Operating instructions implementation Fast Checkout

MultiSafepay
Fast Checkout

Index
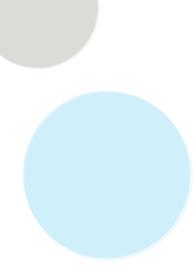
MultiSafepay

Introduction

This manual describes how to implement Fast Checkout in your webshop. The document contains information about the flow of a Fast Checkout process, the implementation of Fast Checkout and the corresponding XML structures.

Before we go to the technical part of this manual, we first explain what Fast Checkout actually is. MultiSafepay Fast Checkout is MultiSafepay's latest payment service. With Fast Checkout customers can pay quickly, easily and safely on the internet.

With Fast Checkout customers will be able to pay:

- in 3 steps only
- in a secure environment
- where all transactions are orderly together in just one account
- and every purchase is guaranteed up to € 200,-

MultiSafepay Fast Checkout operates not only perfectly alongside an existing checkout process, but also operates perfectly as a single checkout process.

MultiSafepay Fast Checkout process

A Fast Checkout transaction starts in the webshop as soon as the customer has added items in the shopping cart. The shopping cart will then be checked out via Fast Checkout. The Fast Checkout process consists of 3 steps.

Step 1.
If the customer already has a Fast Checkout account, the checkout simply occurs by entering only an e-mail address and password (refer to image 1). If the customer doesn't have an account yet, then an account will be created during the checkout, just once (refer to image 2).

Step 2.
The customer selects the relevant shipping method and shipping address. In this area it is possible to show extra fields, like for example a birth date.

Step 3.
The customer chooses the preferred payment method and returns to the webshop.
All order information will be returned to the webshop after running the Fast Checkout process. This will occur via the standard notification system.

Image 1.

| Webshop | Shopping cart overview | Login | Confirmation | Pay |

Image 2.

| Webshop | Shopping cart overview | Create account | Confirmation | Pay |

Implementation

A Fast Checkout implementation starts in the shopping cart of the webshop. The customer will be routed via his shopping cart to Fast Checkout in order to checkout his purchases.

Request
In the request for a Fast Checkout transaction the shopping cart needs to be sent along, this will enable Fast Checkout to show all information during the checkout process. Also all possible shipping methods with corresponding costs needs to be sent along. During the Fast Checkout process the customer will select the relevant shipping method and enter his shipping address or select an address out of a list of saved addresses.

Verwerken van transactie van **EUR 310,00** voor **Demoshop**.

Aantal	Product	Prijs
29	Houten trein	€ 290,00
2	Poppenwagen	€ 20,00
	Verzendkosten:	-
⊖ Minimaliseren	**Totaal:**	**€ 310,00**

Kies de verzendmethode en het afleveradres

Verzendmethode:
- ○ TNT - verzending NL € 7.00
- ○ Ophalen € 0.00

Verzend naar:
Jan Modaal
Blastraat 23
1234AB Amsterdam
Nederland
Wijzig

Both the shopping cart and the shipping method need to be sent in the *Google Checkout* format. We recommend to use the standard *Google Checkout* libraries. This way no direct knowledge of the XML structure will be needed.

If there is already customer data available, because the customer is already logged in the webshop, this data can also be included in the checkout request to ease the checkout process. This data will be shown upfront during registration and login.

MultiSafepay

If additional customer information is needed to complete the order, Fast Checkout enables the merchant to create extra fields during the checkout process. These fields can be included in it's corresponding checkout request or turned on per website via the MultiSafepay account. To keep the implementation as simple as
possible there already are a few pre-defined fields available, which can be turned on or off.

Vraag	Ja ▼
Vraag	○ Ja ⦿ Nee Dit is een testveld
Test	test2
Company	MultiSafepay2

Feedback

After the customer has checked out via Fast Checkout, the webshop will request a notification URL. The web shop can request for all transaction information via a status request. The status request includes among others: transaction information, all information included in the shopping cart, customer information, the shipping method and data available from the extra fields.

If the order has been pre-generated, the order can be complemented with the information obtained. You also have the choice to generate the order as soon as the feedback from Fast Checkout occurs.

Other settings

MultiSafepay Fast Checkout enables you to show your own logo during the Fast Checkout process. You can upload your logo via your MultiSafepay account and select your logo by going to 'websites'.

Request structure

Transaction

A Fast Checkout transaction request has the following (XML) structure:

checkouttransaction	(root node)
merchant	merchant data
customer	possible available customer data
customer-delivery	possible available delivery address
transaction	general transaction details
checkout-shopping-cart	shopping cart in *Google Checkout* format
custom-fields	extra fields shown during checkout
signature	MD5 signature

The fields in bold are additional next to the normal MultiSafepay transaction request.

A successful Fast Checkout transaction response has the following (XML) structure:

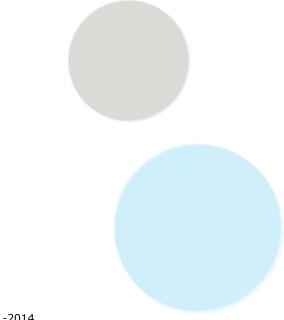|	
checkouttransaction	(root node)
transaction	
id	given transaction id
payment_url	payment link for the customer

The customer needs to be routed to the payment_url, where the customer can run the transaction process.

MultiSafepay

Via the status request all order data and transaction data will be sent back. A status request has the following (XML) structure. The request is similar to MultiSafepay' standard status request.

status		(root node)
	merchant	merchant data
	transaction	
	id	transaction ID included in the transaction request

A Fast Checkout status response has the following (XML) structure:

status		(root node)
	ewallet	MultiSafepay transaction details
	customer	data / customer' invoice address
	customer-delivery	customer' delivery address
	transaction	merchant transaction details
	paymentdetails	payment details
	checkoutdata	order information (shopping-cart, shipping method, extra fields)

The fields in bold are additional next to the normal MultiSafepay status response.

Transaction request

checkouttransaction	(root node)
merchant	merchant data
customer	possible available customer' data
customer-delivery	possible available delivery address
transaction	general transaction details
checkout-shopping-cart	shopping cart in Google Checkout format
custom-fields	extra fields shown during checkout
signature	MD5 signature

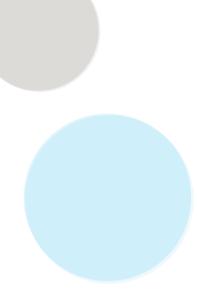The transaction request has as root node **checkouttransaction**.

```
<checkouttransaction ua="Magento 1.1.1">
        …
</checkouttransaction>
```

The attribute **ua** stands for User Agent and needs to show a self chosen string describing the implementation.

MultiSafepay

field	Value	Description
account	Numerical	Account ID
site_id	Numerical	Site ID
site_secure_id	Numerical	Site Code
notification_url	Text	Notification URL
cancel_url	Text	This is where the customer returns during the transaction when pressing 'Cancel'.
redirect_url	Text	The customer will automatically return to this URL after the transaction.
close_window	true/false	Will close the window after the transaction.

Important fields:

notification_url

This URL will be used by the system to notify the webshop that the status order has changed. The URL will be requested with a GET-variable (transactionid). This GET-variable processes the status request. This URL will be generated at the end of the transaction process. The HTML output of the URL will be shown on the last window and often contains a link back to the webshop.

cancel_url

If the transaction request contains a cancel_url, the customer has the possibility to cancel the transaction and return to the relevant URL (web page).

redirect_url

If you enclose the redirect_url, the last step of the transaction process will not be visible for the consumer. The consumer will automatically be routed to the relevant URL (web page). When using the redirect_url, the HTML output of the notification_url will not be shown.

MultiSafepay

Customer

If customer data is available, the data could be enclosed via the customer section.

Fields	Value	Description
locale	AB_cd	Language (ISO 639) and Country (ISO 3166)
ipaddress	n.n.n.n	Customer' IP-address
forwardedip	n.n.n.n	Customer' IP-address, when via proxy
firstname	Text (25)	First name
lastname	Text (25)	Last name
address1	Text (64)	Address line 1 (street name)
address2	Text (64)	Address line 2
housenumber	Text (10)	House number
zipcode	Text (10)	Postal code
city	Text (50)	City
country	AB	Country code (ISO 3166)
phone	Text (25)	Phone number
email	Text (50)	E-mail address

Important fields:

local

The 'local' determines the language of the payment area for the customer.

email

The e-mail address of the customer will be used a.o. for sending the payment confirmation and 'Second Chance'* e-mails from MultiSafepay.

* Second Chance

It often happens that someone doesn't finish his payment during the payment process. Mostly the customer doesn't have the proper tools within reach; a digital card reader or the right information. In this case MultiSafepay automatically send an e-mail with a payment link. This way the customer can finish the transaction at another time.

MultiSafepay

If customer data is available, the data can be enclosed via the customer-delivery section.

Fields	Value	Description
firstname	Text (25)	First name
lastname	Text (25)	Last name
address1	Text (64)	Address line 1 (street name)
address2	Text (64)	Address line 2
housenumber	Text (10)	House number
zipcode	Text (10)	Postal code
city	Text (50)	City
country	AB	Country code (ISO 3166)
phone	Text (25)	Phone number

MultiSafepay

Checkout shopping cart

The section of the **checkout shopping cart** contains information needed to check out the shopping cart. It contains among others the items of the shopping cart and the available shipping methods. The **checkout shopping cart** consists of the *Google Checkout Cart format.*

```
checkout-shopping-cart
        shopping-cart
                items
                        item
                        item
                        item

        checkout-flow-support
                merchant-checkout-flow-support
                        shipping-methods
                                flat-rate-shipping
                                flat-rate-shipping
                                pickup
                        tax-tables
                                default-tax-table
                                alternate-tax-tables
```

Shopping-cart

```
<shopping-cart>
        <items>
                <item>
                        <item-name>aaa</item-name>
                        <item-description></item-description>
                        <unit-price currency="EUR">10.0000</unit-price>
                        <quantity>2</quantity>
                        <merchant-item-id>111</merchant-item-id>
                </item>
                <item>
                        <item-name>bbb</item-name>
                        <item-description></item-description>
                        <unit-price currency="EUR">12.0000</unit-price>
                        <quantity>3</quantity>
                        <merchant-item-id>123</merchant-item-id>
                        <tax-table-selector>BTW6</tax-table-selector>
                </item>
        </items>
</shopping-cart>
```

MultiSafepay

A shopping-cart item could contain de following fields:

item-name	Text	Name of the product
item-description	Text	Description
unit-price	Number	Price per item
quantity	Number	Number of items
merchant-item-id	Text	Product ID in the web shop
tax-table-selector	Text	Possible name of alternative tax table.
item-weight	Complex	Weight of the product <item-weight unit="KG" value="5.5" />
merchant-private-item-data	Complex (XML Sequence)	Extra product data <merchant-private-item-data> <item-data>Test<item-data> <extra-name>Name2<extra-name> </merchant-private-item-data>

Fields in bold are obligatory.

__Shipping-methods__

```
<checkout-flow-support>
 <merchant-checkout-flow-support>
  <shipping-methods>

    <flat-rate-shipping name="TNT - verzending NL">
     <price currency="EUR">7</price>
     <shipping-restrictions>
      <allowed-areas>
       <postal-area>
        <country-code>NL</country-code>
       </postal-area>
      </allowed-areas>
     </shipping-restrictions>
    </flat-rate-shipping>

    <flat-rate-shipping name="UPS – Verzending Worldwide">
     <price currency="EUR">30</price>
    </flat-rate-shipping>

    <pickup name="Ophalen">
     <price currency="EUR">0</price>
    </pickup>

   </shipping-methods>
  </merchant-checkout-flow-support>
```

```
</checkout-flow-support>
```

Within the section of the shipping-methods more shipping methods could be enclosed. It is also possible to enclose a pickup option.

The following two tags are possible:

- flat-rate-shipping
- pickup

In both the pickup options as well as the flat-rate-shipping a price should be enclosed.
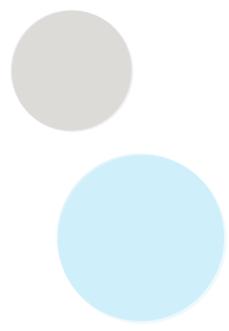
Country restrictions

For a flat-rate-shipping option it is possible to enclose country restrictions (shipping-restrictions). You could enclose both allowed-areas as well as excluded-areas. If enclosed the shipping method will or will not be shown when using specified countries based on the country of the delivery address.
If no restrictions are enclosed the shipping method will always be shown.

Countries need to contain the following XML structure:

```
<postal-area>
   <country-code>NL</country-code>
</postal-area>
```

Countries enclosed in allowed-areas are always permitted. If 'allowed-areas' is not empty, all other countries are not permitted.
If 'allowed-areas' is empty or if country code 'ALL' has been used, all countries are permitted with exception of the countries mentioned in the excluded-areas.

MultiSafepay

Example shipping methods:

```
<flat-rate-shipping name="TNT - verzending NL">
 <price currency="EUR">7</price>
 <shipping-restrictions>
  <allowed-areas>
   <postal-area>
    <country-code>NL</country-code>
   </postal-area>
  </allowed-areas>
 </shipping-restrictions>
</flat-rate-shipping>

<flat-rate-shipping name="TNT - verzending BE">
 <price currency="EUR">12</price>
 <shipping-restrictions>
  <allowed-areas>
   <postal-area>
    <country-code>BE</country-code>
   </postal-area>
  </allowed-areas>
 </shipping-restrictions>
</flat-rate-shipping>

<flat-rate-shipping name="TNT - verzending overige">
 <price currency="EUR">25</price>
 <shipping-restrictions>
  <excluded-areas>
   <postal-area>
    <country-code>NL</country-code>
   </postal-area>
   <postal-area>
    <country-code>BE</country-code>
   </postal-area>
  </excluded -areas>
 </shipping-restrictions>
</flat-rate-shipping>
```

In below example three shipping methods are defined:

- Shipping to the Netherlands – € 7,-
- Shipping to Belgium – € 12,-
- Shipping to all other countries – € 25,-

All amounts excluding VAT should be enclosed. The taxes will be calculated in the Fast Checkout section using the given tax-tables.

The default-tax-table determines the tax calculated for items without a specified table-tax-selector. If the shipping-taxed notifies 'true', this rate will also be included in the calculation of the shipping methods.

The alternate-tax-tables could contain a few alternative tax-tables. These tables have the attribute called 'name'. This name could then be enclosed in the table-tax-selector with an item. For that specific item the alternative table will be used.

```xml
<checkout-flow-support>
 <merchant-checkout-flow-support>
  <shipping-methods>

  </shipping-methods>
  <tax-tables>

   <default-tax-table>
    <tax-rules>
     <default-tax-rule>
      <shipping-taxed>true</shipping-taxed>
      <rate>0.19</rate>
     </default-tax-rule>
    </tax-rules>
   </default-tax-table>

   <alternate-tax-tables>

    <alternate-tax-table standalone="true" name="BTW19">
     <alternate-tax-rules>
      <alternate-tax-rule>
       <rate>0.19</rate>
      </alternate-tax-rule>
     </alternate-tax-rules>
    </alternate-tax-table>

    <alternate-tax-table standalone="true" name="BTW6">
     <alternate-tax-rules>
      <alternate-tax-rule>
       <rate>0.06</rate>
      </alternate-tax-rule>
     </alternate-tax-rules>
    </alternate-tax-table>

    <alternate-tax-table standalone="true" name="BTW0">
     <alternate-tax-rules>
      <alternate-tax-rule>
       <rate>0.00</rate>
      </alternate-tax-rule>
     </alternate-tax-rules>
    </alternate-tax-table>

   </alternate-tax-tables>
  </tax-tables>
```

```
    </merchant-checkout-flow-support>
</checkout-flow-support>
```
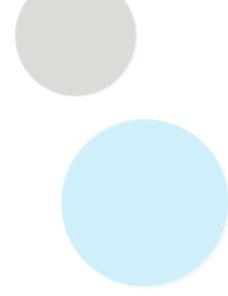
In the examples below are four tables defined:

- Standard table with a tax rate of 19%. This rate also applies for the shipping costs.
- Alternative table 'BTW19' with a rate of 19%.
- Alternative table 'BTW6' with a rate of 6%.
- Alternative table 'BTW0' with a rate of 0%.

Custom-fields

If extra customer data is required, these data can be requested via the custom fields. With custom fields extra data can be enclosed, which will be shown as a form on the checkout page. A field could contain a **standardtype** field, or it should describe the whole field. Currently the following standard fields are available:

| msp_birthday | Birth date of the customer |
| msp_comment | Space for an extra message when ordering |

Example:

```
<custom-fields>
  <field>
    <standardtype>birthday</standardtype>
  </field>
  <field>
    <standardtype>comment</standardtype>
  </field>
</custom-fields>
```

Besides a standard type it is also possible to enclose own custom fields. A field could contain the next content:

name	Text	Name of the field
type	Text	Type (textbox, selectbox, checkbox , radiolist)
label	Text	Name which will be shown
default	Text	Default value
options	Complex	List with options (concerning selectbox en radiolist)
validation	Complex	Validation of the field
description-top	Complex	Extra description above the field
description-right	Complex	Extra description on the right of the field
description-bottom	Complex	Extra description below the field
savevalue	true/false	Save the value of this customer. Data of this field will be shown upfront in a next transaction.
field-restrictions	Complex	Possible country restrictions for showing the fields

Example:

```
<custom-fields>
  <field>
    <name>testvraag</name>
    <type>radiolist</type>
    <label>Vraag</label>
    <default>no</default>
    <options>
      <option>
        <value>yes</value>
        <label>Ja</label>
      </option>
      <option>
        <value>no</value>
        <label>Nee</label>
      </option>
    </options>
    <description-right>
      <value>In de vorm dd/mm/jj</value>
      <style>small</style>
    </description-right>
    <validation>
      <regex>[a-zA-Z0-9 ]+</regex>
      <error xml:lang="nl">Testvraag mag alleen letters en cijfers bevatten</error>
    </validation>
  </field>
</custom-fields>
```

Language

The different text and labels shown on the payment page could be shown in different languages via an xml:lang attribute.

```
<custom-fields>
  <field>
    <name>vraag</name>
    <type>textbox</type>

    <label xml:lang="nl">Vraag</label>
    <label xml:lang="fr">Quistione</label>
    <label xml:lang="en">Question</label>
    <label xml:lang="de">Frage</label>

  </field>
</custom-fields>
```

This is possible for the below fields:

- field/label
- field/options/option/label
- field/description-top/value
- field/description-bottom/value
- field/description-right/value
- field/validation/error

Country restrictions

It is possible to only show that specific field to customers from a particular country by enclosing field-restrictions (in the same size as the shipping-restrictions with the shipping methods).

```
<field-restrictions>
 <allowed-areas>
  <postal-area>
   <country-code>NL</country-code>
  </postal-area>
 </allowed-areas>
</field-restrictions>
```

Validation

It is possible to do a validation check on the field that needs to meet certain values.

regex	Text	Regular expression that meets the value
error	Text	Error if the validation didn't succeeded

```
<validation>
 <regex>[a-zA-Z0-9 ]+</regex>
 <error xml:lang="nl">Testvraag mag alleen letters en cijfers bevatten</error>
</validation>
```

MultiSafepay

Example XML request

```xml
<?xml version="1.0" encoding="UTF-8"?>
<checkouttransaction ua="Magento 1.1.1">
 <merchant>
  <account>1234567</account>
  <site_id>123</site_id>
  <site_secure_code>12345</site_secure_code>
  <notification_url> http://example.com /notify/ </notification_url>
  <cancel_url>http://example.com /cancel/</cancel_url>
  <redirect_url></redirect_url>
  <close_window></close_window>
 </merchant>
 <customer>
  <locale>nl</locale>
  <ipaddress>87.65.43.21</ipaddress>
  <forwardedip></forwardedip>
  <firstname></firstname>
  <lastname></lastname>
  <address1></address1>
  <address2></address2>
  <housenumber></housenumber>
  <zipcode></zipcode>
  <city></city>
  <state></state>
  <country></country>
  <phone></phone>
  <email></email>
 </customer>
 <customer-delivery>
  <firstname></firstname>
  <lastname></lastname>
  <address1></address1>
  <address2></address2>
  <housenumber></housenumber>
  <zipcode></zipcode>
  <city></city>
  <state></state>
  <country></country>
  <phone></phone>
  <email></email>
 </customer-delivery>
 <checkout-shopping-cart xmlns="">
  <shopping-cart>
   <items>
    <item>
     <item-name>Test</item-name>
     <item-description></item-description>
     <unit-price currency="EUR">10.0000</unit-price>
     <quantity>29</quantity>
     <merchant-item-id>test123</merchant-item-id>
     <item-weight unit="LB" value="20" />
    </item>
    <item>
     <item-name>aaa</item-name>
     <item-description></item-description>
     <unit-price currency="EUR">10.0000</unit-price>
```

Telefoon 020-8500500 Fax 020-8500501 Postbus 37159 1030 AD Amsterdam E-mail info@multisafepay.com Website www.multisafepay.com KVK nummer: 34245794

```xml
        <quantity>2</quantity>
        <merchant-item-id>111</merchant-item-id>
        <tax-table-selector>BTW6</tax-table-selector>
        <item-weight unit="LB" value="1" />
      </item>
    </items>
  </shopping-cart>
  <checkout-flow-support>
    <merchant-checkout-flow-support>
      <shipping-methods>
        <flat-rate-shipping name="TNT - verzending NL">
          <price currency="EUR">7</price>
          <shipping-restrictions>
            <allowed-areas>
              <postal-area>
                <country-code>NL</country-code>
              </postal-area>
            </allowed-areas>
          </shipping-restrictions>
        </flat-rate-shipping>
        <flat-rate-shipping name="TNT - verzending BE en FR">
          <price currency="EUR">12</price>
          <shipping-restrictions>
            <allowed-areas>
              <postal-area>
                <country-code>BE</country-code>
              </postal-area>
              <postal-area>
                <country-code>FR</country-code>
              </postal-area>
            </allowed-areas>
          </shipping-restrictions>
        </flat-rate-shipping>
        <pickup name="Ophalen">
          <price currency="EUR">0</price>
        </pickup>
      </shipping-methods>

      <tax-tables>
        <default-tax-table>
          <tax-rules>
            <default-tax-rule>
              <shipping-taxed>true</shipping-taxed>
              <rate>0.19</rate>
            </default-tax-rule>
          </tax-rules>
        </default-tax-table>
        <alternate-tax-tables>
          <alternate-tax-table standalone="true" name="BTW19">
            <alternate-tax-rules>
              <alternate-tax-rule>
                <rate>0.19</rate>
              </alternate-tax-rule>
            </alternate-tax-rules>
          </alternate-tax-table>
          <alternate-tax-table standalone="true" name="BTW6">
            <alternate-tax-rules>
```
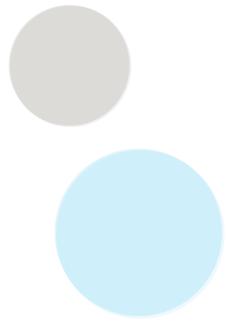
```
        <alternate-tax-rule>
          <rate>0.06</rate>
        </alternate-tax-rule>
      </alternate-tax-rules>
    </alternate-tax-table>
    <alternate-tax-table standalone="true" name="BTW0">
      <alternate-tax-rules>
        <alternate-tax-rule>
          <rate>0.00</rate>
        </alternate-tax-rule>
      </alternate-tax-rules>
    </alternate-tax-table>
    </alternate-tax-tables>
  </tax-tables>
 </merchant-checkout-flow-support>
 </checkout-flow-support>
</checkout-shopping-cart>

<custom-fields>
 <field>
  <standardtype>birthday</standardtype>
 </field>
 <field>
  <name>test</name>
  <savevalue>true</savevalue>
  <type>textbox</type>
  <label>Test</label>
  <default>test</default>
  <description-right>
    <value xml:lang="nl">in de vorm dd/mm/jjjj</value>
    <style>standard</style>
  </description-right>
  <description-top>
    <value>Dit is een testveld</value>
    <style>small</style>
  </description-top>
 </field>

 <field>
  <name>companyname</name>
  <savevalue>true</savevalue>
  <type>textbox</type>
  <label>Company</label>
  <default></default>
 </field>

 <field>
  <name>blaat</name>
  <type>radiolist</type>
  <options>
    <option>
      <value>yes</value>
      <label xml:lang="nl">Ja</label>
      <label xml:lang="fr">Oui</label>
      <label xml:lang="en">Yes</label>
      <label xml:lang="de">Ja</label>
    </option>
```
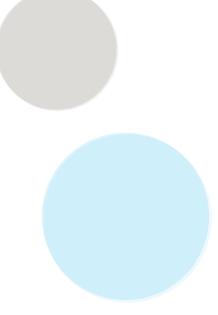
```xml
      <option>
        <value>no</value>
        <label xml:lang="nl">Nee</label>
        <label xml:lang="fr">Non</label>
        <label xml:lang="en">No</label>
        <label xml:lang="de">Nein</label>
      </option>
    </options>
    <description-bottom>
      <value>Ik heb een vraagje...</value>
      <style>small</style>
    </description-bottom>
    <default>no</default>
    <label xml:lang="nl">Vraag</label>
    <label xml:lang="fr">Quistione</label>
    <label xml:lang="en">Question</label>
    <label xml:lang="de">Frage</label>
  </field>
</custom-fields>

<transaction>
  <id>9692729</id>
  <currency>EUR</currency>
  <amount>31000</amount>
  <description>Order #9692729 at test</description>
  <var1></var1>
  <var2></var2>
  <var3></var3>
  <items></items>
  <manual>false</manual>
</transaction>
<signature>66d32f176c790d12b3727d61649c85f9</signature>
</checkouttransaction>
```

XML transaction response

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<checkouttransaction result="ok">
   <transaction>
     <id>9692729</id>
     <payment_url>https://user.multisafepay.com/pay/?trans...</payment_url>
   </transaction>
</checkouttransaction>
```

XML error response

```xml
<?xml version="1.0" encoding="UTF-8"?>
<checkouttransaction result="error">
   <error>
     <code>1013</code>
     <description>MD5 mismatch</description>
   </error>
   <transaction>
     <id>9692729</id>
   </transaction>
</checkouttransaction>
```

Refund transaction

The Refund function allows refunding via the API into the MultiSafepay payment platform. This is done by sending a refundtransaction XML request to the API. Below is an example of a refundtransaction XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<refundtransaction ua="refund">
        <merchant>
                <account>' .        $account_id . '</account>
                <site_id>' .        $site_id . '</site_id>
                <api_key>' .                    $api_key . '</api_key>
                <signature>' .                  $signature . '</signature>
        </merchant>
        <transaction>
                <id>' .          $transaction_id . '</id>
                <amount>' .         $transaction_amount . '</amount>
                <currency>' .          $transaction_currency . '</id>
        </transaction>
</refundtransaction>
```

The following items are important with a refundtransaction:

API Key: This can be found in the MultiSafepay backoffice. This is only visible with an admin or refund user.
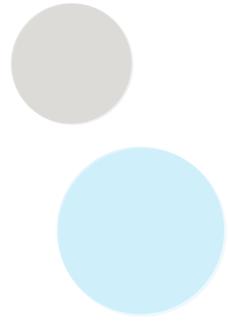
Signature: This is the sha1 hash from: $site_id . $secure_code . $transaction_id

If the amount and currency is not specified, the entire transaction will be refunded.
If a different currency is used for the refund than that of the orinigal transaction an error will be shown.
If an amount is specified, but no currency has been specified, then an error will be shown.

Status request

The status request requests for the payment status of a transaction. At this point all order data returns to the web shop.

The following statuses are possible:

Status	Description
completed	Successfully completed (paid)
initialized	initialized, but not completed yet (paid)
uncleared	initialized, but not cleared yet (possible with credit cards, debit payment)
void	Cancelled
declined	Declined
refunded	Refundend
expired	Expired

Fields

Merchant

	Value	Description
account	Numerical	Account ID
site_id	Numerical	Site ID
site_secure_id	Numerical	Site Code

Transaction

	Value	Description
id	Text (50)	Order ID

MultiSafepay

```xml
<?xml version="1.0" encoding="UTF-8"?>
<status ua="Example 1.0">
   <merchant>
      <account>123456</account>
      <site_id>123</site_id>
      <site_secure_code>12345</site_secure_code>
   </merchant>
   <transaction>
      <id>123</id>
   </transaction>
</status>
```

Status response

status		(root node)
	ewallet	MultiSafepay transaction details
	customer	data / customer' invoice address
	customer-delivery	customer' delivery address
	transaction	merchant transaction details
	paymentdetails	payment details
	checkoutdata	order information (shopping-cart, shipping method, extra fields)
	custom-fields	the values entered by the customer at the custom fields

E-wallet

MultiSafepay transaction details

id	Numerical	Created MultiSafepay transaction ID
status	Numerical	Status of the transaction
created	Numerical	Date created
modified	Numerical	Date last adaptation

Customer

Customer data (invoice address)

firstname	Text	First name
lastname	Text	Last name
address1		
address2		
address3		
housenumber		
city	Text	City
state	Text	State
country	ISO 3166	Countrycode (written as NL)
email		

Customer-delivery

Delivery address of the customer

firstname	Text	First name
lastname	Text	Last name
address1	Text	Address line 1
address2	Text	Address line 2
address3	Text	Address line 3
housenumber	Text	House number
city	Text	City
state	Text	State
country	ISO 3166	Country code (written as NL)

Transaction

Merchant transaction details

id	Numerical	Transaction ID as enclosed
currency	Numerical	Currency
amount		Amount in cents
description		Description of the transaction
var1	Text	Own enclosed variable
var2	Text	Own enclosed variable
var3	Text	Own enclosed variable
items	HTML	Enclosed items in html format

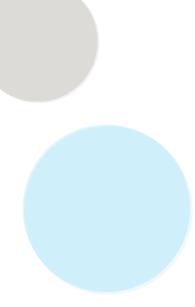| accountid	Numerical	Account number if available
accountholdername	Text	Account holder name
externaltransactionid	Text	External transaction ID / payment number

Checkoutdata

The checkoutdata contains data regarding the order, like shopping cart data enclosed in the transaction request and the chosen shipping method.

Shopping-cart

Shopping cart enclosed in the transaction request:

```xml
<shopping-cart>
      <items>
            <item>
                    <item-name>aaa</item-name>
                    <item-description></item-description>
                    <unit-price currency="EUR">10.0000</unit-price>
                    <quantity>2</quantity>
                    <merchant-item-id>111</merchant-item-id>
            </item>
            <item>
                    <item-name>bbb</item-name>
                    <item-description></item-description>
                    <unit-price currency="EUR">12.0000</unit-price>
                    <quantity>3</quantity>
                    <merchant-item-id>123</merchant-item-id>
            </item>
      </items>
</shopping-cart>
```

Order-adjustment

Contains the chosen shipping method and the adjusted order amount.

```
<order-adjustment>
 <shipping>
  <flat-rate-shipping>
   <shipping-name>TNT - verzending NL</shipping-name>
   <shipping-cost currency="EUR">7.00</shipping-cost>
  </flat-rate-shipping>
 </shipping>
 <adjustment-total currency="EUR">21.37</adjustment-total>
 <total-tax currency="EUR">14.37</total-tax>
</order-adjustment>
```

Order-total

Contains the new order total.

```
<order-total currency="EUR">117.37</order-total>
```

Custom-fields

Contains a list of values from the extra fields, which are entered by the customer. Here the XML tag equals the **name** as enclosed or the name of the **standardfield**.

```
<custom-fields>
 <msp_birthday>19-12-1980</ msp_birthday>
 <companyname>MultiSafepay</companyname>
</custom-fields>
```

MultiSafepay

```xml
<?xml version="1.0" encoding="UTF-8"?>
<status result="ok">
 <ewallet>
  <id>11794</id>
  <status>completed</status>
  <created>20101124111051</created>
  <modified>20101124111054</modified>
 </ewallet>
 <customer>
  <amount>30700</amount>
  <currency>EUR</currency>
  <exchange_rate/>
  <firstname>Jan</firstname>
  <lastname>Jansen</lastname>
  <city>Amsterdam</city>
  <state/>
  <country>NL</country>
  <countryname/>
 </customer>
 <customer-delivery>
  <firstname>Piet</firstname>
  <lastname>Jan</lastname>
  <address1>Hallostraat</address1>
  <address2>address 2</address2>
  <address3>address 3</address3>
  <housenumber>34</housenumber>
  <city>Almere</city>
  <state>state</state>
  <countrycode>NL</countrycode>
  <country>Netherlands</country>
  <phone1>123123</phone1>
  <phone2>2222222</phone2>
  <email>aaa@test.com</email>
 </customer-delivery>
 <transaction>
  <id>123</id>
  <currency>EUR</currency>
  <amount>30700</amount>
  <description>Order #9817853 at test</description>
  <var1/>
  <var2/>
  <var3/>
  <items/>
 </transaction>
 <paymentdetails>
  <accountid>117301582</accountid>
  <accountholdername>Hr J A T Verfürth en/of Mw T V Chet</accountholdername>
  <externaltransactionid>0080000004410089</externaltransactionid>
 </paymentdetails>
 <checkoutdata version="0.1">
  <shopping-cart>
   <items>
    <item>
     <item-name>Test</item-name>
     <item-description/>
```

```xml
      <unit-price currency="EUR">10.0000</unit-price>
      <quantity>28</quantity>
      <merchant-item-id>test123</merchant-item-id>
      <tax-table-selector>2</tax-table-selector>
      <item-weight unit="LB" value="20"/>
    </item>
    <item>
      <item-name>aaa</item-name>
      <item-description/>
      <unit-price currency="EUR">10.0000</unit-price>
      <quantity>2</quantity>
      <merchant-item-id>111</merchant-item-id>
      <tax-table-selector>2</tax-table-selector>
      <item-weight unit="LB" value="1"/>
    </item>
   </items>
  </shopping-cart>
  <order-adjustment>
   <shipping>
    <flat-rate-shipping>
     <shipping-name>TNT - verzending NL</shipping-name>
     <shipping-cost currency="EUR">7.00</shipping-cost>
    </flat-rate-shipping>
   </shipping>
   <adjustment-total currency="EUR">21.37</adjustment-total>
   <total-tax currency="EUR">14.37</total-tax>
  </order-adjustment>
  <order-total currency="EUR">117.37</order-total>
  <custom-fields>
   <msp_birthday>yes</msp_birthday>
   <companyname>MultiSafepay</companyname>
  </custom-fields>
 </checkoutdata>
</status>
```

Example XML error

```xml
<?xml version="1.0" encoding="UTF-8"?>
<status result="error">
  <error>
    <code>1006</code>
    <description>Invalid transaction ID</description>
  </error>
  <transaction>
    <id>123</id>
  </transaction>
</status>
```

Error codes

The following error codes are possible in a XML error response:

Code	Description	Part	Field
0000	OK		
1000	Message type unknown	error XML message	
1001	Invalid amount	transaction	amount
1002	Invalid currency	transaction	currency
1003	Invalid Merchant Account ID	merchant	account
1004	Invalide Merchant Site ID	merchant	site_id
1005	Invalid Merchant Site Security Code	merchant	site_secure_code
1006	Invalid transactie ID	transaction	id
1007	Invalid ip-address	customer	ipaddress, forwardedip
1008	Invalid description	transaction	description
1009	*Reserved*		
1010	Invalid variable	transaction	var1, var2, var3
1011	Invalid Customer Account ID		
1012	Invalid Customer Security Code		
1013	Invalid signature	signature	
1014	Unspecified error		
1015	Account unknown		
1016	Missing data		
1017	Insufficient balance	for customer	
1018	*Reserved*		
1019	*Reserved*		
9999	Unknown error		

Questions?

If you have any questions during the implementation, don't hesitate to contact our technical support team for support. You can contact us on week days from 9.00 hour till 17.30 hour via phone number 020 8500 500. You can also contact our technical support team by using the Live Chat function on our website.