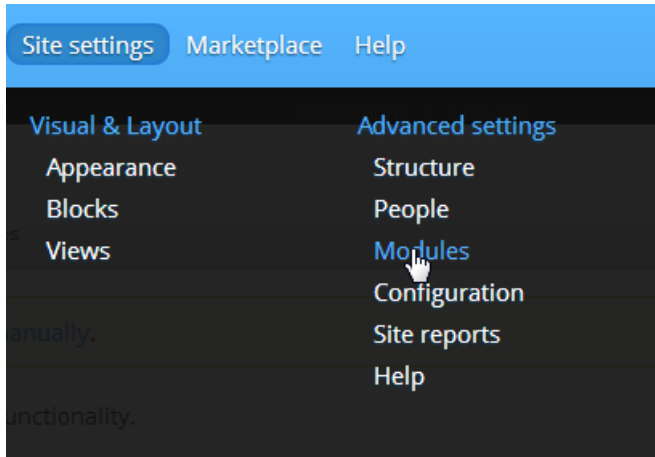
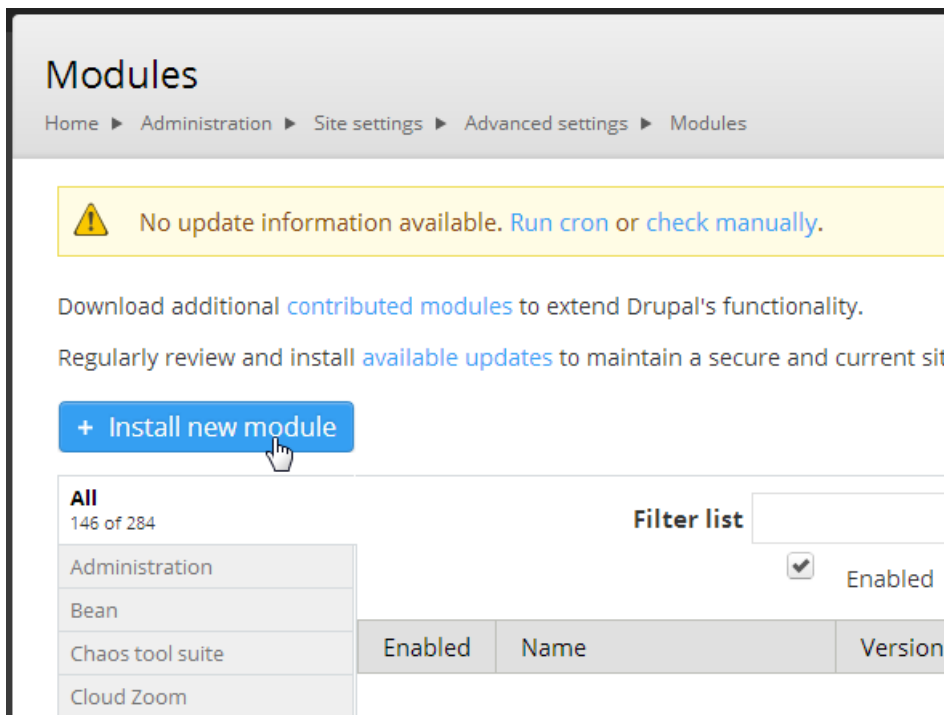


Payson DrupalCommerce.

When you have completed the installation of Drupal commerce and logged in to the admin mode, you can then add our module by clicking on the Site setting tab and select the sub menu Modules.



When you enter this site click on the Install new module button to start the installation of our Payson module.



Upload our module that you saved earlier, then click Install.

Modules

Home Administration Site settings Advanced settings Modules Install new module

You can find [modules](#) and [themes](#) on [drupal.org](#). The following file extensions are supported: *zip tar tgz gz bz2*.

Install from a URL

For example: *http://ftp.drupal.org/files/projects/name.tar.gz*

Or

Upload a module or theme archive to install

Valj fil

Payson_drupalCommerce.zip

For example: *name.tar.gz* from your local computer

Install

When the upload and installation is complete it is time to enable and setup our Payson module. Enter Payson into the search field and our module should appear, click on checkbox as seen in the picture below. Then click on the Save configuration.

Modules

Home Administration Site settings Advanced settings Modules

List Update Uninstall

No update information available. [Run cron](#) or [check manually](#).

Download additional [contributed modules](#) to extend Drupal's functionality.

Regularly review and install [available updates](#) to maintain a secure and current site. Always run the [update script](#) each time a module is updated.

+ Install new module

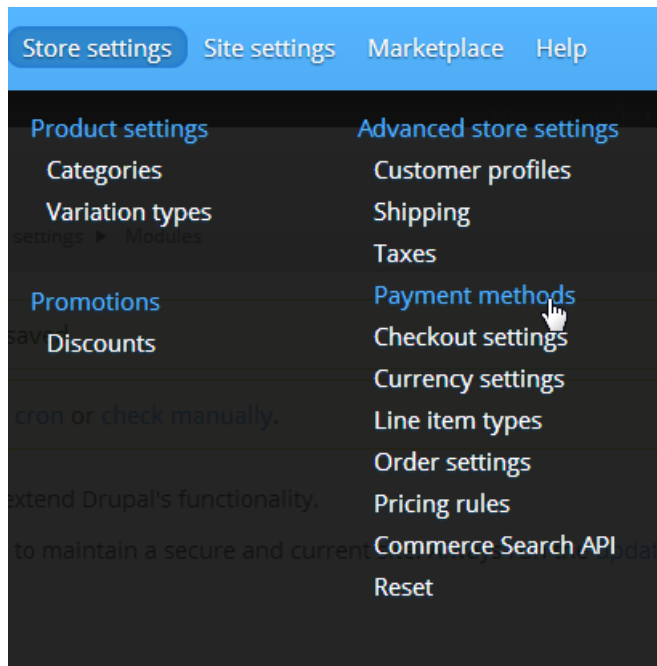
All146 of 284

Filter listPayson

☒ Enabled☒ Disabled☒ Required☒ Unavailable

Enabled	Name	Version	Description	Operations
<input checked="" type="checkbox"/>	Payson DrupalCommerce		Implements Swedish Payson (www.payson.se) payment API services for use with Drupal Commerce. Requires: Commerce (enabled), System (enabled), Entity API (enabled), Rules (enabled), Entity tokens (enabled), Commerce UI (enabled), Payment (enabled), Order (enabled), Customer (enabled), Address Field (enabled), Chaos tools (enabled), Line Item (enabled), Price (enabled)	

It is now time to configure our Payson module. Click on the tab Store settings and select the sub menu Payment settings.



When you enter this site you should find our module under the headline Disabled payment method rules, as seen in the picture below. Enable this module and then click on edit.

Payment methods

Home ► Administration ► Store settings ► Advanced store settings ► Payment methods

Disabled *reaction rule* Payson API (Visa, Mastercard, Internet Banking).

Payment methods are enabled for use by the rule configurations listed below. An enabled payment rule can specify a payment method to enable using one of the available *Enable payment method* actions. The action's settings form will contain any necessary settings for the payment method that must be configured before it may be used.

[+ Add a payment method rule](#)

DO YOU NEED HELP WITH INTERNATIONAL PAYMENTS?
International payment providers can be complicated. Check out our [webinar](#) to learn more about this complex environment.

Enabled payment method rules

Name	Event	Status	Operations
There are no active payment methods.			

Disabled payment method rules

Name	Event	Status	Operations
American Express Machine name: commerce_payment_amex_hosted, Weight: 0 Tags: Commerce Payment <ul style="list-style-type: none">Available on the checkout formAvailable on the order payment terminal	Select available payment methods for an order	Default	edit enable clone export
Example payment Machine name: commerce_payment_commerce_payment_example, Weight: 0 Tags: Commerce Payment <ul style="list-style-type: none">Available on the checkout formAvailable on the order payment terminal	Select available payment methods for an order	Overridden	edit enable clone revert export
Payson API (Visa, Mastercard, Internet Banking) Machine name: commerce_payment_payson_api_pay, Weight: 0 Tags: Commerce Payment <ul style="list-style-type: none">Available on the checkout formNot available on the order payment terminal	Select available payment methods for an order	Overridden	edit enable clone revert export

Under the headline Actions you will now see our module click on edit.

Actions

Show row weights

Elements	Operations
<div><div><div><div><div></div><div></div></div><div>Enable payment method: Payson API (Visa, Mastercard, Internet Banking)</div><div>Parameter: Order:[commerce-order]</div></div></div></div>	<div><div>edit</div><div>delete</div></div>
<div><div>+ Add action</div><div>+ Add loop</div></div>	

It is now time to fill in the credential that you have been provided by Payson. We have filled in the fields below, with Payson test-credentials. When done click Save.

Payment settings

Primary Payson e-mail address *

testagent-1@payson.se

The e-mail address you use for the Payson account you want to receive payments.

First Name *

Payson

Receiver First Name

Last Name *

Tester

Receiver Last Name

Agent-ID *

4

Your Payson Agent-ID.

MD5-Key *

2acab30d-fe50-426f-90d7-8c60a7eb31d4

Your Payson MD5-key.

Currency code

SEK

Transactions can only be processed in one of the listed currencies.

Payson guarantee

☒ Do not offer Payson guarantee

☐ Offer Payson guarantee

☐ Optional Payson guarantee

[Read more about Payson Guarantee here](#)

Payson transaction url

☒ Test API - Used for testing purposes

☐ Merchant - Used for live transactions

Payson Test e-mail address

testagent-1@payson.se

The e-mail address you use for the Payson account you want to receive payments.

Test Agent-ID

4

Test Payson Agent-ID. Default 3.

Test MD5-Key

2acab30d-fe50-426f-90d7-8c60a7eb31d4

Test Payson MD5-key. Default: 54e5eb76-3888-4673-a5ba-be2f4187c3d5

Instant Payment Notification (IPN) logging

☒ Log notifications during IPN validation and processing.

☐ Log notifications with the full IPN during validation and processing (used for debugging).

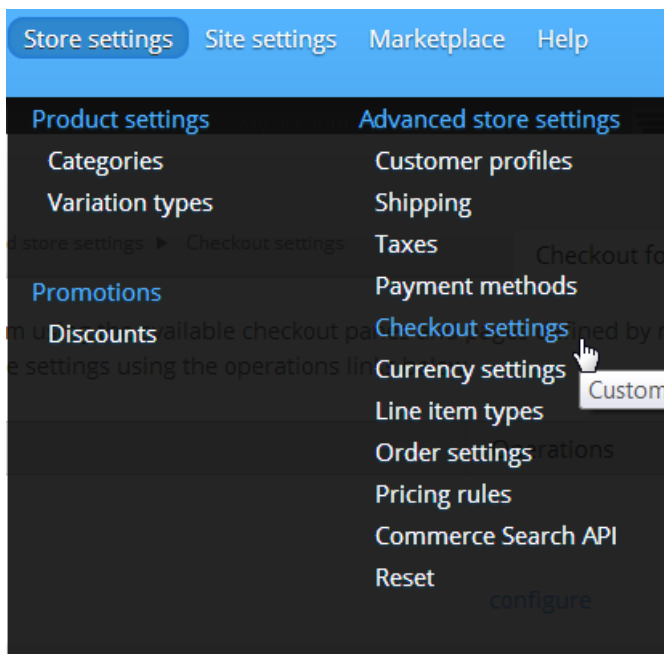
Save

The basic setup is now done, and you can start testing. But in order to receive an accurate order status after each order you must complete following steps listed below.

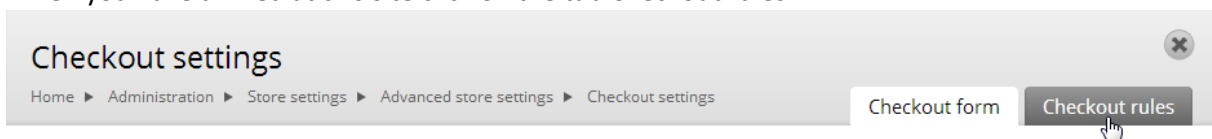
In the picture below you will see a few orders that has been placed and which has returned with different payment status on the order. In order to get accurate payment status on your orders you will have to setup some local settings.

<input type="checkbox"/>	ID	Created	Name/Email	Total	Status	Operations
<input type="checkbox"/>	31	07/07/2014 - 11:30	Bo Anderssson karlbrundin@hotmail.com	231,00 kr	Completed	Quick Edit
<input type="checkbox"/>	30	07/07/2014 - 11:29	Bo Anderssson karlbrundin@hotmail.com	879,00 kr	Canceled	Quick Edit
<input type="checkbox"/>	29	07/04/2014 - 10:33	Bo Anderssson karlbrundin@hotmail.com	216,00 kr		Quick Edit

To begin we will enter the Store settings tab and select Checkout settings.



When you have arrived at this site click on the tab Checkout rules.



When you have arrived at Checkout rules you should see a list of different rules among them you will find a rule called "Update the order status on the checkout completion". When found click on edit.

Update the order status on checkout completion	Completing the checkout process	Custom	edit	disable	clone	delete	export
<small>Machine name: rules_update_the_order_status_on_checkout_completion, Weight: 0, Tags: Commerce Order</small>							

If you do not find this rule, you can add it yourself by clicking on the button Add a checkout rule.

Checkout settings

Home ▶ Administration ▶ Store settings ▶ Advanced store settings ▶ Checkout settings ▶ Checkout rules

Checkout formCheckout rules

When a customer advances to the checkout completion page, rules reacting on the *Completing the checkout process* are evaluated. Default rules handle standard tasks like updating the order status, sending order e-mails, and creating accounts for anonymous users. You can edit these or add additional rules to customize your checkout workflow.

+ Add a checkout rule

Then add the following data.

Checkout settings

Home ▶ Administration ▶ Store settings ▶ Advanced store settings ▶ Checkout settings ▶ Checkout rules ▶ Add a checkout rule

Name *

Update the order status on checkout completion

Machine name:

Tags

Commerce Order

Tags associated with this configuration, used for filtering in the admin interface. Separate multiple tags with commas.

Save

Cancel

When editing the rule “Update the order status on the checkout completion”, click on the button Add event.

Events

Event	Operations
Completing the checkout process	delete
+ Add event	

You will then see a drop down menu, here you will select the option “When order is first paid in full” which you will find under the headline “Commerce Payment”. Then click save.

- Select -

- After deleting a commerce discount offer
- After saving a new commerce discount offer
- After updating an existing commerce discount offer
- Before saving a commerce discount offer
- Commerce Line item**
 - After deleting a commerce line item
 - After saving a new commerce line item
 - After updating an existing commerce line item
 - Before saving a commerce line item
 - Commerce Line item is viewed
- Commerce Order**
 - After deleting a commerce order
 - After saving a new commerce order
 - After updating an existing commerce order
 - Before saving a commerce order
 - Commerce Order is viewed
- Commerce Payment**
 - Select available payment methods for an order
 - When an order is first paid in full**
- Commerce Payment transaction**

Next it is time to add an Action to this rule.

Actions

Elements	Weight	Operations
None		
+ Add action + Add loop		

This action will be selected again with a drop down menu. Under the headline “Commerce Order” you will find “Update the order state”

Select the **action** to add

Add a variable

- Multiply the unit price by some amount
- Set the unit price to a specific amount
- Set the unit price's currency code
- Subtract an amount from the unit price
- Commerce Order**
 - Update the order state**
 - Update the order status

You should now see the following options, you will now select what status an order will have when a payment has been paid, we would suggest Completed. You should be able to select this under “Order State” the with the dropdown menu “Completed”. Then click save.

Order to update
Data selector *

commerce_order

The data selector helps you drill down into the data available to Rules. To make entity fields appear in the data selector, you may have to use the condition 'entity has field' (or 'content is of type'). More useful tips about data selection is available in [the online documentation](#).
Data types: Select data of the type *Commerce Order*.

▸ Data selectors

Switch to the direct input mode

Order state
Select the order state whose default status the order will be updated to.
Value *

- Select -

- Select -
Canceled
Shopping cart
Checkout
Pending
Completed

Save

The checkout rule should now have the following appearance.

Events

Event	Operations
Completing the checkout process	<div>delete</div>
When an order is first paid in full	<div>delete</div>
<div>+ Add event</div>	

Conditions

Elements	Weight	Operations
None		
<div>+ Add condition</div> <div>+ Add or</div> <div>+ Add and</div>		

Actions

[Show row weights](#)

Elements	Operations
<div>+ Update the order state</div> <div>Parameter: Order to update:[commerce_order], Order state:Completed</div>	<div>edit</div> <div>delete</div>
<div>+ Add action</div> <div>+ Add loop</div>	

To get status Canceled when a payment is denied. Then we can use the rule we just modified/created and click clone and rename the rule to something appropriate.

[Update the order status on checkout completion](#)
Machine name: rules_update_the_order_status_on_checkout_completion,
Weight: 0
Tags: Commerce Order

Completing the checkout process,
When an order is first paid in full

Custom

edit


disable

clone

delete

export

When you enter the edit mode for the new rule, you should then remove the event we created in the previous rule ("When an order is first paid in full"). You should also edit the Action "Update the order state" to preferred order status we will choose "Canceled", but "Pending" can also be appropriate. Below you can see how the new rule should appear.

Editing reaction rule "Update the order status on checkout Failure" 

Home ► Administration ► Store settings ► Advanced store settings ► Checkout settings ► Checkout rules ► Editing reaction rule "Update the order status on checkout Failure"

Events

Event	Operations
Completing the checkout process	delete
+ Add event	

Conditions

Elements	Weight	Operations
None		
+ Add condition + Add or + Add and		

Actions [Show row weights](#)

Elements	Operations
+ Update the order state Parameter: Order to update:[commerce_order], Order state:Canceled	edit delete

This rule will enact when a payment response from us is a denied payment or if it do not get any response from our payment solution. Now we have completed all the settings for the order statuses to work properly.

Good luck with your new web store.

Best regards

Payson