



Local Shop : General configuration

Introduction

The shop must be generally configured with seller information in order to operate.

Path: Site administration/plugins/local plugins/shop

The configuration involves four set sections and a certain amount of variable sections depending on the number of payment plug-ins installed.

Global settings

- **Default front office navigation:** This setting indicates the sequencing order of front office screens. The shop stores all pre-purchase data during a session. This data can be in a different order according to the business or what the operator wants.



Be careful: navigation sequence implicates all front office screens. Some combinations can generate shop malfunctions, such as, for example, attempting to display a bill before a payment. There are consequently certain rules that need to be followed, notably:

- Differentiate between “pre-payment” screens and “after payment” screens
- Provide the bill after payment

Pre-payment steps:

- **shop** : establish the cart
- **purchaserequ** : collect product settings
- **users** : enter information on seats
- **customer** : collect customer information / existing customer login
- **order** : order summary

Payment steps (or steps after payment):

- **payment** : payment or redirect to a payment interface
- **produce** : digital production of purchased products
- **invoice** : presentation of bill or pro forma bill
- **Default currency** : Selects the currency given for the shop.



Note : The multi-currency shop operation is planned, but still requires some finalisation. (Use of several currencies in the same catalogue instance.)

- **Test mode** : If this mode is enabled, then all payment interfaces are enabled based on the sandbox parameters of the different payment partners. Furthermore, persons who are not logged in or who are not administrators are prohibited from payment phase use. Transactions conducted in test mode however will generate actual records of invoices and products. You might have to reset transactions before live implementation of the shop.
- **Test lock override** : If this mode is enabled, then the test mode purchase lock is lifted to complete a transaction.
- **Max items** : Determine the max. size of list displays in the back office.
- **Hide products if unavailable** : Défaut : No. If enabled, unavailable products (for availability check reasons, but yet published for sale) will not be shown at all on front. If disabled, they still appear just shaded and without the order button.

The screenshot shows the 'Global settings' configuration page. It includes several settings:

- Default front office navigation**: A text input field containing 'shop.purchaserequ.users.customer.order.payment.produce.invoice'. Below it, a note states: 'Default: shop.purchaserequ.users.customer.order.payment.produce.invoice. Some office owners may reorder the steps in the front process. Keep in mind some orders might NOT work correctly, i.e. trying to get payment before thoc cart and customer are defined.'
- Default Currency**: A dropdown menu set to 'Euro'. Below it, it says 'Default: Euro' and 'Currency suffix:'.
- Test mode**: A checked checkbox. Below it, it says 'Default: No' and 'Enables the mode test for the payments'.
- Test lock override**: An unchecked checkbox. Below it, it says 'Default: No' and 'Overrides test lock on purchase to test simulated external users'.
- Max items**: A text input field. Below it, it says 'Default: Empty' and 'Max items per page'.
- Hide products if not available**: An unchecked checkbox. Below it, it says 'Default: No' and 'If enabled, non available products (for availability check reasons, but yet published for sale) will not be shown at all on front. If disabled, they still appear but shadowed and without the ordering button.'

Discount settings

The Moodle shop handles a three-tiered discount system. A discount can only be attributed by logged in users based on certain capabilities at system level.

The discount is applied to the order total and is triggered by a sales threshold. A customer can benefit from a more or less significant discount according to the level of discount they earn by right, but the trigger threshold remains unique. (You cannot tier up different automatic discounts according to cart amount).

To modulate prices according to local quantities with a product, view the settings for a single product. Only one discount threshold can be applied at a time (discounts are therefore not cumulative).

- **Discount threshold**: A threshold that triggers the effectivity of the discount. Users must have **local/shop:discountagreed** capability to benefit from it.
- **Discount rate**: Is expressed as a % of total order amount. A discount rate applied on the overall amount. May be applied unconditionally if customer has the local/shop:discountagreed capability on.
- **Second discount rate**: This rate is applied to logged in users : A discount rate applied when customer has the local/shop:seconddiscountagreed capability on You can, for example, create a

site role with this capability and have this role attributed via a badge to identify preferred customers.

- **Third discount rate:** Likewise, a third discount rate can be triggered for. ***local/shop:thirddiscountagreed*** capability.

Discount settings

Discount Threshold <small>local_shop discountthreshold</small>	0	Default: 0 A threshold that triggers the effectivity of the discount
Discount Rate <small>local_shop discountrate</small>	0	Default: 0 A discount rate applied on the overall amount. May be applied inconditionally if customer has the local/shop:discountagreed capability on
Discount Rate 2 <small>local_shop discountrate2</small>	0	Default: 0 A discount rate applied when customer has the local/shop:seconddiscountagreed capability on
Discount Rate 3 <small>local_shop discountrate3</small>	0	Default: 0 A discount rate applied when customer has the local/shop:thirddiscountagreed capability on

Information about seller's identity

- Seller name: Seller name displayed on screens.
- Seller address : physical adress.
- Seller postal/zip code: Zip code displayed for the seller identity.
- Seller city/town : City of the merchant.
- Seller country: Country displayed.
- Seller email: Direct email.
- Mailing adress support: Mail address of the customer support team.
- Phone support number: Phone number of the customer support team.

Lastly, accounting information is used to display any different data on documents produced (purchase orders, pro forma bills, bills, etc.)

- Seller Registrar ID: The official company registrar ID of the seller.
- Seller Billing address: Address of seller's accountant.
- Seller Billing Zip Code: Zip code of accountant.
- Seller Billing city/town : City of accountant.
- Seller Billing country: Country of accountant.
- Seller Logo : the logo that will be printed on bills and orders.

Information about vendor's identity

Seller name <small>local_shop sellername</small>	<input type="text" value="ActiveProLearn"/> Default: Empty The official name of the vendor
Seller Address (physical) <small>local_shop selleraddress</small>	<input type="text" value="Rond Point Benjamin Franklin"/> Default: Empty The address (postal) of the merchant
Seller Zip Code <small>local_shop sellerzip</small>	<input type="text" value="34000"/> Default: Empty Zip code for the seller identity
Seller City <small>local_shop sellercity</small>	<input type="text" value="Montpellier"/> Default: Empty City of the merchant
Seller Country <small>local_shop sellercountry</small>	<input type="text" value="France"/> Default: Empty Country of the merchant
Seller Email <small>local_shop sellermail</small>	<input type="text" value="ventes@activeprolearn.com"/> Default: Empty Mail address of the merchant
Mail support address <small>local_shop sellermailsupport</small>	<input type="text" value="support@activeprolearn.com"/> Default: Empty Mail address of the sales customer support team
Phone support number <small>local_shop sellerphonesupport</small>	<input type="text"/> Default: Empty Phone number of the sales customer support team
Seller Registrar ID <small>local_shop sellersiret</small>	<input type="text" value="Siret 801 675 299 00011"/> Default: Empty The official company registrar ID of the seller
Seller Billing Address <small>local_shop sellerbillingaddress</small>	<input type="text" value="Rond Point Benjamin Franklin"/> Default: Empty The accountance location of the merchant
Seller Billing Zip Code <small>local_shop sellerbillingzip</small>	<input type="text" value="34000"/> Default: Empty Zip code of the accountant service
Seller Billing City <small>local_shop sellerbillingcity</small>	<input type="text" value="MONTPELLIER"/> Default: Empty City of accountance service
Seller Billing Country <small>local_shop sellerbillingcountry</small>	<input type="text" value="FRANCE"/> Default: Empty Country of accountant service
Seller logo <small>local_shop sellerlogo</small>	<div style="text-align: right; font-size: small;">Maximum size for new files: Unlimited, maximum attachments: 1</div> <div style="border: 1px solid #ccc; padding: 5px;"><div style="display: flex; justify-content: space-between; align-items: center;">Files📁 📄 🗑️</div><div style="text-align: center; margin: 10px 0;"> logo-APL-gd.png</div></div> <p>Default: Empty the logo that will be printed on invoices and orders</p>

Settings related to payment plug-ins

Each installed payment plug-in can add a set of settings dedicated to this plug-in. Here are some

examples.

Mercanet configuration option

The Mercanet interface is the interface used by BNP and banks in that same group.

The screenshot shows a configuration form titled "Mercanet configuration options". It contains several fields:

- Mercanet Merchant ID:** A text input field. Below it, the text reads "local_shop | mercanet_merchant_id", "Default: Empty", and "Merchant ID. Provided by Mercanet."
- Country code:** A dropdown menu currently showing "France". Below it, the text reads "local_shop | mercanet_country" and "Mercanet country".
- Generate the Pathfile file:** A section header.
- Currency code:** A dropdown menu currently showing "Euros". Below it, the text reads "local_shop | mercanet_currency_code" and "Currency code".
- Processor type:** A dropdown menu currently showing "32 bits". Below it, the text reads "local_shop | mercanet_processor_type" and "Determines API Processor type to be used (Linux versions)".

Ogone configuration option

Ogone (now Ingénico), is a multi-bank payment intermediary that can centralise payments received via a unique interface for several different banking systems.

The screenshot shows a configuration form titled "Ogone/Ingenico configuration options". It contains several fields:

- Ogone/Ingenico Merchant Identifier:** A text input field. Below it, the text reads "local_shop | ogone_pid", "Default: Empty", and "Ogone/Ingenico Merchant Identifier".
- Secret glue for gateway input:** A text input field. Below it, the text reads "local_shop | ogone_secret_in", "Default: Empty", and "This glue is used to secure the transfer of your merchant data to the ogone gateway. It needs to be identical to the secret you have setup in the ogone account backoffice."
- Secret glue for gateway output:** A text input field. Below it, the text reads "local_shop | ogone_secret_out", "Default: Empty", and "This glue is used to check the answers of the ogone gateway. It needs to be identical to the secret you have setup in the ogone account backoffice."
- Multiple Shop diverter info:** A text input field. Below it, the text reads "local_shop | ogone_paramvar", "Default: Empty", and "This is to be used in case you have multiple shops hosted on the same facility (See Ingenico/Ogone Integration Documentation)".
- Logo URL:** A text input field. Below it, the text reads "local_shop | ogone_logourl", "Default: Empty", and "An URL to your logo for the remote payment page. (must be an HTTPS url as ogone payment service is secured)".

Paypal

When using test mode, you will send all purchases to the Paypal sandbox. Paypal sandbox do NOT

automate the IPN process (asynchronous return to merchant site) and you will have to simulate it using the Paypal IPN Simulator. The IPN end point of this site is : http://ma.formation-enligne.com/local/shop/paymodes/paypal/paypal_ipn.php.

Paypal configuration options

When using test mode, you will send all purchases to the Paypal sandbox. Paypal sandbox do NOT automate the IPN process (asynchronous return to merchant site) and you will have to simulate it using the Paypal IPN Simulator. The IPN end point of this site is : http://ma.formation-enligne.com/local/shop/paymodes/paypal/paypal_ipn.php.

Paypal seller test account (paypal sandbox) <small>local_shop paypal_seller_test_account</small>	<input type="text" value="vent@activeprolearn.com"/>
	Default: Empty The account id of the reseller
Sales Service (test mode) <small>local_shop paypal_seller_test_service</small>	<input type="text" value="Sandbox Paypal APL"/>
	Default: Empty An arbitrary label that allows sorting transactions against multiple selling services
Paypal seller account <small>local_shop paypal_seller_account</small>	<input type="text" value="vent@activeprolearn.com"/>
	Default: Empty The account id of the reseller
Sales Service <small>local_shop paypal_seller_service</small>	<input type="text" value="Ventes Paypal APL"/>
	Default: Empty An arbitrary label that allows sorting transactions against multiple selling services

System Pay PLUS

System Pay PLUS, developed by ATOS, is the payment system used by Caisse d'Epargne, the Banques Populaires, and banks in that group.

System Pay PLUS Settings

Service URL <small>local_shop systempay_service_url</small>	<input type="text" value="https://payzen.societegenerale.eu"/>
	Default: Empty Url for Payment Transaction Server
Merchant ID <small>local_shop systempay_merchant_id</small>	<input type="text" value="9522"/>
	Default: Empty This is the ID of the merchant service provided in your backoffice interface
Test certificate <small>local_shop systempay_test_certificate</small>	<input type="text" value="32411"/>
	Default: Empty The certificate key to encode test submissions
Prod certificate <small>local_shop systempay_prod_certificate</small>	<input type="text" value="79110999506"/>
	Default: Empty The certificate key to encode production submissions
Use 3D Secure service <small>local_shop systempay_use_3dsecure</small>	<input checked="" type="checkbox"/> Default: No 3D Secure is an option to the merchant payment contract that will callback the customer for a confirmation code.
Bank brand <small>local_shop systempay_bank</small>	<input type="text" value="Payzen / Société Générale"/> Default: Caisse d'Epargne
Country <small>local_shop systempay_country</small>	<input type="text" value="France"/> Country
Currency Code <small>local_shop systempay_currency_code</small>	<input type="text" value="Euro"/> Currency Code
Use local time <small>local_shop systempay_use_localtime</small>	<input checked="" type="checkbox"/> Default: Yes Use local time for transactions, or UTC/GMT time if not checked.

Notifications

- **Pre sales notifications** : Enable or disable the order input notification.

Notifications

Pre sales notifications

local_shop | presalenotification Default: Yes

Enable or disable the order input notification

Reseller banking information

Introduction

The vendor's banking information is used for offline payment procedures to provide bank details for interbank transfers or for issuing cheques. This information is found in your bank documentation.

- Bank name :Bank company name
- Bank code :Bank code of the merchant
- Bank office: Agency code
- Bank Account : Agency code
- Bank account key :Account key of the merchant
- IBAN : The IBAN number of the merchant
- BIC : BIC identification code of the merchant's bank
- VAT European Intracommunautary:European VAT number

Reseller banking information

Banking
local_shop | banking

SOCIE [REDACTED] RALE

Default: Empty
Bank company name

Bank code
local_shop | bankcode

300 [REDACTED]

Default: Empty
Bank code of the merchant

Bank Office
local_shop | bankoffice

01 [REDACTED]

Default: Empty
Agency code

Bank Account
local_shop | bankaccount

00027004001

Default: Empty
Agency code

Bank Account Key
local_shop | bankaccountkey

9 [REDACTED]

Default: Empty
Account key of the merchant

IBAN
local_shop | iban

FR [REDACTED] 195

Default: Empty
The IBAN number of the merchant

BIC
local_shop | bic

SO [REDACTED] PP

Default: Empty
BIC identification code of the merchant's bank

VAT European Intracommunautary
local_shop | tvaeurope

Default: Empty
European VAT number

Apparence

Introduction

Cette section permet de gérer l'apparence des produits en front-office

Appearance

Thumb height <small>local_shop productimageheight</small>	<input type="text" value="150"/>
	Default: 150 Product thumb height (px)
Thumb width <small>local_shop productimagewidth</small>	<input type="text" value="150"/>
	Default: 150 Product thumb width (px)
Thumb right margin <small>local_shop productimagemargin</small>	<input type="text" value="40"/>
	Default: 40 Product thumb right margin (px)
Element thumb height <small>local_shop elementimageheight</small>	<input type="text" value="80"/>
	Default: 80 Product subelement thumb height (px)
Element thumb width <small>local_shop elementimagewidth</small>	<input type="text" value="80"/>
	Default: 80 Product subelement thumb width (px)
Element thumb right margin <small>local_shop elementimagemargin</small>	<input type="text" value="25"/>
	Default: 25 Product subelement thumb right margin (px)

Experimental

These features are enabled but still subject to change.

- **Use shipping** : Default: No: If enabled, delivery charges are handled.
- **Use sales delegation**: Default: No: If enabled, users can own products and sell them on their own behalf : The Moodle shop is also designed to be a marketplace for course authors. This feature will only be available in the extended version of the shop (PRO). The sales delegation provides that the catalogue products can belong to delegate seller accounts (generally associated with the course author) so that the platform can manage sales on behalf of the author. Representatives will have access to a portion of the back office in order to monitor sales management.
- **Use master/slave catalogues**: Default: No. If enabled, master/slave catalogues are handled.
- **Use renewable products**: Default: No: If enabled, renewable products are handled.

Experimental features

- Uses shipping
local_shop | useinshipping
Default: No
Enables the shipping costs engine
- Use sales delegation
local_shop | usedelegation
Default: No
If enabled, some users may get ownership of products and sales in the shop.
- Uses master/slave catalogs
local_shop | useslavecatalogs
Default: No
Uses master/slave catalogs
- Uses renewable products
local_shop | userenewableproducts
Default: No
Uses renewable products

Plugin distribution

- Emulate community version
report_trainingissions | emulatecommunity
Default: No
If enabled, the plugin will behave as the public community version. This might loose features !

Credits

- Valéry Frémaux (valery@activeprolearn.com) Conception - Developments - Documentation
- Florence Labord (florence@activeprolearn.com) Fonctionnal review - Tests - Documentation

[back to local shop composant](#) - [Back to catalogue](#)

From:
<https://docsen.activeprolearn.com/> - **Documentation Moodle ActiveProLearn**

Permanent link:
<https://docsen.activeprolearn.com/doku.php?id=local:shop:globalconfig&rev=1526056439>

Last update: **2026/01/13 07:58**

