Valitor WooCommerce Module

Integrating with Valitor could not be easier. Choose between Hosted, HTTP POST or XML integration options, or alternatively browse our selection of client libraries and e-commerce platform modules. Imagine having full creative control over payment flow, checkout design, and A/B testing without any card data risk.

Table of Contents

Prerequisites

We highly recommend gathering all the below information before starting the installation:

- installing the plug-in: "AltaPay for WooCommerce" payment plug-in needs to be installed on the merchant WooCommerce/WordPress website;
- configuration of AltaPay bundle: AltaPay will provide the credentials for the payment gateway, terminals and custom gateway (i.e. yourShopName.altapaysecure.com);
- the minimum requirements are:
    - WordPress min. 4.5.3 – max. 4.9.4
    - WooCommerce 2.6.2+; for the plugin version up to 1.6.0, included
    - WooCommerce min. 3.0.0 – max. 3.3.3 for the plugin version 1.6.1+
    - PHP 5.5+;
    - PHP-bcmath library installed;
    - PHP-curl MUST be enabled.
- the latest tested versions are:
    - Wordpress 4.5.3, WooCommerce 2.6.2 and PHP 7.0.4;
    - Wordpress 4.5.4, WooCommerce 2.6.3 and PHP 7.0.8;
    - Wordpress 4.7.8, WooCommerce 3.2.5 and PHP 5.6.32.
    - Wordpress 4.9.9, WooCommerce 3.2.5 and PHP 7.0.32

Installation

Installing this plug-in will enable the web shop to handle card transactions through AltaPay's gateway.
There are two ways of installing the plug-in: by uploading the build package or by search for the plug-in on the WC Marketplace.

1. By uploading the build package:

From the admin panel:

- navigate to: Plugins → Add new;
- select Upload Plugin;
- select the build package (zip file) that contains the plug-in;
- once the build package has been selected click on Install Now.

2. By searching for the plug-in in the WC Marketplace:

- navigate to Plugins → Add new;
- in the search text box (Search plugins...) type in the keyword "altapay" and wait for the plug-in to be found.

Once the plug-in has been set to be installed a successful message should be seen (Plugin installed successfully). By clicking on Activate Plugin button the message "Plugin activated should be displayed".

Configuration

In this step the AltaPay plug-in can be configured in order to fit merchant's needs (adding payment methods and configuring payments).
The standard configuration connects the plug-in with the test gateway. Follow the steps below to connect the plug-in with your custom payment gateway.

Bound the payment plug-in with the custom payment gateway

In order to connect the plug-in with the custom payment gateway, the next steps are required:

- navigate to: Plugins → Installed Plugins;
- click on Settings button from the "Altapay for WooCommerce" section;
- fill in the Gateway URL, Username and Password fields with the information provided by AltaPay;
- click on Create Page button (very important);
- click on Save Changes button (from the bottom of the page).

To select the Terminals, follow the steps below:

- click on Refresh Connection, where a list with one or more payment terminals will be shown;
- select the payment terminals which should be shown on the checkout page by checking the checkbox related to each one;
- click on Save Changes button.

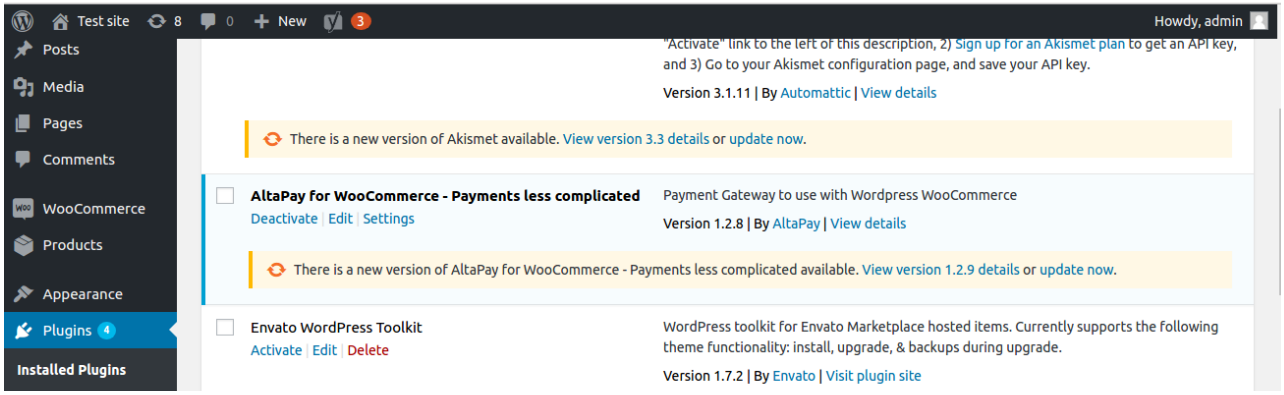
Configure the terminals for the checkout page

For the AltaPay payment method to appear in the checkout page, follow the steps below:

- navigate to: WooCommerce → Settings → Checkout;
- for each AltaPay payment terminal, there must be a link at the top of the page. Refresh the page if you do not see any links, or if the links do not seem updated;
- click on the desired payment terminal;
- enable the payment terminal;
- fill in the text field "Title"; this field will appear in the checkout page as the title of the payment terminal;
- fill in the field "Message"; this field must contain the message that will appear for the user in the checkout page and describes the payment terminal;
- check if the selected currency matches WooCommerce currency (WooCommerce currency can be viewed in WooCommerce → Settings → General);
- save the changes.

Update the plug-in

- navigate to: Plugins → Installed Plugins;
- if there is a new version of the plug-in, you should see a warning message, as is shown in the image below;



- the plug-in can be updated by clicking on the "Update now" link.

	If you customized the plugin source code, the changes may be overwritten in the update process.

Customize the checkout page

The checkout page is implemented by the following file:
<wordpress>/wp-content/plugins/altapay/templates/altapay-payment-form.php
To customize this page, copy the file mentioned above (altapay-payment-form.php) to the folder of the theme that Wordpress is currently using. The file inside the theme folder will override the file inside the plugin folder.
CSS code may be added to this file. Check the documentation about customizing the checkout page here:
https://testgateway.altapaysecure.com/merchant/help/FrontendIntegration#Page_Design___Customization

Auto-capture when order status changes to "Completed"

When the status of an order is manually changed to 'Completed', the plug-in will automatically try to capture this order – there is no need of any setup for this.

If the order was already fully or partially captured, no capture will be made.

Troubleshooting

PHP Warning: Input variables exceeded 1000

For orders that contain many products, this PHP warning may be issued. In the file "php.ini" there is a setting called "max_input_vars" that need to have the limit increased (i.e. from 1000 to 3000). Once the changes are made a restart to the web server is required.

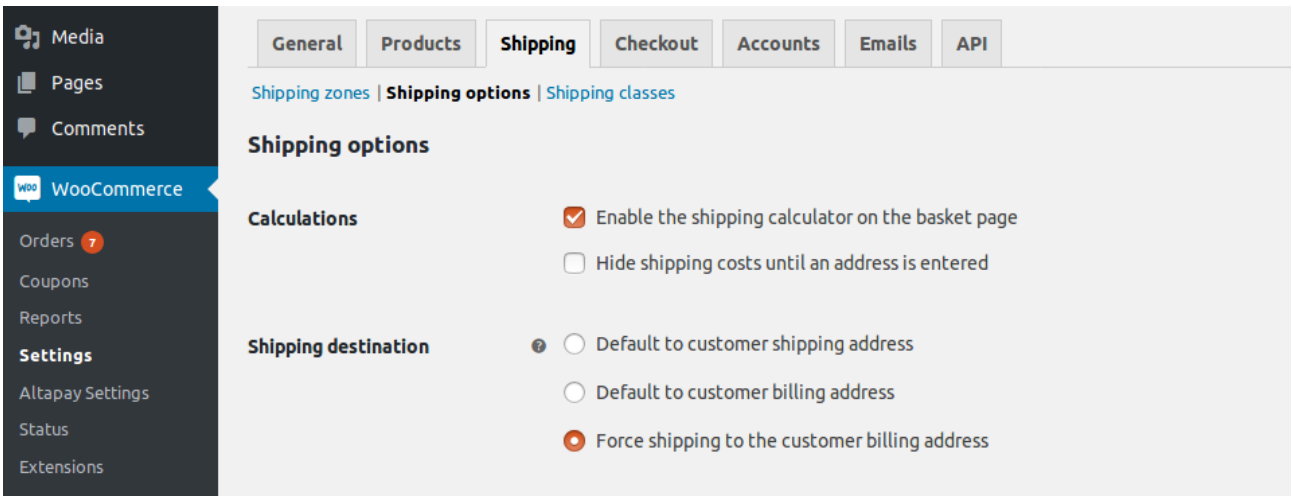
Description/UnitPrice/Quantity is required for each orderline, but it was not set for line: xxxx

The same problem as above: the request has been truncated because the number of variables are exceeding the max_input_vars limit.

Order failed: Could not create the payment request

For the case when checkout fails with error "Could not create the payment request" in most cases the issue is that Shipping Options were not set to "Force shipping to the customer billing address" and the detailed error in the Order notes will be "customer_info[shipping_country] was not a valid country".

The setting is located is in: WooCommerce → Settings → Shipping → Shipping Options:



For any other error messages shown in the "Order notes" section, please contact support for assistance!

Changelog

Version	Note
2016	First release
25-04-2017	Troubleshooting section added
02-05-2017	Auto-capture section added
20-11-2017	FAQ Section is updated
14-12-2018	Documentation rebranded to Valitor