

Resurs Bank
Magento 2 module
Checkout

Contents

[Contents](#)

[Module installation](#)

[Frontend](#)

[Cart](#)

[Shipping methods](#)

[Discount code](#)

[The iframe](#)

[Data syncing](#)

[Order placement](#)

[Admin](#)

[Payment methods](#)

[Callback settings](#)

[Salt-key](#)

[Automatic registration](#)

[Orders](#)

[After-shop settings](#)

[Charging a payment](#)

[Canceling a payment](#)

[Refunding a payment](#)

[Order history](#)

[Logs](#)

[How to activate](#)

[Type of logs](#)

Module installation

1. Navigate to your magento installation directory:

```
cd ~/path/to/your/magento/installation
```

2. Then, inside your magento installation directory, execute the following command:


```
composer require resursbank/checkout
```

This will install the packages `resursbank/checkout` and `resursbank/ecomphp`.

Note that you need the SOAP extension for your version of PHP (for example `php5-soap` or `php7.0-soap`). The extension is used by `resursbank/ecomphp` to communicate with Resurs Bank when handling existing payments.

NOTE: Don't forget to exclude index.php from your base URL, otherwise there may be issues during the checkout process.

Frontend

Item	Price	Qty	Subtotal
 Wayfarer Messenger Bag	56,25 US\$ <small>Excl. Tax: 45,00 US\$</small>	<input type="text" value="1"/>	56,25 US\$ <small>Excl. Tax: 45,00 US\$</small>

Shipping Methods

<input type="radio"/> 5,00 US\$	Fixed	Flat Rate
<input type="radio"/> 0,00 US\$	Free	Free Shipping

[Apply Discount Code](#)

[Apply Discount](#)

1. Hämta dina uppgifter

[Hämta adress](#)
[Eller fyll i dina uppgifter](#)

2. Välj betalsätt

☐ Resurskort

☐ Nytt kort

☐ Faktura

☐ Delbetalning

☐ Kort (test)

☐ Kort

3. Slutför

☐ Spara mina uppgifter för framtida köp
När du sparar dina uppgifter blir det enklare att handla nästa gång du använder Resurs Checkout. Informationen sparas i en cookie i din webbläsare. [Läs mer om hur vi skyddar dina personuppgifter.](#)

Summa: 56.25 kr

[Genomför köp](#)

Genom att lämna information i kassan godkänner jag villkoren för Resurs Checkout och användning av cookies.
Genom att klicka på "Genomför köp" accepterar jag betalningsansvar och godkänner köpvillkoren hos [mageparts](#).

Safe shopping by:

© 2017 Resurs Bank AB (publ) Org.nr 516401-0208, Box 222 09, SE-250 24 Heisingborg, info@resursbank.se, www.resursbank.se
[Användarvillkor Resurs Checkout](#)




This is how the frontend checkout page will look with the module enabled. When the user wants to go to checkout, they are redirected to this page. The shopping cart page can be skipped if specified in the configuration.

As you can see, the page is divided in three parts. At the top is our cart with controls to add/remove/edit individual items; in the middle you can choose shipping methods and apply a

discount code; and at the bottom you fill out your address information and select a payment method.

NOTE: As you change the information on this page, the module will update and prevent the user from interacting with certain things, which can be either the entire page or a specific component as it's updating. Read on to learn more.

Cart

Item	Price	Qty	Subtotal
 Wayfarer Messenger Bag	56,25 US\$ <small>Excl. Tax: 45,00 US\$</small>	<input type="text" value="1"/>	56,25 US\$ <small>Excl. Tax: 45,00 US\$</small>
 			

As you change either the quantity of an item or remove an item from the cart, the module will update the grand total, shipping methods and any applied discount. This will temporarily disable interaction with items, shipping methods and the discount code element.

NOTE: If you change the quantity to 0 (zero) the item will be removed from the cart.

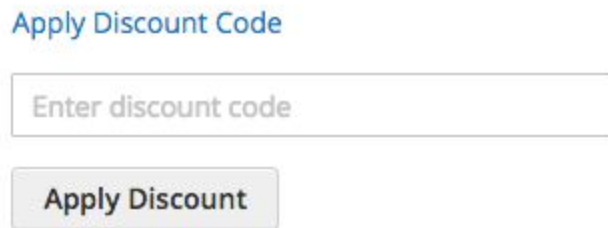
Shipping methods

Shipping Methods

- ☐ **5,00 US\$** Fixed Flat Rate
- ☐ **0,00 US\$** Free Free Shipping

As you change shipping methods the module will update the grand total. This will temporarily disable interaction with shipping methods.

Discount code

A screenshot of a web form for applying a discount code. At the top, the text "Apply Discount Code" is displayed in blue. Below it is a text input field with the placeholder text "Enter discount code". Underneath the input field is a button labeled "Apply Discount".

Apply Discount Code

Apply Discount

When you apply a discount code, the module will temporarily disable the discount code block. When completed, it will display the outcome below the input field: either a red text saying it failed, or a green one saying what amount is withdrawn from the grand total.

NOTE: The outcome is displayed temporarily when failing, but is permanent when successful.

The iframe

NOTE: If you encounter any problems with the iframe, first check that you exclude `index.php` from your base URL.

The iframe communicates with the website through the `window.postMessage()` API. It enables safe cross-origin communication between websites when properly used. We use it to send and receive messages from the Resurs Bank iframe that are about events and changes made to the information within the iframe. Do note that if you're going to use this API to extend or modify any existing functionality, ***you MUST know how to use it properly! Therefore, here are some useful warnings and advice:***

WARNING: Please be aware that improper use of this API can leave holes in your security that attackers can exploit and do things like cross-site scripting attacks!

It is HIGHLY recommended that you thoroughly read the MDN article and get a good understanding of the `window.postMessage()` API before using it:

<https://developer.mozilla.org/en-US/docs/Web/API/Window/postMessage>.

Data syncing

Every part of the module uses AJAX to update the quote object in the client session when you:

- Change the quantity of an item.
- Remove an item from the cart.
- Apply a discount code.
- Select shipping method.
- Enter billing/shipping information.
- Select payment method.

During these AJAX calls, the module will update any affected parts as a result of the change, temporarily disabling them and/or the whole page until the update is done.

Order placement

The order is placed when the client clicks their confirmation button to complete their purchase. If this process would for any reason fail, the user's shopping cart will be rebuilt and as much information as possible will be automatically filled out again to minimize the number of steps required to try placing the order again. Remaining required information needs to be provided by the user manually.

NOTE: The order confirmation email won't be sent after the order has been created (more on this in the section about callbacks).

Admin

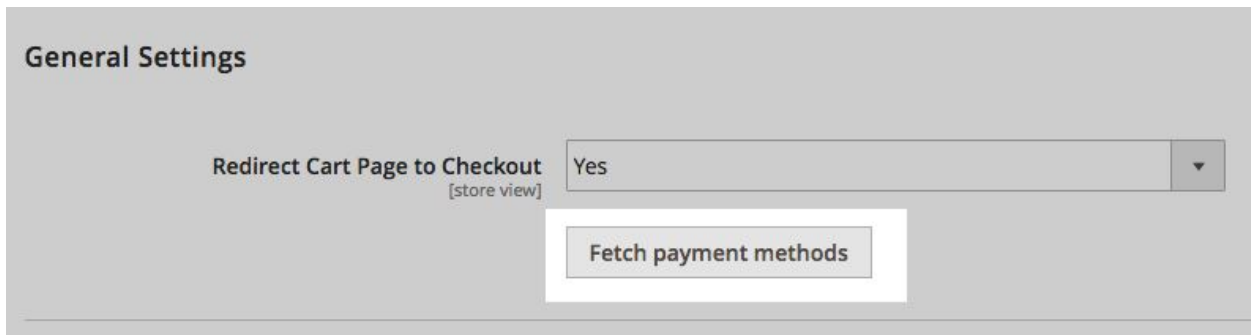
Settings for the module are located under **Stores -> Configuration -> Resursbank -> Checkout**.

You need to specify your **“Username”** and **“Password”** under **API** settings for the module to function at all.

Also, the **“Unit Measure”** setting under **API MUST** have a value or API calls won't work.

The **“Environment”** setting will define what servers you use for your outgoing API calls. **“Test”** should be used in test/development environments, while **"Production"** should only be used on your production environment (live website).

Payment methods



The screenshot shows the 'General Settings' section of the Magento admin interface. It features a configuration option 'Redirect Cart Page to Checkout' with a dropdown menu set to 'Yes'. Below this is a button labeled 'Fetch payment methods'.

This button contacts the Resurs Banks API and fetches all available payment methods. It also creates the necessary files for these to work in Magento. You can find the generated model files in **app/code/local/Resursbank/Checkout/Model/Payment/Fetched/**

NOTE: If (for any reason) any of the generated models should be missing the checkout will still function properly, but the payment method will fall back to "resursbank_default" and simply state "Resurs Bank" on your order view. This can also affect statistics regarding how often specific payment methods are used. You can reinstate missing payment method models by fetching them again from the configuration.

Callback settings

Callback Settings

Salt Storage

[store view]

var

Location of callback salt key. This value should not be changed unless explicitly instructed by Resurs Bank.

Order Status - Confirmed

[store view]

Complete

The status an order will receive after its payment has been unfrozen or if it passes the automatic fraud control.

Order Status - Suspected Fraud

[store view]

Suspected Fraud

The status an order will receive if its payment fails the automatic fraud control.

Callback Registration

Update callbacks

Registered Callbacks

[UNFREEZE]

http://resursbank.dev/rest/V1/resursbank_checkout/order/unfreeze/paymentId/{paymentId}/digest/{digest}

[AUTOMATIC_FRAUD_CONTROL]

http://resursbank.dev/rest/V1/resursbank_checkout/order/automatic_fraud_control/paymentId/{paymentId}/digest/{digest}/result/{result}

[ANNULMENT]

http://resursbank.dev/rest/V1/resursbank_checkout/order/annulment/paymentId/{paymentId}/digest/{digest}

[FINALIZATION]

http://resursbank.dev/rest/V1/resursbank_checkout/order/finalization/paymentId/{paymentId}/digest/{digest}

[BOOKED]

http://resursbank.dev/rest/V1/resursbank_checkout/order/booked/paymentId/{paymentId}/digest/{digest}

[TEST]

http://resursbank.dev/rest/V1/resursbank_checkout/order/test

In the callback settings, to begin with, there is only a button to register callbacks. When clicking it, the callback URLs will register at Resurs Bank. These URLs are based on the base URL for your website.

These URLs are used by Resurs Bank to contact your website and notify it of various events in relation to payments. For example, the module prevents order confirmation emails from being sent when orders are created. They will instead be sent when Resurs Bank has registered payment of the order. This process can take seconds to hours, depending on the user's preconditions.

Some callbacks will modify the order status. These are callbacks from methods which mark the payment as confirmed or being under suspicion of fraud. Which callbacks that trigger these events depend on the selected method of payment. You can configure what order status will be used for these events from the configuration as illustrated by the image above.

NOTE: Since the callback URLs are based on your base URL, you should be careful when registering them from a test/development environment so they do not accidentally overwrite the URLs used by your production environment! You should never use the value "Production" for the "Environment" setting in a test/development environment to avoid such issues.

Salt-key

The salt-key is used to ensure that incoming calls actually come from Resurs Bank.

The path to the directory where the salt-key is stored can be configured from **Callback settings** -> **Salt storage**. By default this file is created as **var/.resursbank_key**. You may wish to change the path if the file containing the key would be insecure due to permissions of that directory in combination with your host configuration, or if it gets deleted as part of your deploy routine.

The salt-key is generated when you register your callbacks, and submitted as part of the callback URLs to Resurs Bank. When incoming calls to the store are made by Resurs Bank, the callback handler (**Model/Callback.php**) will evaluate the provided salt against the content of your local salt-key file to confirm that the calls do not originate from any other source.

If you wish to regenerate your salt-key, or if your salt-key for any reason would be deleted by for example an automated process, you can simply register your callbacks again from the configuration to have it automatically regenerated.

Automatic registration

Automatic callback registration will occur under very specific conditions. It will happen when the configuration is saved, if the following requirements are met:

1. You must have provided an API "Username"
2. You must have provided an API "Password"
3. Your "Environment" must be set to "Production"
4. No callback URLs are currently registered, or your local salt-key file is missing.

Orders

After-shop settings

After-shop Settings



Enabled
[store view]

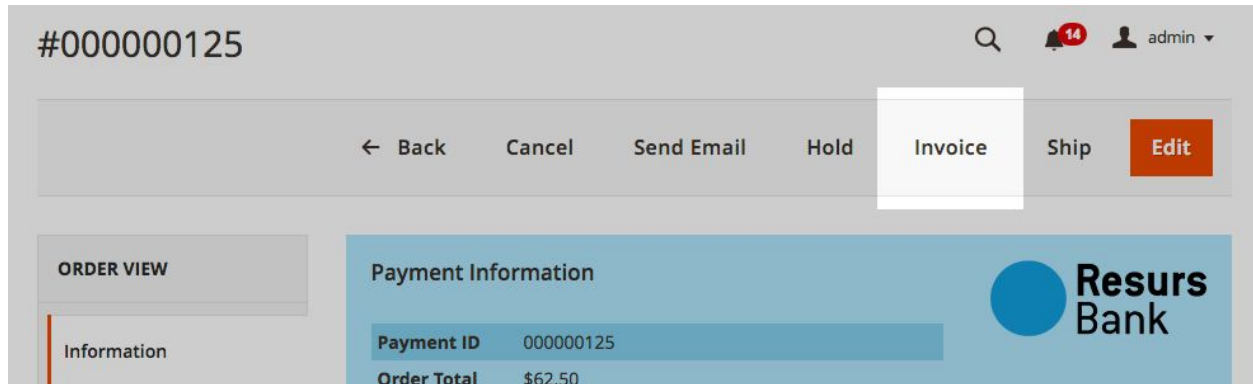
Yes

When this is enabled invoicing (debiting),
crediting and annulment will be synced to
the Resurs Bank payment.

When this feature is enabled you can **charge**, **cancel** and **refund** payments directly from the Magento administration panel.

Charging a payment

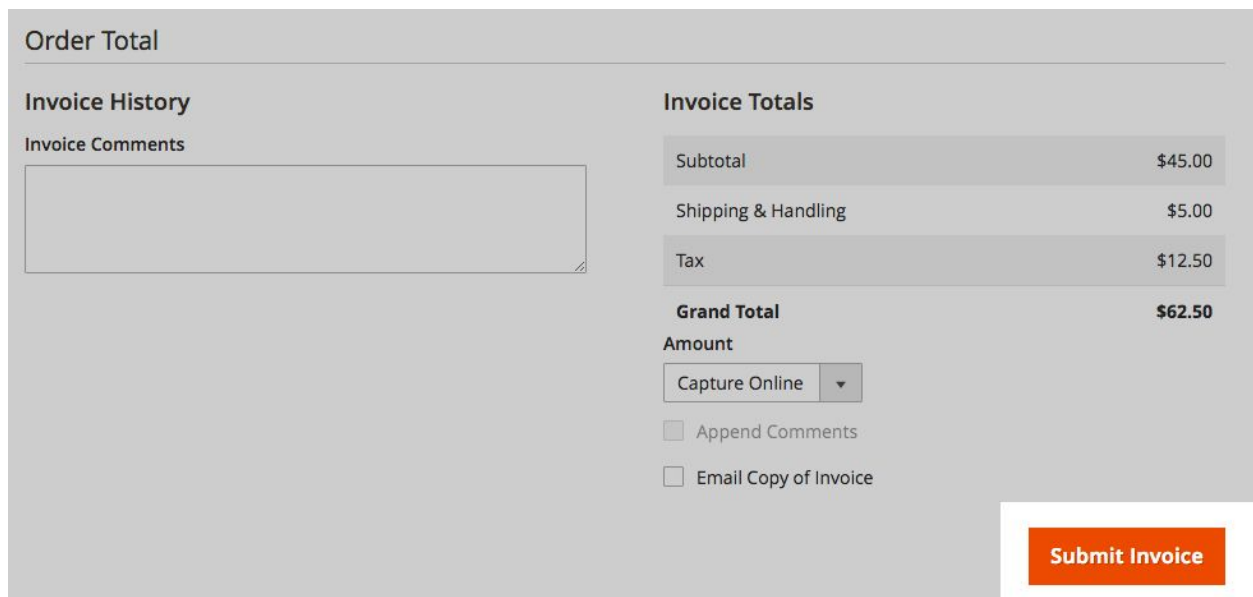
Payments are automatically charged when you create an **invoice** for your order.



The screenshot shows a web interface for order management. At the top left, the order ID is #000000125. On the right, there are icons for search, notifications (14), and a user profile (admin). Below the header, a row of action buttons is displayed: Back, Cancel, Send Email, Hold, Invoice (highlighted with a white box), Ship, and Edit. On the left side, there is a sidebar with 'ORDER VIEW' and 'Information'. The main content area is titled 'Payment Information' and features the Resurs Bank logo. It displays the following details:

Payment ID	000000125
Order Total	\$62.50

On a pending order, press the button in the image above. This will take you to another page.



The screenshot shows the 'Order Total' page. It is divided into two main sections: 'Invoice History' and 'Invoice Totals'. The 'Invoice History' section includes a label 'Invoice Comments' and a large text area for comments. The 'Invoice Totals' section displays a breakdown of costs:

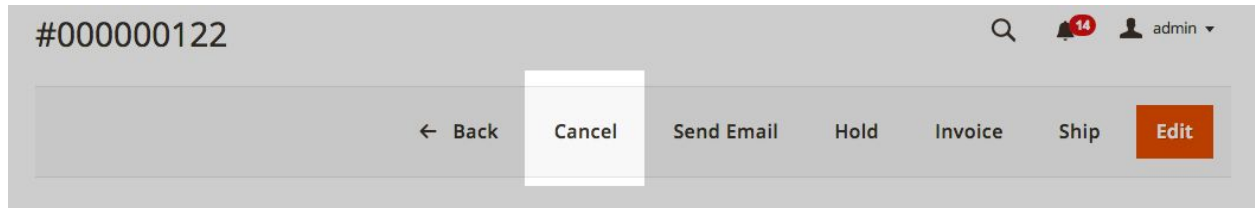
Invoice Totals	
Subtotal	\$45.00
Shipping & Handling	\$5.00
Tax	\$12.50
Grand Total	\$62.50

Below the totals, there is an 'Amount' section with a dropdown menu set to 'Capture Online'. There are also two checkboxes: 'Append Comments' and 'Email Copy of Invoice'. At the bottom right, there is a prominent orange 'Submit Invoice' button.

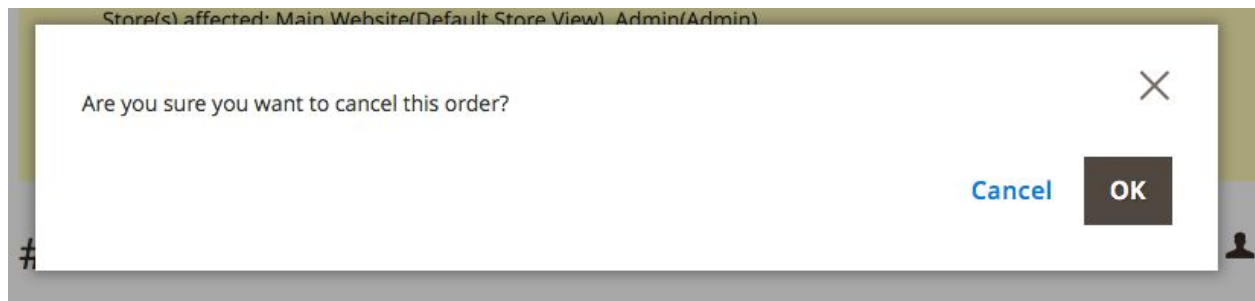
At the bottom is the submit button. Press it, and an invoice will be created. At the same time the payment will be charged by Resurs Bank.

Canceling a payment

Payments are automatically cancelled along with the order.

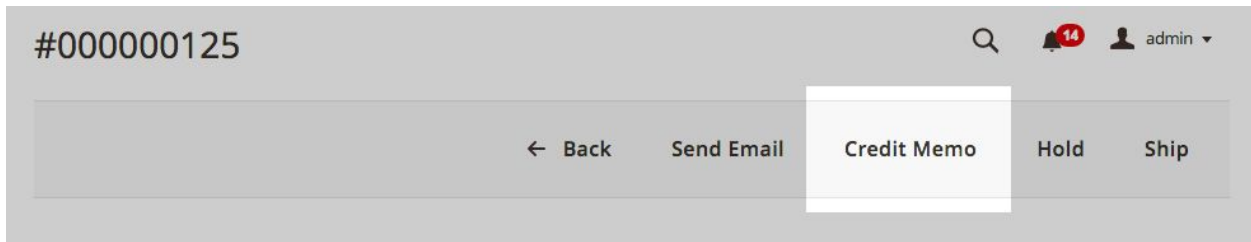


After you click the cancel button, a confirmation window will show up. Just click "OK" and the cancellation process should start.

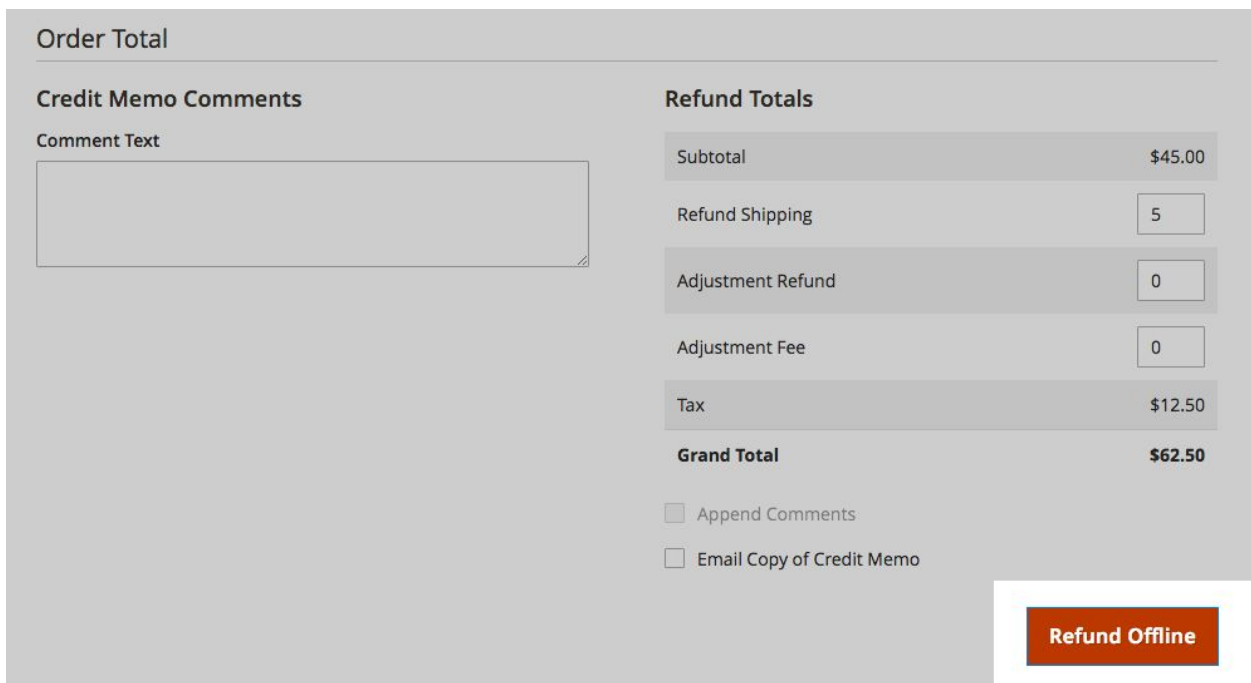


Refunding a payment

Payments are automatically refunded when **credit memos** are issued.



On a completed order, press the button in the image above. This will take you to another page.



Refund Totals	
Subtotal	\$45.00
Refund Shipping	5
Adjustment Refund	0
Adjustment Fee	0
Tax	\$12.50
Grand Total	\$62.50

☐ Append Comments

☐ Email Copy of Credit Memo

Refund Offline

At the bottom is the refund button. Press it, and a credit memo will be created and the payment refunded.

Please note that you can choose to refund specific parts of an order. For more complex cases of partial refunds, for example for orders including percentage based discounts, it may be preferable to handle it directly from Resurs Banks administration panel to ensure the numbers are what you intend.

Order history

You can view the order history by clicking on the tab **Comments history**. **Callbacks** made by Resurs Bank, as well **After-shop** related actions, will append entries to this section.

ORDER VIEW		
Information	Aug 28, 2017, 6:01:57 AM	6:01:57 AM Processing
Invoices	Aug 28, 2017, 6:01:57 AM	6:01:57 AM Pending
Credit Memos	Aug 28, 2017, 6:01:57 AM	6:01:57 AM Invoice #000000003 created
Shipments	Aug 28, 2017, 6:12:33 AM	6:12:33 AM Credit memo #000000002 created
Comments History	Aug 28, 2017, 6:12:50 AM	6:12:50 AM Processing
Transactions	Aug 28, 2017, 6:12:50 AM	6:12:50 AM Processing

Notes for this Order

Captured amount of \$62.50 online Transaction ID: "000000125"

Comment added Aug 28, 2017, 6:01:57 AM 6:01:57 AM

Resursbank: payment debited.

Comment added Aug 28, 2017, 6:01:57 AM 6:01:57 AM

We refunded \$62.50 offline.

Comment added Aug 28, 2017, 6:12:50 AM 6:12:50 AM

Resursbank: payment refunded.

Comment added Aug 28, 2017, 6:12:50 AM 6:12:50 AM

Logs

How to activate

Logs are activated along with the normal system logs in Magento. We log every outgoing and incoming call the module makes/receives. For that reason it's very important that logs are disabled on your production environment (unless required for debug/test purposes), otherwise they will grow very rapidly.

Type of logs

There are 4 kinds of log files where content from various operations will be stored:

- `resursbank_general.log` (for anything without a specified context)
- `resursbank_api.log` (for outgoing API calls to Resurs Bank)
- `resursbank_cache.log` (anything cache related)
- `resursbank_callback.log` (for incoming calls from Resurs Bank)