Maksuturva ®

Maksuturva
Payment Gateway Module
PrestaShop 1.4. - 1.7.

eMaksut
Suomen Maksuturva Oy

Revision 210

SuomenMaksuturva

Table of Contents

1   Changes

Date	Module revision	Description
04/16/12	101	Initial release.
07/13/12	108	Handling of multibyte characters fixed. Previous module versions may run into hash calculation problems with some characters or character encodings.
07/17/12	110	The module is now compatible beginning with Prestashop version 1.4.3
03/15/13	120	Fixes to handling product name and description (use other one if the other is blank). Also some changes in the common code. No effect on functionalities in this implementation.
08/06/14	121	Bugfixes: • the module now supports Prestashop up to version 1.6. • the module now removes quotation marks from product names and description – previously it caused errors in hash calculation • product SKU is added when available, shipping method name used as shipping name when available • Optional parameters (e.g. pre-selected payment method, buyer's identification code) are now included in the hash calculation when available (currently not available, though, but enabling is possible in further development - see `_construct` -method in `MaksuturvaGatewayImplementation.php` and Maksuturva Payment interface description). • Module directory structure updated to enable installation through Prestashop user interface. Documentation updated.
10/16/14	122	New: • Support for additional payment fees in response message. • Product article number referes now to product reference number. If reference number is not available, then to EAN13-number if available. • A default Finnish translation package added for the payment module. Bugfixes: • Fixed translation variables and messages. Note: this change will overwrite ALL previous module translations. See section 5.1 Upgrading an existing module for instructions. • Payment module sends user locale to Maksuturva (the customer sees Maksuturva payment page in that language by default) Documentation updated

08/14/15	123	Bugfix: fixed secret key version number validation to support integer validation changes in Prestashop 1.6.1.0 Bugfix: ignores escrow parameter (that is, if the service is eMaksut or Maksuturva) in responses. Previously Maksuturva did not accept payments with false escrow value (that is, if webstore admin has set eMaksut as "active" even though the used payment service is a Maksuturva service, or vice versa). Today Maksuturva accepts false escrow values in payments but responses with service's actual escrow value. This has caused modules to reject succesful payments if the webstore admin has had false value for "eMaksut" option.
02/18/16	200	Changed: - Complete refactoring of module source code to follow PrestaShop standards and best practices. Fixed: - Fixed conflicting messages on "thank you"-page after successful payment of orders containing backorder products. Removed: - Removed eMaksut option from module admin, and always send "Y" as value for the "escrow" field. Maksuturva handles this mode internally, and the setting in the module has no effect. Added: - Configurable payment prefix that is added to the "pmt_id" field if set.
05/15/17	210	Added: - Support for PrestaShop 1.7.x

2   System requirements

The Maksuturva Payment Gateway module for PrestaShop was tested on and requires the following set of applications in order to fully work:

Requirement	Supported versions
PrestaShop	• 1.4.1. - 1.7.1.1
PHP	• 5.3 or higher
PHP cURL Module	• Any
PHP mbstring module	• n/a

There is no guarantee that the module is fully functional in any other environment which hasn't such requirements.

3   Limitations and notes

The following limitations have been found in this revision of the payment module or in the web store software in question:

- Even though the module automatically validates payment status for delayed payments (see section 8.1), it does not validate status for payments that have returned with error or cancel values.

- Additional payment fees are added to the order total in PrestaShop. PrestaShop does not support having custom "line-items" attached to the order that would show up like the "Shipping costs", however, a notice (including the fee amount) is added to all orders where an additional payment fee has been added. This notice is shown on the order admin page, the order detail page (customer account) and in the PDF invoice.

4 Package contents

This guide assumes that all the package files were correctly downloaded by the user from Maksuturva's GitHub releases page:

https://github.com/maksuturva/prestashop_payment_module/releases

File	Description
backward_compatibility/*	PrestaShop 1.4. compatibility classes that enables the module to use newer PrestaShop features. These classes are not used when running PrestaShop 1.5. or newer.
controllers/*	Controller classes for handling requests on the checkout page and from Maksuturva when validating a payment.
docs/*	Module documentation.
includes/*	Class files included by the main module class to separate business logic into logical containers.
views/*	Images, CSS files and template files used by the module both in the frontend and the backend.
config.xml	Module config.
en.php, fi.php	Module translations.
LICENSE.txt	Module license.
logo.gif, logo.png	Maksuturva logos used by PrestaShop (.gif for 1.4. and 1.5., .png for 1.6.)
maksuturva.php	Main module file.
payment.php	Entry script for handling checkout page requests in older PrestaShop versions where module controllers were not supported. This file re-routes the request to the newer controller using the backward compatibility features.
README.md	Module readme file.
validation.php	Entry script for handling maksuturva requests in older PrestaShop versions where module controllers were not supported. This file re-routes the request to the newer controller using the backward compatibility features.

5 Installation

Prior to any change in your environment, it is strongly recommended that a backup of the entire existing PrestaShop installation is performed.

5.1 Upgrading an existing module

PrestaShop Smarty cache can sometimes prevent module upgrades to take effect. For these situations, see e.g.: http://www.blmodules.com/prestashop-tutorials-users/prestashop-cache-when-modification-do-not-see/).

Please note that the new module also rewrites all the previous translations. If you have made any custom translations to your module, backup the module's folder before the upgrade and after upgrading rewrite the required translations one by one manually in PrestaShop admin. See chapter 7 Language customization.

5.2 Installing the module

The simplest way of installing or upgrading the module is to use PrestaShop admin user interface. From the main menu choose "Modules" and "Add a new module" ("Add a module form my computer" in PrestaShop 1.4). Upload the .zip file that you have previously download from Maksuturva's GitHub release pages. See chapter 4.
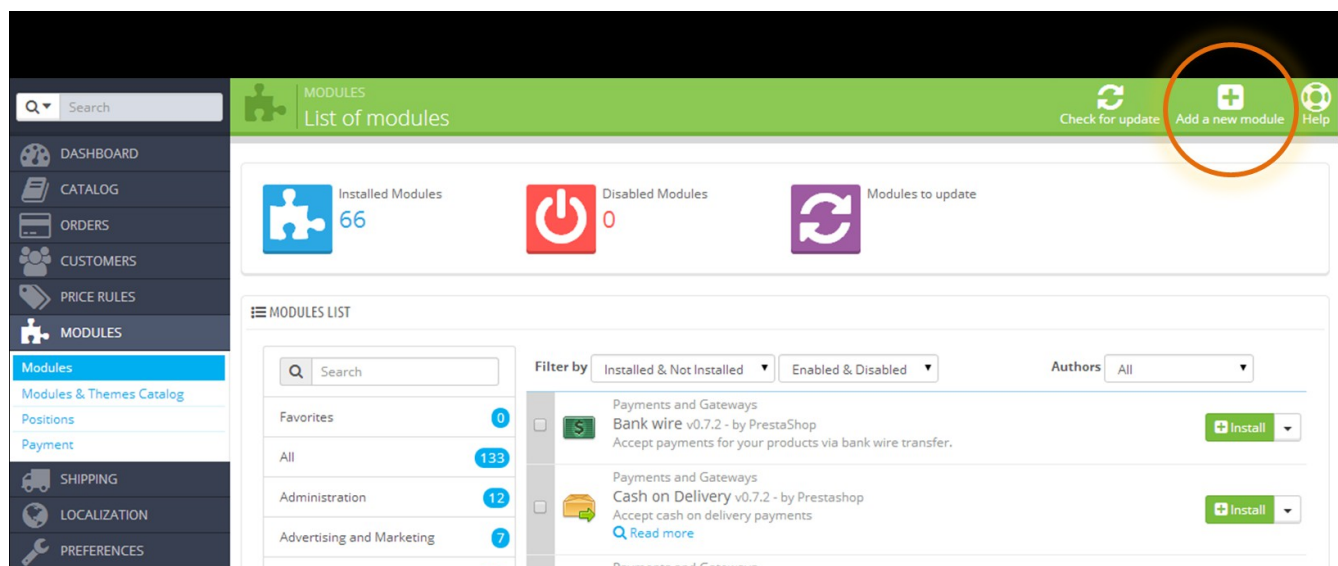

Illustration 1: Add a new module option in PrestaShop 1.6

Another option is to copy the uncompressed maksuturva directory with its content to the *modules* directory in your PrestaShop installation. Both methods can be used for upgrading the module.

5.3 PrestaShop setup

When the module is uploaded into PrestaShop, open the administrative panel of your PrestaShop installation and go to the modules section. Choose "install" as shown in Illustration 2. The module is now shown as installed, and the configuration can be performed.
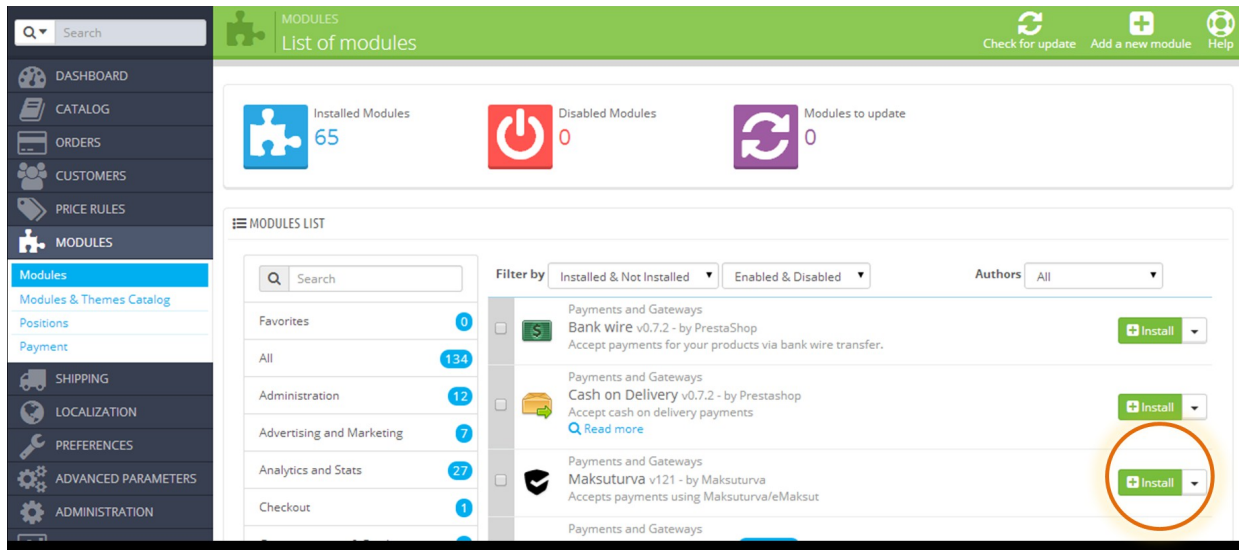
Illustration 2: Install the uploaded payment module in PrestaShop 1.6

6   Configuration

The configuration section for Maksuturva's module is reached by accessing the installed modules, as shown in the setup section. The parameters and their possible configurations are presented in Illustration 3. All the configuration entries are detailed below. Remember to save the changes after your actions.

It is strongly encouraged to test the module and payment service before using the module in production mode with actual money transfers. The settings varies depending on if the module is used in production mode, for testing in Maksuturva test environment or for sandbox testing. Read more about testing in chapter 9 Testing the installation.

Illustration 3: Maksuturva payment module configuration options

6.1   Seller ID

The seller id is your account identifier that you have received from Maksuturva. It is used in every transaction inside your store. Copy the value and paste it in this field.

When used for testing in *Maksuturva test environment* use your test environment seller id.

When used in *sandbox mode*, general test account is used and the value filled in this option will be ignored.

6.2   Secret key

Together with the Seller ID, the secret key is also provided by Maksuturva. Check your secret key and input this value into the configuration field. Please note that this key must not be shared with any person, since it allows many operations to be done in your Maksuturva account.

When used for testing in *Maksuturva test environment* use your test environment secret key.

When used in *sandbox mode*, general test account is used and the value filled in this option will be ignored.

6.3   Secret key version

This parameter is also provided by Maksuturva. Check your secret key version and input this value into the configuration field. Please note that this key must not be shared with any person, since it

allows many operations to be done in your Maksuturva account.

6.4   Communication URL

This configures which URL the payment and query status operations will send the data to Maksuturva. The default value 'https://www.maksuturva.fi' refers to Maksuturva *production environment*.

The communication URL for Maksuturva *test environment* is 'http://test1.maksuturva.fi' (note, that transfer protocol for test environment is *http*, not *https*).

6.5   Payment prefix

Optional payment prefix that is prepended to the payment IDs in Maksuturva. This can be useful when re-installing the module to prevent ID collisions.

6.6   Sandbox mode

You can use module in Sandbox test mode through activating the Sandbox mode. When this option is selected, default test parameters (*Seller ID* and *Secret key*) will be used in your store payments. Purchases made during sandbox will be directed to sandbox test page to verify payment information module is sending to Maksuturva as well as responses from Maksuturva service to PrestaShop.

When set to "Inactive", the production mode will be used. If *Maksuturva production service* credentials and communication URL parameters are used real commercial transactions will be performed when payments are done. Using *Maksuturva test credentials and test environment URL* makes possible to test the full payment process without actual money transactions.

Note that the default installation value for this configuration entry is "Active".

## 6.7   Communication encoding

There are two possible character encodings that are accepted by Maksuturva. Even if you are using a different encoding the module will attempt to convert the data from your PrestaShop encoding to the encoding set on this configuration. You might need to test which of them works better for your encoding by using the sandbox mode. The possible values are 'ISO-8859-1' and 'UTF-8' and they are selected from radio button options. The default value for this setting is 'UTF-8'.

Wrong encoding might cause the hash verification fail in Maksuturva service so if you get errors in Maksuturva service page informing that hash value is invalid, you should check if changing the encoding value fixes the problem.

7 Language customization

It is possible to customize the language and messages used in the module to the languages enabled in PrestaShop language settings. The customization takes effect by changing the text entries using the administrative tool, accessed by the modules configuration page (see Illustration 4) or though Localization menu (see Illustration 5) by selecting "Installed modules translations" from "Type of translation" menu. Look for the section "Module: maksuturva", and apply all the changes for the selected language.

In PrestaShop 1.6, when translations are accessed through Localization menu it is also possible to make different translations for each theme (See "select your theme" -option in Illustration 5) that overrule core translations.
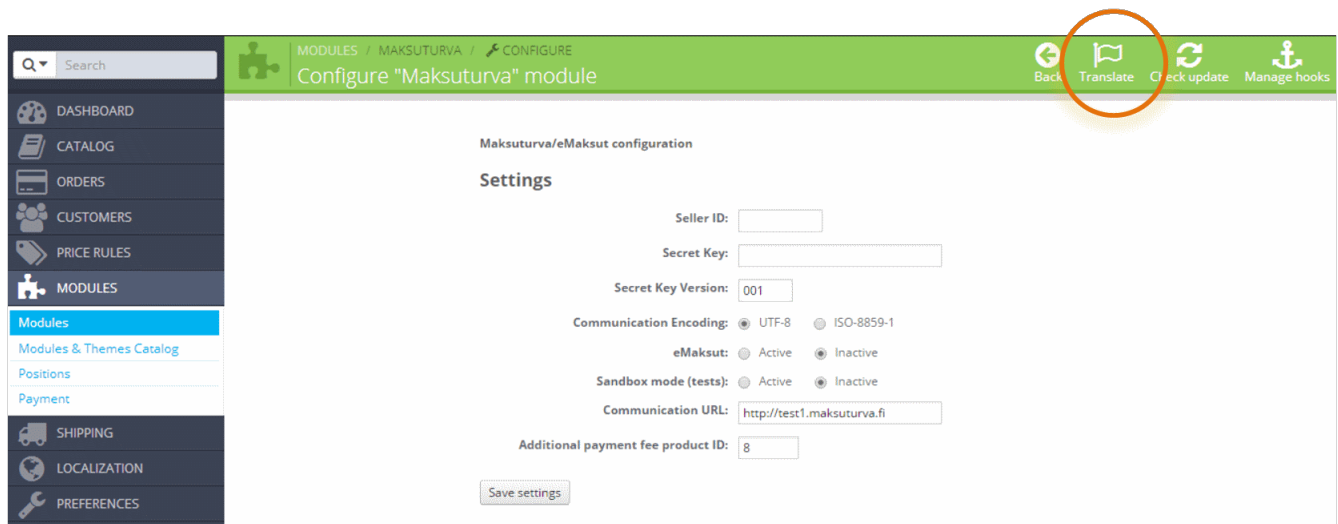
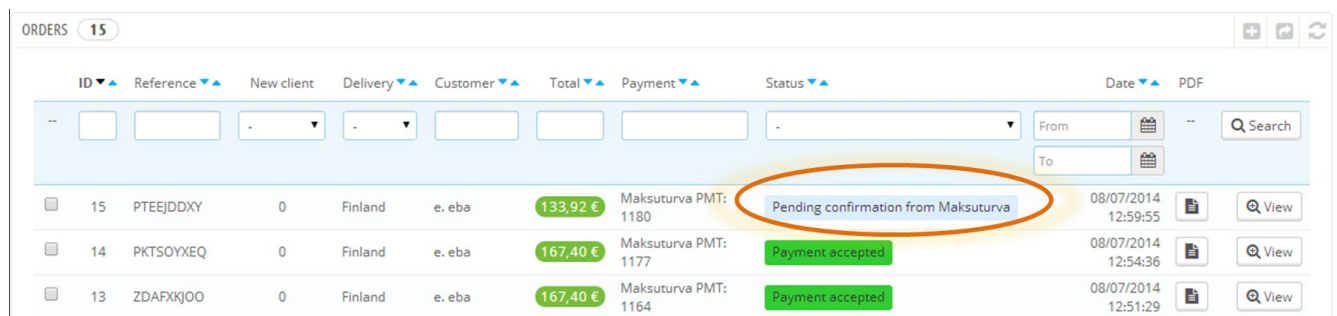*Illustration 4: Accessing module translations through module configuration page*
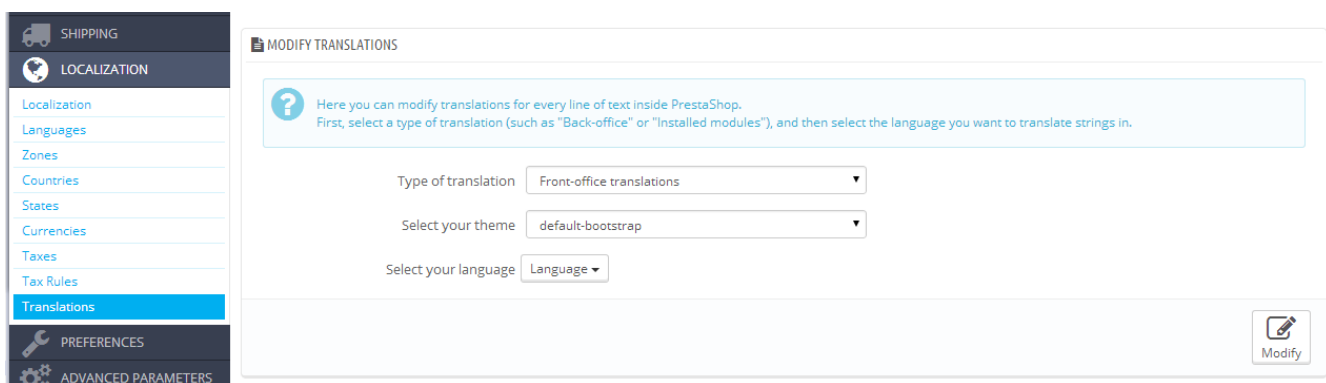
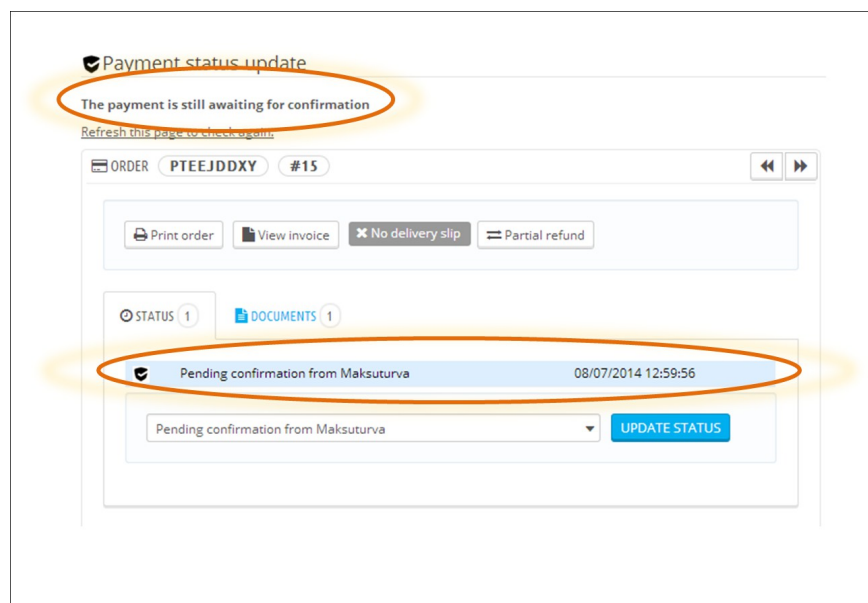*Illustration 5: Accessing module translations through Localization menu*

8 Administration

8.1 Delayed payments (if available)

When the buyer has paid the order by credit card, online bank payment, invoice, part payment etc. Maksuturva service will respond to PrestaShop that the payment has been *accepted*. The Maksuturva Gateway Payment module has also a support for delayed payments when the payment method is available (see Illustration 6). For that purpose, an order which payment has not yet been verified can be automatically verified by opening the order itself. When an order with an unverified payment is opened, the verfication process will automatically start. If verification is received from Maksuturva payment status will change from unverified (see Illustration 7) to verified (Illustration 8).
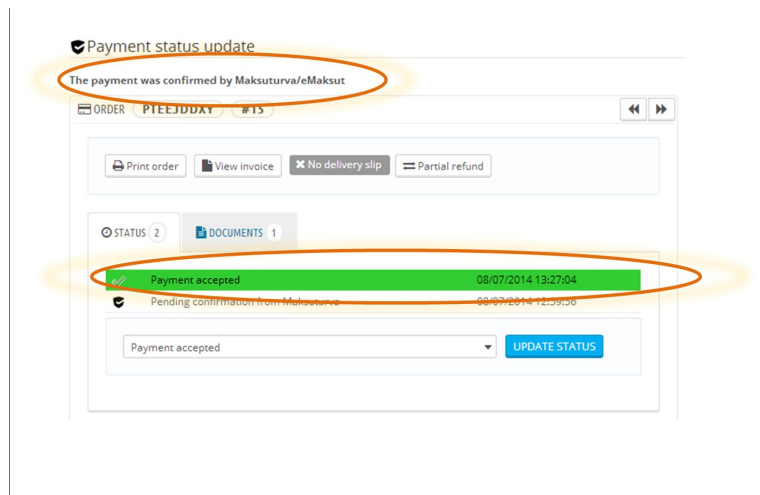

Illustration 6: Unconfirmed payment in PrestaShop Orders list

Illustration 7: Order view with an unverfied payment

Illustration 8: Order view with a verified payment

8.2 Accepting additional payment fees

After the customer has chosen Maksuturva as his payment method, he is redirected to Maksuturva service. In the payment service can choose a payment method from multiple options. The available payment methods depend on webstore's service agreement on Maksuturva.

For invoice and part payment payments it is possible to issue an extra invoicing fee. If a customer chooses to pay the order with this kind of a method, Maksuturva charges the extra fee from the customer. When customer returns to PrestaShop, the invoicing fee is added to the order total. This means that the total amount of the order will change on every page it is displayed, while there will be NO "line-item", like the "Shopping cost", because PrestaShop does not support it. Instead, there will be a notice on the pages that the order was subject to an additional payment surcharge of X amount.

9 Testing the installation

It is strongly encouraged that the Maksuturva payment module is thoroughly tested before taking it into use in on production web store. It is essential to verify that customer and product information, prices, taxes and discount as well as payment message verifying hash are transmitted correctly to Maksuturva.

9.1 Sandbox testing

First, one should validate payment messages by general a.k.a sandbox testing by activating Sandbox mode (see chapter 6.4 Communication URL). Activating the Sandbox mode, the module will use default test credentials and all confirmed orders are directed to a Sandbox test page where one can see the sent payment message data. Also all possible errors in data are presented.

The payment module calculates a checksum / hash from the payment data and webstore secret key combination. Secret key itself is never sent. When Maksuturva service receives the payment message it verifies the authenticity by doing the same calculation and comparing the two checksums.

If Sandbox test page reports errors in hash, the following checks are recommended:

- all the message data on the Sandbox test page are accurate and match the ones that have been sent. No extra spaces or symbols are added or removed and no symbol is changed to something else neither.

- Check that the selected encoding is accurate and change it if required (see chapter Error: Reference source not foundError: Reference source not found).

- Check if product information (e.g. name, description) includes problematic special symbols which might cause yet unreported errors (see chapter 3 Limitations and notes)

- If the product information is copied from word processing software (Office Word etc.), the copied text might include markings (header, bulletpoint and emphasis markings etc.) which – even if they are not visible – cause errors in hash calculation. In these cases the product information data should be verified by copying it to more simple text editor (e.g. Notepad) which removes all possible invisible symbols and then copied back to PrestaShop replacing previous product information content.

Sandbox test page also shows links to responses to webstore (see Illustration 9). By selecting appropriate link, one can verify the information and messages that buyer and store administration see when payment is e.g. accepted or canceled.

Ok paluu / OK return Palaa välittäen ns request stringissä maksamisen vahvistusviestin osoitteeseen, joka tuli pmt_okreturn -kentässä.
Virhe paluu / Error return Palaa ilman lisättyjä parametreja osoitteeseen, joka tuli pmt_errorreturn -kentässä
Peruutus paluu / Cancel return Palaa ilman lisättyjä parametreja osoitteeseen, joka tuli pmt_cancelreturn -kentässä
Viivästetty maksu (tilisiirto) paluu / Delayed payment (giro) return Palaa ilman lisättyjä parametreja osoitteeseen, joka tuli pmt_delayedpayreturn -kentässä

Illustration 9: Return options in Maksuturva Sandbox test page

9.2 Testing in Maksuturva test environment

When Sandbox mode is disabled, module acts as in production mode and order confirmation directs buyer to Maksuturva payment page. If actual Maksuturva credentials are used real money transactions will occur.

If communication URL is set to refer to Maksuturva test environment (http://test1.maksuturva.fi) it is possible to try the full online bank payment and invoice payment process without using actual money. Test environment uses different credentials than Maksuturva production environment and test environment service needs to be ordered separately from the actual Maksuturva service. More information about testing is available on Maksuturva web site www.maksuturva.fi.
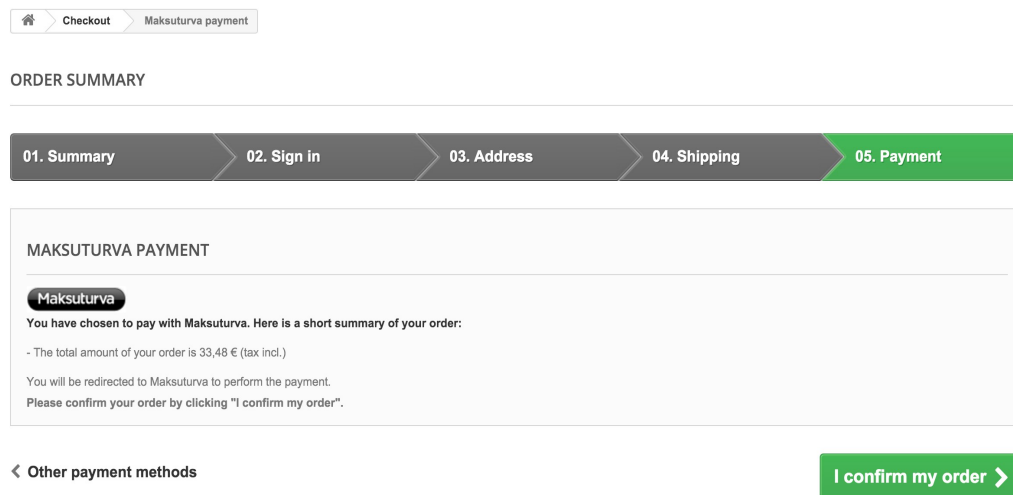
10 Paying through Maksuturva payment gateway

Step 1 - Log in as a customer. Perform a purchase of items you choose and select Maksuturva as the payment option during the checkout (see Illustration 10).

*Illustration 10: Selecting Maksuturva as a payment method*

Step 2 – Confirm the order

Illustration 11: Confirming the order

Step 3 – If using Sandbox mode verify payment by clicking "I confirm my order". If using Maksuturva test or production environment, choose the payment method from the available options and complete the payment process.

Step 4 – Now the payment was confirmed. Buyer sees the order confirmation page (see Illustration 12) and store administration the new confirmed order (see e.g. illustrations 6 and 8 at page 13).

🏠 〉 Order confirmation
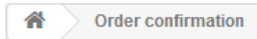
ORDER CONFIRMATION

Your order on SMT testikauppa is complete.

Maksuturva

Your order will be shipped as soon as possible.

For any questions or for further information, please contact our customer support.

‹ Back to orders

Illustration 12: Order confirmation message to buyer after one has been redirected back to PrestaShop from Maksuturva service