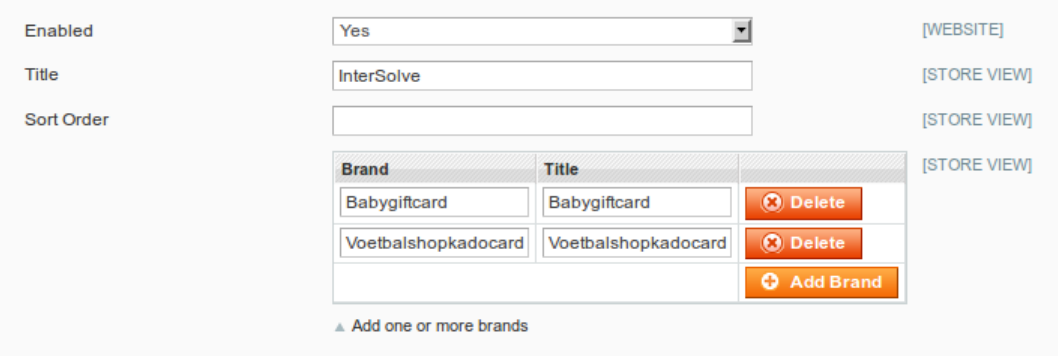
2.2.1.1 Open Invoice

If you enable Open Invoice payments, you must configure your shop to request the customer's birthday and gender. In case of Open Invoice NL, Open Invoice DE or Open Invoice AT also the customers VAT number is required. You can activate these settings at System -> Configuration -> Customer Configuration.

Please note: Please be informed that, if you use Afterpay as acquirer on PayEngine side, it's not possible to perform partial captures with Open Invoice NL.

2.2.1.2 InterSolve

In case of payment method "InterSolve" it is possible to define several brands, which corresponds to its own PayEngines payment method. Please notice, that the brands have to be configured at PayEngine before you can use it for your shop. You can assign a title to this brand, which is shown to your customer at the checkout.



Configuration of InterSolve brands

2.2.1.3 Cc with Alias Manager

The Alias Manager is a synonym for the PayEngine Alias Manager. Your customer can reuse his credit card payment information if this feature is activated. Payment information can be saved only for credit card types with inline payment.

Configuration for Alias Manager

Setting	Description
Enabled Alias Manager	If enabled the customer has the option to saves his payment information to reuse the saved information for further payments.
Show Alias Manager information for guests	If this option is enabled a hint informs your guest customers about the advantages of saving his credit card details, since it's not available to customers who are not logged in.

Important:

For a proper usage of this feature you need assure that the **cron.php**, which is located in the Magento's root directory is executed periodically. On a *nix or Linux based System you'll have to add one of the following entries to your crontab:

*/5 * * * * /bin/sh /absolute/path/to/magento/cron.sh

or

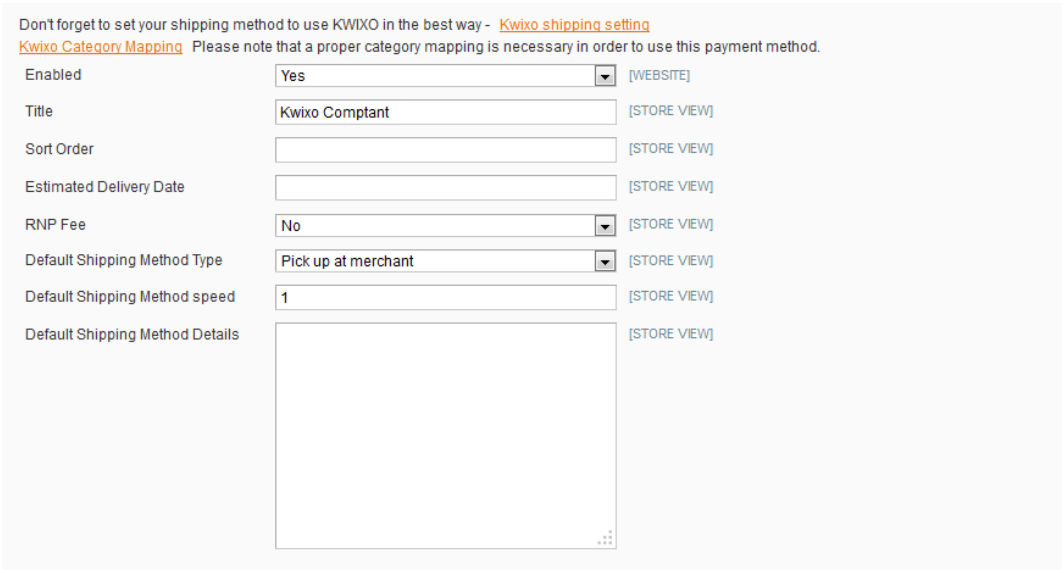
0,5,10,15,20,25,30,35,40,45,50,55 * * * * /bin/sh /absolute/path/to/magento/cron.sh

For further details of the usage of the credit card with Alias Manager please refer Credit card with stored payment information (Alias Manager)

2.2.1.4 Configuration for Kwixo payment methods

In order to use the Kwixo payment methods such as Kwixo Credit, Kwixo Apres Reception and Kwixo Comptant you will have to adjust further settings which are described in this section.

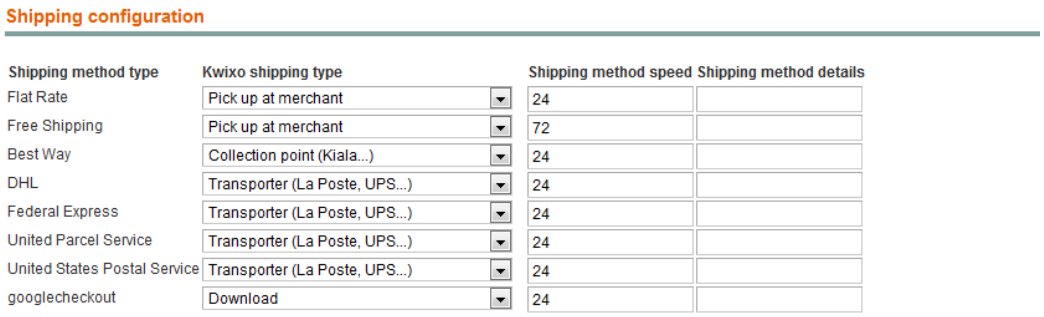
The following screenshot shows the configuration options of the Kwixo payment methods which are described in the table below.



Configuration of a Kwixo payment method

Setting	Description
Enabled	If you want to offer this payment method to your customers then choose 'yes'.
Title	If you want to use a custom title for this payment method, you can enter it into this field.
Sort order	Defines the position for this payment method in the checkout.
Estimated delivery date	Estimated number of days until delivery after the order is paid
RNP fee	You can invoice the RNP fee. To do so, choose 'yes' in this input field.
Default shipping method type	You can choose the default Kwixo shipping method for the orders which are paid with this payment method.
Default Shipping method speed	Defines the default number of hours required for the delivery.
Default Shipping method details	Defines further details for the shipping methods. This setting is optional.

In order to use the Kwixo payment method in the best way, you need to configure a proper shipping method setting. The following screenshot illustrates the setting of the shipping methods:

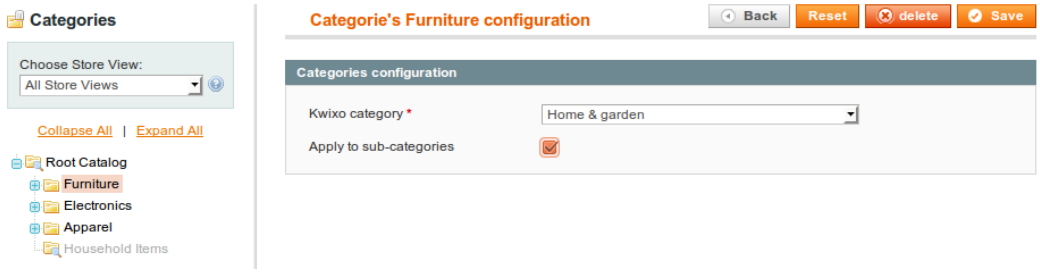


Kwixo shipping settings

Setting	Description
Shipping method type	Defines the Magento's shipping method.
Kwixo Shipping method type	You can choose the kwixo delivery method for the orders which are shipped with the Magento's shipping method.
Shipping method speed	Defines the number of hours required for the delivery.
Shipping method details	defines further default details for the shipping method.

If the customer uses a Kwixo payment method, the settings made here will be transmitted to Kwixo. If no values are set, the default values are used.

If you want to use the Kwixo payment methods, you need to set up a proper configuration mapping for Kwixo categories and the categories of your store. To do so please click on the link 'Kwixo Category Mapping'. The Kwixo category configuration section opens. Here you select one of Magento's categories and you'll see a screen similar to the one shown below.



Configuration for Kwixo categories

In order to assign the selected category to the according Kwixo category you choose one of the Kwixo categories. If you want to apply this assignment to all of the subcategories you need to check 'Apply to sub-categories'. After you've finished your configuration just click 'save' and the connection between the Magento category and Kwixo category is set up properly. The screenshot shows an example configuration for Magento's Furniture-category which is mapped to the 'Home & Garden'-Kwixo category.

To edit or delete the setting please open the Kwixo category configuration and choose the Magento category for which you want to edit or delete the Kwixo category mapping. In case of editing the Kwixo category mapping you just follow the same steps for creating the Kwixo category configuration. If you want to delete the setting just click on delete and the setting is removed. Please keep in mind that a Kwixo category is required for payments using the Kwixo payment methods, so please make sure all your Magento categories have an according Kwixo category.

2.2.1.5 Paypal with Paypal Seller protection

If you want to use Paypal with the Paypal seller protection, you need to adjust the following settings:

Setting	Description
Configuration -> General -> State options -> State is required for	Please select the countries for which the state is mandatory according to the requirements of Paypal's seller protection.
Configuration -> Payment Services -> PayEngine account -> Submit extra parameters	Please set this option to 'Yes' so the neccessary parameters will be transmitted to PayEngine.

3 Use

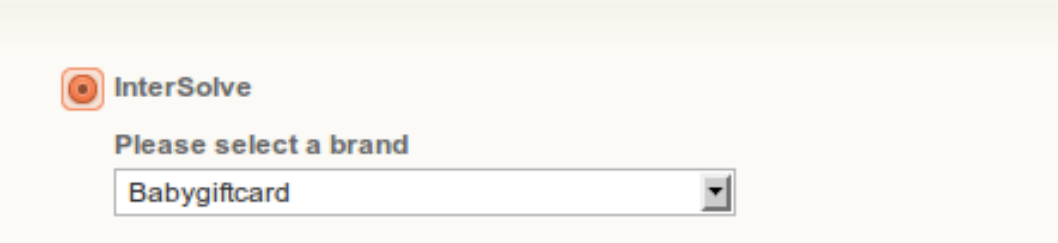
3.1 General

This module works like most Magento connections. If you have selected you would like to process credit cards, they will be processed via the Magento platform. However, when 3D Secure is activated, the cardholders will be redirected to an PayEngine payment page. This is also applicable for any other payment method where the account holder's details need to confirmed by the issuer.

When the transaction is processed, you can see the transaction in your Magento back office via Sales -> Orders.

3.1.1 InterSolve

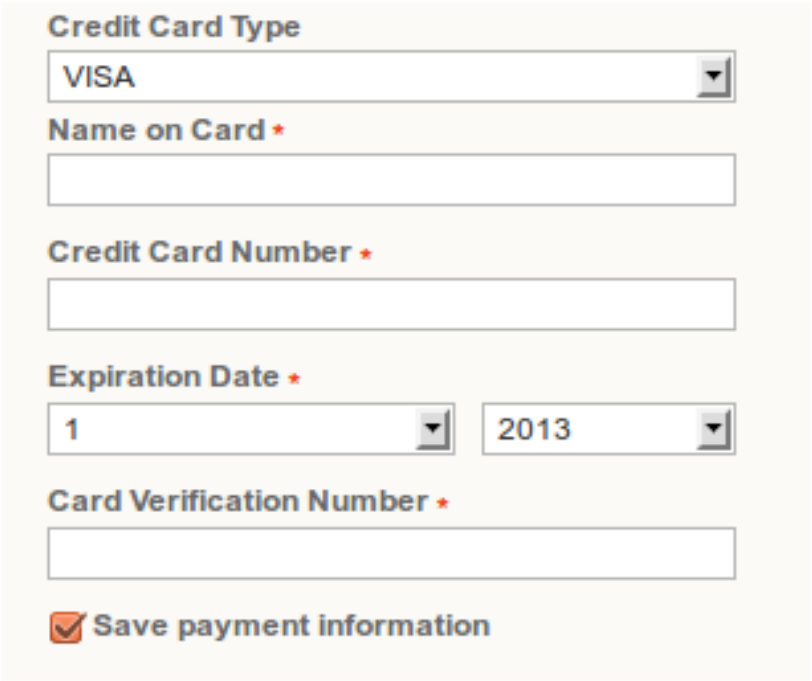
In case of Intersolve payment you can define several brands, which can be selected by the customer during the checkout process. The selected brand will be transmitted to PayEngine. There is no selection available, if only one brand is defined. In this case the brand is displayed to the customer and will be transmitted to PayEngine.



Selection of InterSolve brands

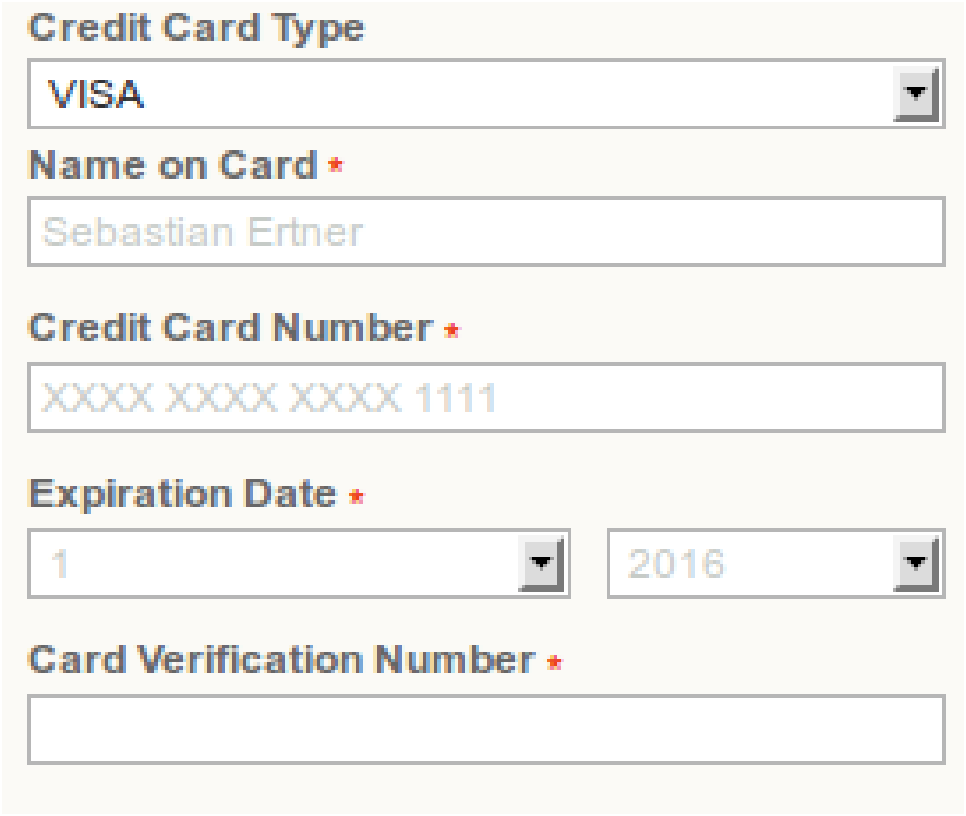
3.1.2 Credit card with stored payment information (Alias Manager)

When using the payment method "credit card" and the activated feature Alias Manager a logged in customer gets his stored credit card information shown, if he saved it in a previous step. If the customer is not logged in a hint is displayed that informs the customer about the possibility to store his credit card data for reuse. In the hint text a link is shown, if the customer clicks that link he will get directed to the "checkout method" step and the checkbox "register" is checked. The selection of the displayed payment information depends on billing address and shipping address in order to prevent abuse. If your customer wants to save his payment information, he has to select the checkbox 'save payment information' on credit card payment section.



saving the credit card data

ConCardis



prefilled credit card form

There is also the posibility to update the stored credit card data. If there are stored credit card data the fomular gets preffiled and the input fields are greyed out. If the customer wants to update his saved credit card data, he needs to click in one of the greyed out input fields. If he does so, all input fields will be cleared and the customer has to enter the new credit card data. To save this new credit card data, the checkbox "Save payment information" has to be checked. If it's not checked the new payment information is used for the payment but it's not stored and the old payment information is still available for further use.

In addition your customer can list and delete his payment information in the section 'payment information' his account area. This opportunity is also available in the customers administration section in the Magento backend. Select the customer and then select 'payment information' for a list of stored payment information.

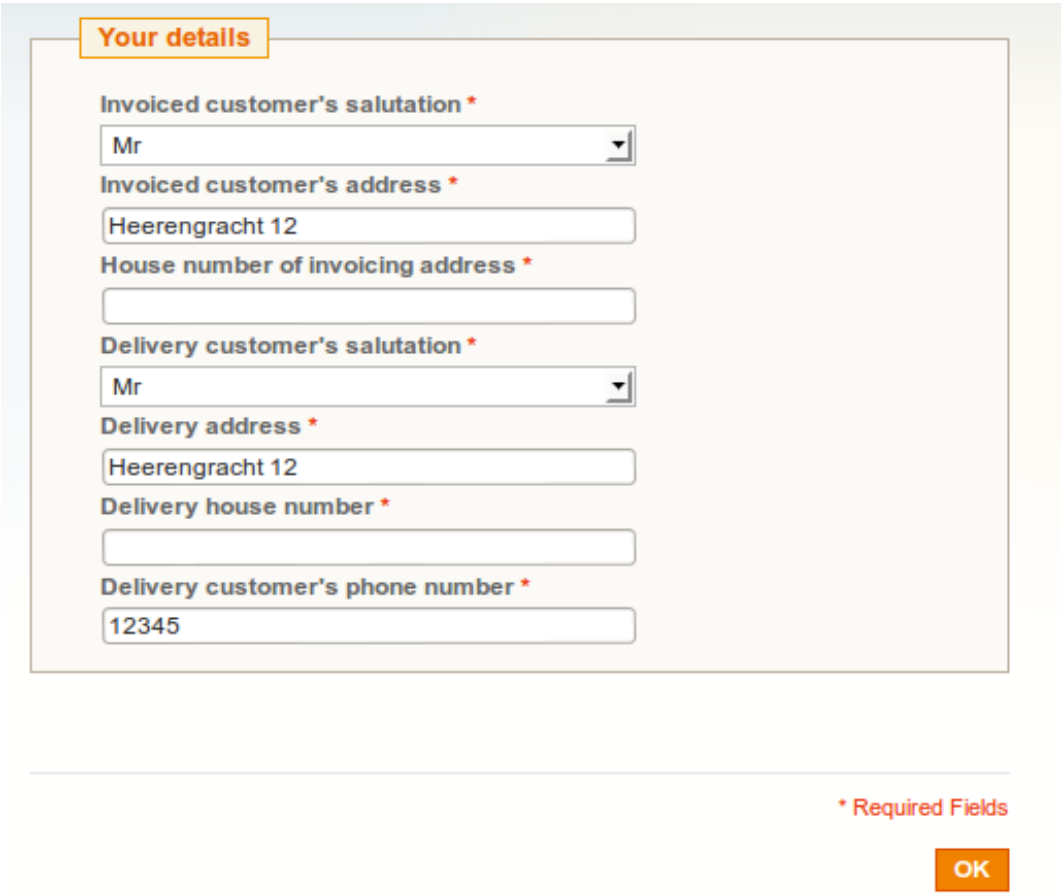
3.1.3 Direct Debit payments

You can enable direct debit payments for Austria, Germany and the Netherlands which allows you customer to use his account data for payment. In case for Austrian and German direct debit payment methods your customer is asked for for his account number (Kontonummer) and the bank code (BLZ). If your customer uses the dutch direct debit payment method he's only asked for his account number. Furthermore the german and the dutch direct debit payment methods support the usage of IBAN and BIC (dutch direct debit only). Please note if both alternatives, IBAN (and BIC) and the according account data are provided, the IBAN and BIC is used for processing the payment.

You can also use the direct debit for MOTO transactions (aka creating an order in the backend), which behaves the same way as described above.

3.1.4 Kwixo payment methods

After placing the order your customer is redirected to PayEngine. In case that not all parameters could be detected your customer has the option to enter the missing data into the form which is shown in the screenshot below. If all parameters are present your customer is redirected to PayEngine after submitting the form.



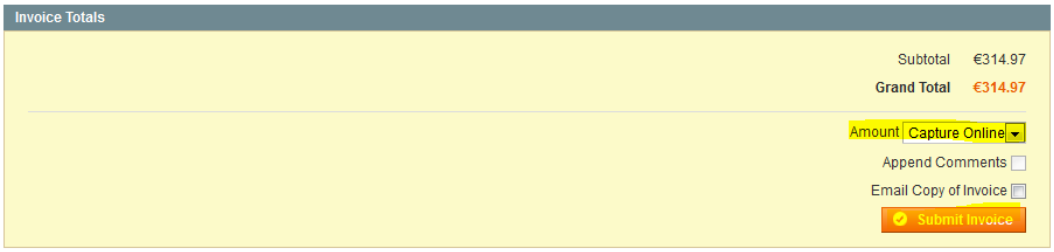
additional information for Kwixo payment methods

3.2 Shipment receipt

The creation of a shipment receipt is made without a connection to the payment service and therefore behaves like Magento-Standard.

3.3 Invoicing

Whether you have choose "Authorization" as "Payment Action" in the Magento configuration, an invoice has to be created for payment methods like credit cart to trigger the capture process by PayEngine. For this purpose click in the order view on button "Invoice" and choose "Capture Online" as amount and finish the invoice creation with "Submit Invoice".



Depending of the payment method the invoice will be created instantly or after a delay by a gateway-response triggered by PayEngine.

3.4 Cancellations

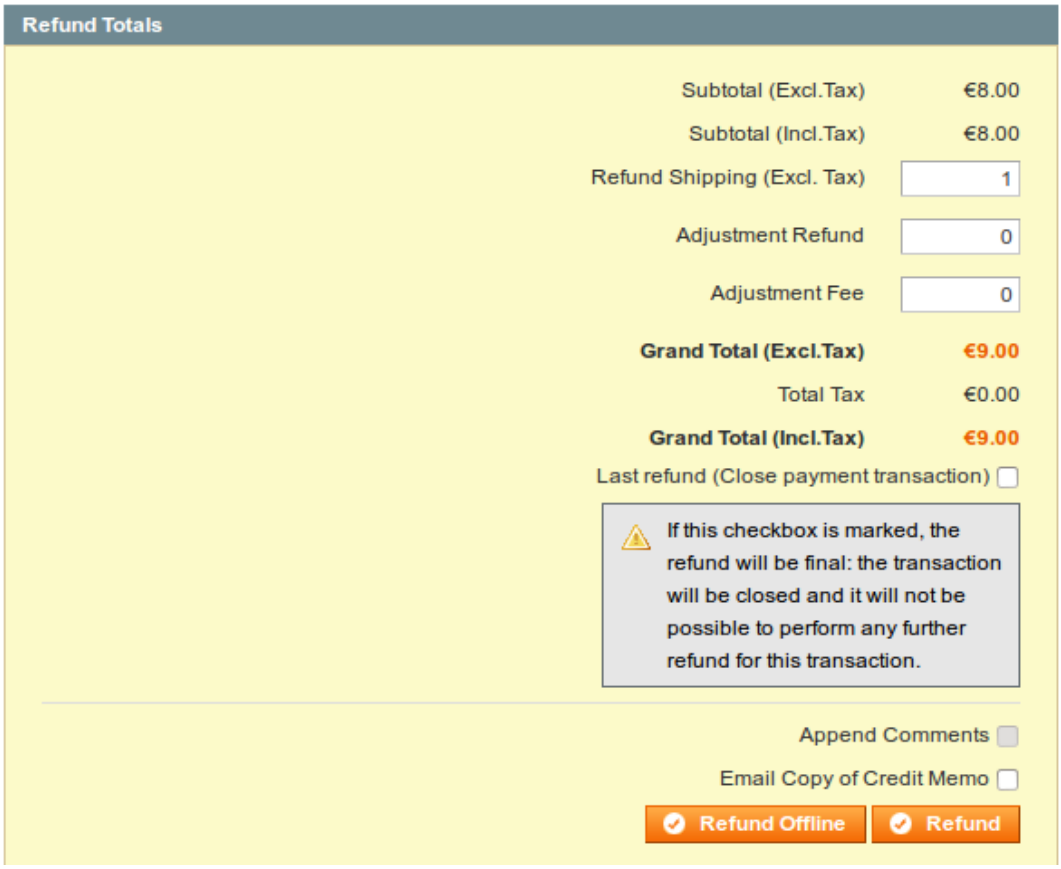
To proceed a cancel (in case the authorisation was successful) use "Void" instead of "Cancel" in the order view. "Cancel" could not be used because of gateway workflow reasons. Depending of the payment method the cancel would be proceeded instantly or after a delay by a gateway-response triggered by PayEngine.

If the cancellation was successful processed by PayEngine it can be necessary in some Magento-versions to click on "Cancel" in the order view to set the correct cancelled status in Magento.

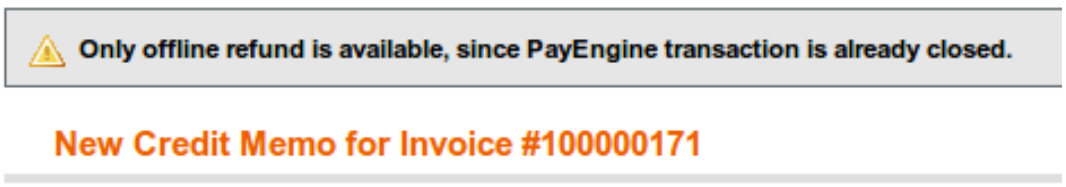
The order can be cancelled directly by the button "Cancel", if the order has the state "Pending Payment" and PayEngine-status 0 or empty. In this case no gateway request to PayEngine will be sent and the product amount will be put back in stock. *Warning:* Please check the payment status in the back-office of PayEngine before. By cancelling this order you won't be able to update the status in Magento anymore.

3.5 Refunds / Credit Memo

To create a refund firstly you have to open the order view and open the "invoices"-tab in the left menu panel. Choose the invoice you want to create a refund for. In the "invoice view" click on button "Credit Memo". In this form you can adjust the amount to refund and you can also decide if you want to close the PayEngine transaction. In order to achieve this, click on the checkbox 'Close PayEngine transaction'. Finally confirm the refund by clicking "Refund" button and not "Refund Offline"! The figure below shows the corresponding input fields:



You can repeat this procedure until the PayEngine transaction is closed, that means either the complete amount is already refunded or you have decided to close the PayEngine transaction manually by the checkbox 'Close PayEngine transaction' before. In this case only offline refunds are available and by creation of the Credit Memo the following message appears:



Depending of the payment method the refund would be proceeded instantly or after a delay by a gateway-response triggered by PayEngine.

The button "Credit Memo" in the "order view" just creates an internal refund and triggers no refund action by PayEngine.

For some payment methods (e.g. iDEAL), you will have to enable refunds in your PayEngine account.

3.6 Additional Information

In the “Payment Information” box on the “Order View” Page (-> Sales -> Orders -> Click “View” -> Select “Information”), you can find additional payment related information about the order. The information includes payment method type, e.g. VISA credit card, Payment ID, recent payment status and currency.

Secondly, you will find in the lower part of the “Order View” Page in the "Comments History" box a chronologically sorted course of payment status changes. This additional information helps to track payment process changes, e.g. to allow you to verify that a credit card authorisation has changed to a payment capture.

3.7 Trigger Payments in Magento Backend (PayEngine MOTO eTerminal)

According to PayEngine MOTO eTerminal some payment methods are available for order creation in Magento backend, too. Since there are no redirect payment methods allowed, you can't use such methods. Even usage of 3D secure ("Verified by VISA", "MasterCard Secure Code") is not possible.

Configuration for MOTO eTerminal features for credit cards and direct debits

Hint: Magento sends some data to the shop immediately after input. So it may appear, that you have to click on the order submit button twice.

3.7.1 Payment using existing PayEngine-Transaction

Using this payment method, which is only available in the Magento backend, you can enter the pay id of an existing PayEngine transcation directly. In this case no request is triggered to PayEngine and all state changes have to be done manually.

4 Notes

4.1 General

No credits in PayEngine Back office

Only send your credit requests via the Magento back office.

3D-Secure (MasterCard, SecureCode and Verified by Visa) must be activated

Your account must accept 3D-Secure.

4.2 Multistore

The module is suitable to be used in Magento-Multistores with

- multiple websites
- multiple stores
- multiple domains
- multiple PayEngine accounts

In order to use multiple PayEngine-accounts, you have to follow the steps mentioned in sections PayEngine Backend and Magento Back Office for each PayEngine-account and website or store.

The most important part of the Multishop configuration concerns the correct setup of the response-urls in PayEngine. You can find this configuration-fields in the PayEngine-backend at "Configuration" -> "Technical information" -> "Transaction feedback"

- Direct HTTP server-to-server request
- HTTP request for status changes.

The correct configuration of this urls should be clarified with the help of the following four examples.

4.2.1 Example - different domains using the same PayEngine account

The Multistore consists of two stores which are assigned to different websites.

- Store A operates under domain
  http://www.my-shop-a.com/
- Store B operates under domain
  http://www.my-shop-b.com/

The correct adjustment in the PayEngine-backend would be the following:

- Direct HTTP server-to-server request:
  http://www.my-shop-a.com/ops/api/postBack

- HTTP request for status changes:

  http://www.my-shop-a.com/ops/api/directLinkPostBack

The feedback will be processed correctly for store B also if the response is send only to the url of store A.

4.2.2 Example - Store code in url using the the same PayEngine account

The Multistore consists of two stores which are assigned to the same website. The example would be also valid in case that both stores were assigned to different websites.

- Store A operates under domain

  http://www.my-shop.com/shopa/
- Store B operates under domain

  http://www.my-shop.com/shopb/

The correct adjustment in the PayEngine-backend would be the following:

- Direct HTTP server-to-server request:

  http://www.my-shop.com/shopa/ops/api/postBack
- HTTP request for status changes:

  http://www.my-shop.com/shopa/ops/api/directLinkPostBack

As before the feedback will be processed correctly for store B also if the response is send only to the url of store A.

4.2.3 Example - different domains and different PayEngine accounts

The multistore consists of two stores which are assigned to different websites and use two diffenent PayEngine accounts.

- Store A operates under domain

  http://www.my-shop-a.com/
- Store B operates under domain

  http://www.my-shop-b.com/

The correct adjustment for Store A in the PayEngine-backend would be the following:

- Direct HTTP server-to-server request:
  http://www.my-shop-a.com/ops/api/postBack
- HTTP request for status changes:
  http://www.my-shop-a.com/ops/api/directLinkPostBack

The correct adjustment for Store B in the PayEngine-backend would be the following:

- Direct HTTP server-to-server request:
  http://www.my-shop-b.com/ops/api/postBack
- HTTP request for status changes:
  http://www.my-shop-b.com/ops/api/directLinkPostBack

4.2.4 Example - different domains using the same PayEngine account

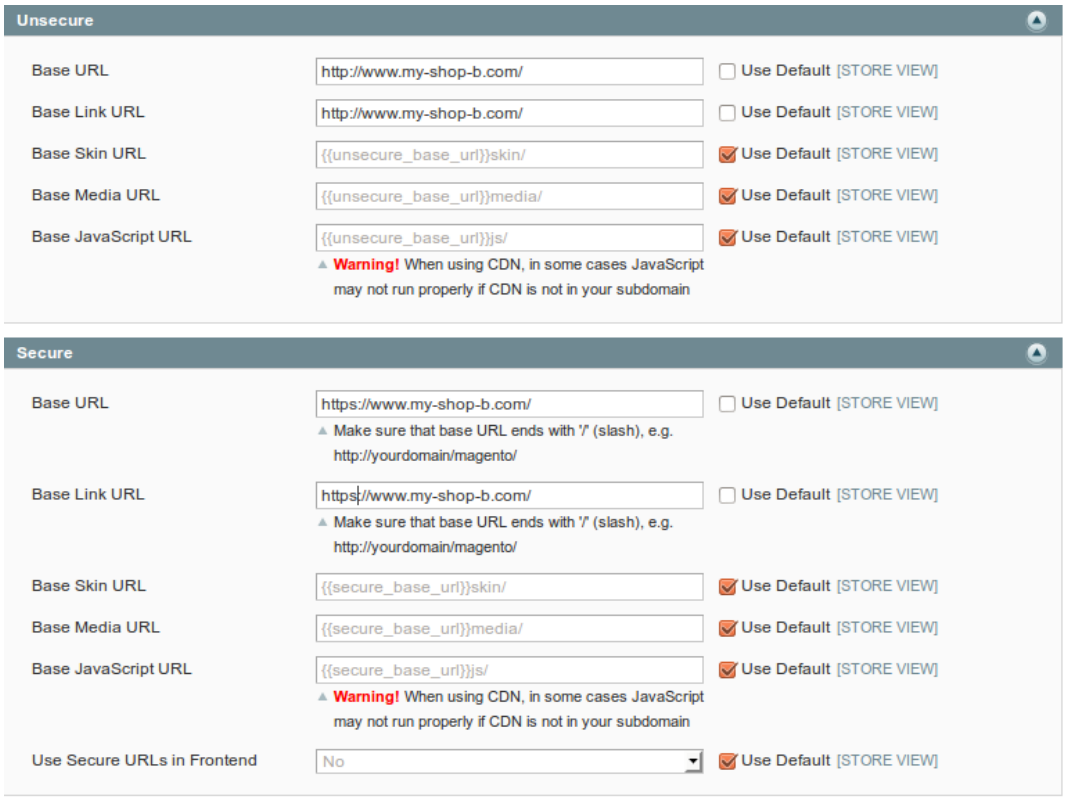
The multistore consists of two stores which are assigned to different websites and the same PayEngine account.

- Store A operates under domain
  http://www.my-shop-a.com/
- Store B operates under domain
  http://www.my-shop-b.com/

The correct adjustment for both stores in the PayEngine-backend would be the following:

- Direct HTTP server-to-server request:
  http://www.my-shop-a.com/ops/api/postBack
- HTTP request for status changes:
  http://www.my-shop-a.com/ops/api/directLinkPostBack

Also you just need to configure Magentos base url for Store B. This must be done under *System -> General -> Web*. Please refer to the following screenshot for the correct settings:



Configuration of the base urls

4.2.5 Example - Store code in url and different PayEngine accounts

The Multistore consists of two stores which are assigned to the same website and use different PayEngine accounts. The example would be also valid in case that both stores were assigned to different websites.

- Store A operates under domain
  http://www.my-shop.com/shopa/
- Store B operates under domain
  http://www.my-shop.com/shopb/

The correct adjustment for Store A in the PayEngine-backend would be the following:

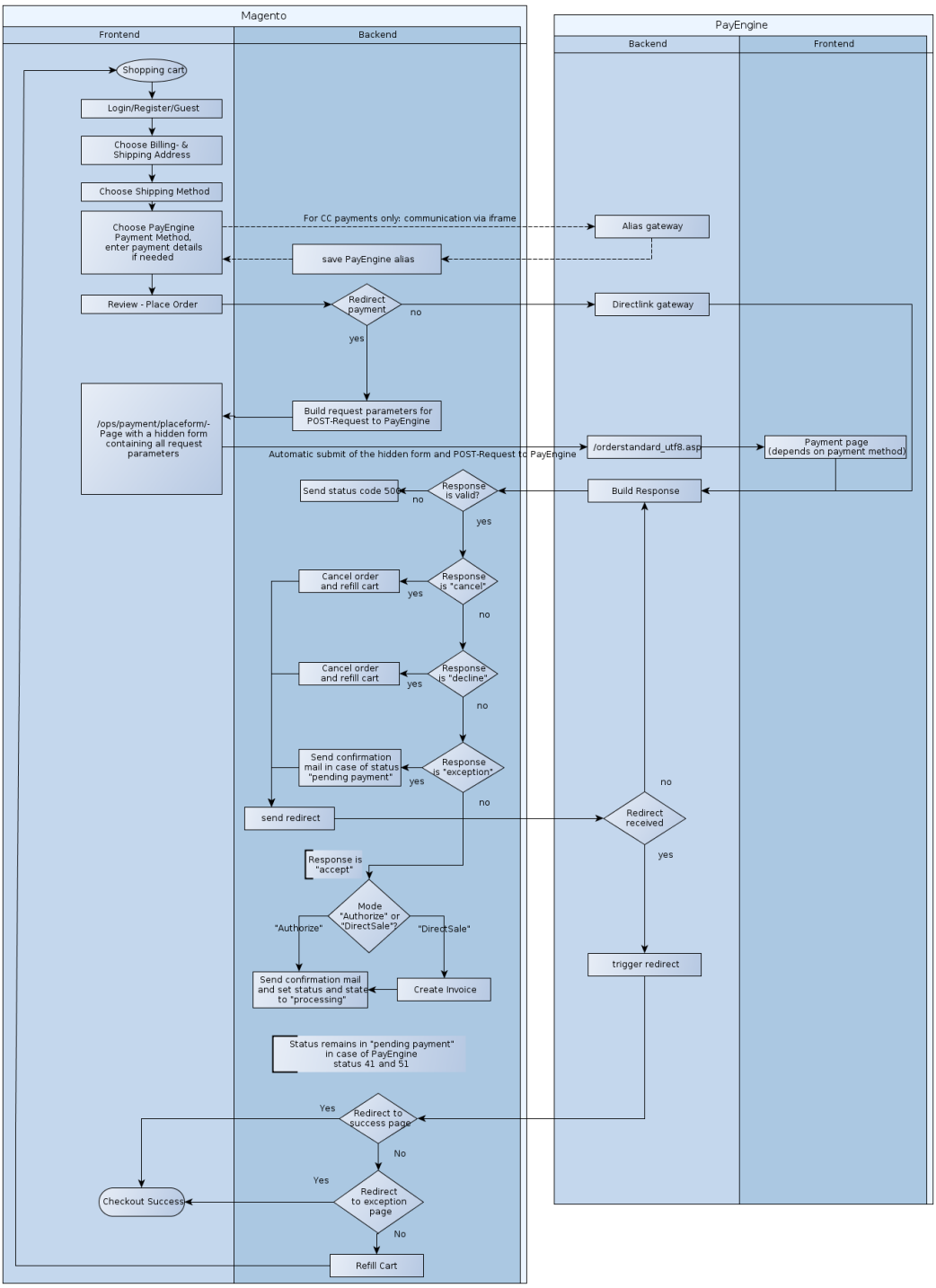
- Direct HTTP server-to-server request:
  http://www.my-shop.com/shopa/ops/api/postBack
- HTTP request for status changes:
  http://www.my-shop.com/shopa/ops/api/directLinkPostBack

The correct adjustment for Store B in the PayEngine-backend would be the following:

- Direct HTTP server-to-server request:
  http://www.my-shop.com/shopb/ops/api/postBack
- HTTP request for status changes:
  http://www.my-shop.com/shopb/ops/api/directLinkPostBack

5 Workflow

5.1 Payment



Payment Workflow of the PayEngine Module