

# Manual implementation fast checkout

**MultiSafepay**   
fast checkout



## Content

<b>CONTENT</b> .....	<b>2</b>
<b>INTRODUCTION</b> .....	<b>3</b>
<b>1.0 MULTISAFEPAY FAST CHECKOUT PROCESS</b> .....	<b>4</b>
<b>2.0 IMPLEMENTATION</b> .....	<b>5</b>
<b>3.0 TRANSACTION FLOW CHART</b> .....	<b>7</b>
<b>4.0 REQUEST STRUCTURE</b> .....	<b>9</b>
4.1 Transaction.....	9
4.2 Merchant .....	9
4.3 Customer.....	10
4.4 Checkout shopping-cart .....	11
4.5 Shopping-cart.....	12
4.6 Shipping-methods .....	13
4.7 Use shipping notification .....	13
4.8 Update variables .....	14
4.9 Country restrictions .....	15
4.10 Tax-tables.....	16
4.11 Custom fields.....	18
4.12 Customer delivery .....	20
4.13 Status .....	20
4.14 Example XML request.....	21
4.15 XML transaction response .....	25
4.16 XML error response .....	25
4.17 XML cart update.....	25
<b>5.0 STATUS REQUEST</b> .....	<b>27</b>
5.1 Fields.....	27
5.2 XML request.....	27
<b>6.0 STATUS RESPONSE</b> .....	<b>28</b>
6.1 E-wallet.....	28
6.2 Customer.....	28
6.3 Customer-delivery .....	28
6.4 Transaction.....	29
6.5 Payment details.....	29
6.6 Checkoutdata .....	29
6.7 Custom-fields .....	30
6.8 Example XML status response .....	31
6.9 Example XML error .....	32
<b>7.0 ERROR CODES</b> .....	<b>33</b>



## 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.

## 1.0 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.



Image 2.



## 2.0 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:		-
<b>Totaal:</b>		<b>€ 310,00</b>

### 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.



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	<input type="text" value="Ja"/>
Vraag	<input type="radio"/> Ja <input checked="" type="radio"/> Nee
Test	<input type="text" value="test2"/> <small>Dit is een testveld</small>
Company	<input type="text" value="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.

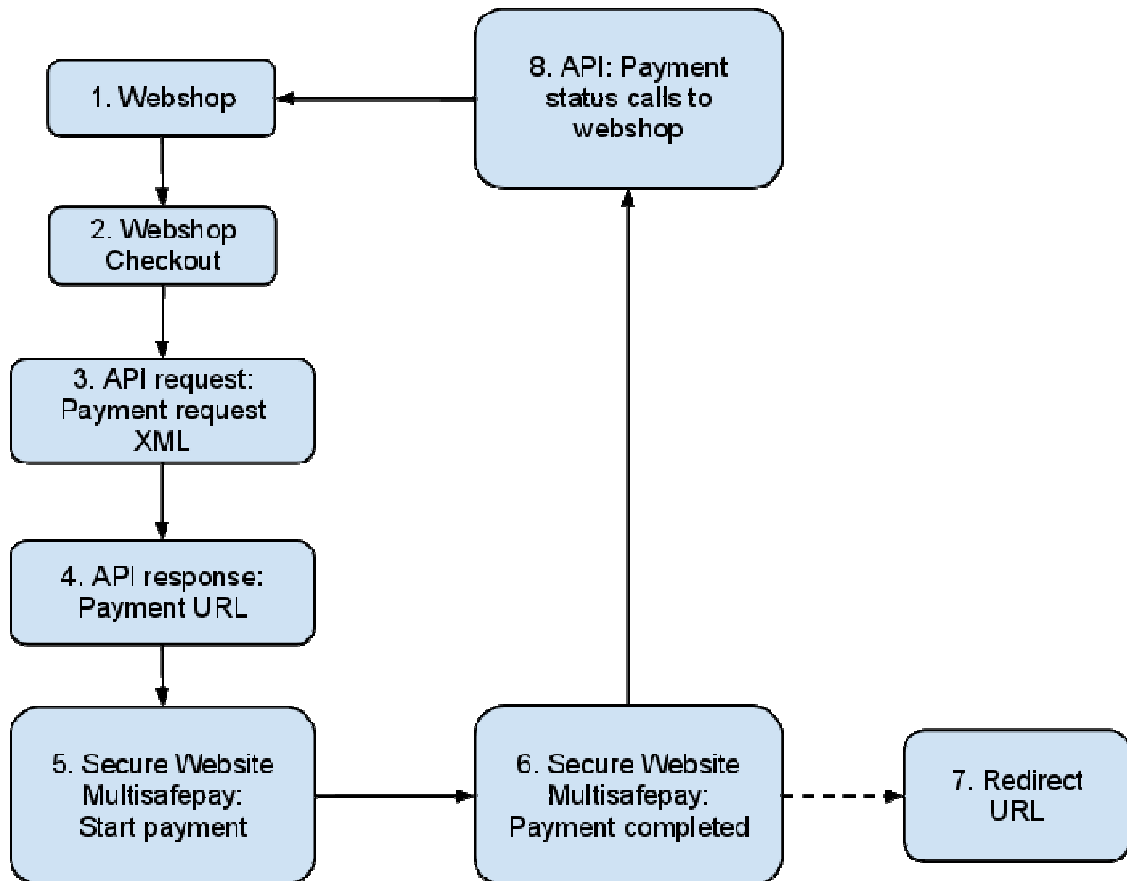
### Other settings

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'.

## 3.0 Transaction flow chart

This section describes the process of a fast checkout transaction using a flow chart. Each step will be explained. Before implementing fast checkout it is extremely important that this process is clear.

### Transaction flow-chart



#### 1. Webshop

This is the starting point from where the consumer starts. For the use of API the choice for the webshop software is not relevant.

#### 2. Webshop checkout

Each transaction starts with the checkout in the webshop. This is almost always the page where the content of the shopping cart will be shown and where the checkout button is placed.

#### 3. API request

From the checkout a XML request will be generated. This XML message must meet the XML specifications specified in the manual. These are derived from the Google Checkout XML API specifications.



#### **4. API response**

After receiving a correct XML message, the API will generate a payment link. Via this URL the consumer can start a payment on the secure website of MultiSafepay. The servers of the MultiSafepay payment environment are updated according to the latest security requirements.

#### **5. Start Payment**

This step means that the consumer has entered the MultiSafepay payment server. The consumer can follow the steps mentioned in section 1.0. Additional questions can be sent along. These will be saved in the transaction.

#### **6. Payment completed**

The checkout process is completed, the consumer completed the payment.

#### **7. Redirect URL**

It is possible to use a redirect URL. This option is often used to let the consumer return to the 'Thank you' page. If this option is not used, the consumer will automatically return to the webshop. When no redirect URL is given, the HTML of the "notification\_URL" will be used.

#### **8. API: Payment status calls to the webshop**

To inform the webshop about the payment statuses MultiSafepay uses a notification system. This enables us to process statuses of the payment in the order of the webshop. The relevant URL can be set up in the MultiSafepay back office.

## 4.0 Request structure

### 4.1 Transaction

A fast checkout transaction request has the following (XML) structure:

checkouttransaction	(root node)
merchant	merchant data
customer	possible available customer data
<b>customer-delivery</b>	possible available delivery address
transaction	general transaction details
<b>checkout-shopping-cart</b>	shopping cart in <i>Google Checkout</i> format
<b>custom-fields</b>	extra fields shown during checkout
<b>checkout-settings</b>	the use of a shipping XML
signature	MD5 signature

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

**TIP: MD5 (Message Digest Algorithm 5) is a common used cryptographic hash function. The hash includes the following values: amount, currency, account\_id, site\_id en transactionid.**

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

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

The attribute **ua** means User Agent and shows a self chosen text, which describes the implementation.

### 4.2 Merchant

field	Value	Description
<b>account</b>	Numerical	Account ID
<b>site_id</b>	Numerical	Site ID
<b>site_secure_id</b>	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.

**TIP: Via this URL it is possible to automate different processes of payment statuses and orders. It may occur that the notification URL will be requested several times containing the same payment status.**

**TIP: The webshop URL will be requested from different servers in the following IP ranges, 213.189.0/122. The webshop URL needs to be accessible from these IP addresses.**

## **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.

### 4.3 Customer

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

<b>Fields</b>	<b>Value</b>	<b>Description</b>
locale	AB_cd	Language (ISO 639) and Country (ISO 3166-1)
ipaddress	n.n.n.n	Customer's IP address
forwardedip	n.n.n.n	Customer's IP address, when via proxy
firstname	Tekst (25)	First name
lastname	Tekst (25)	Last name
address1	Tekst (64)	Address line 1 (street name)
address2	Tekst (64)	Address line 2
houzenumber	Tekst (10)	House number
zipcode	Tekst (10)	Postal code
city	Tekst (50)	City

country	AB	Country code (ISO 3166)
phone	Tekst (25)	Phone number
email	Tekst (50)	Email address

Important fields:

## locale

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

## email

The email 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.*

## 4.4 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
  
```

## 4.5 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>

```

Een shopping-cart item kan de volgende velden bevatten:

item-name	Tekst	Name of the product
item-description	Tekst	Description
unit-price	Getal	Price per item
quantity	Getal	Number of items
merchant-item-id	Tekst	Product ID in the webshop
tax-table-selector	Tekst	Possible name of alternative tax table
item-weight	Attributen	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>

## 4.6 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. Besides predetermining the shipping methods, it is also possible to use a shipping notification.

## 4.7 Use shipping notification

XML also enables you to dynamically load the shipping methods. This XML will be requested via the “notification\_url” in the webshop. This way it is possible to apply certain business rules for the shipping methods.

The shipping call will be done by the following GET variables.

- Type; “shipping”.
- Countrycode; two-letter country code according to ISO 3166 – 1 standard (for example NL is The Netherlands).
- Transactionid; the transaction number of the payment.
- Size; width, height and dept of the package in whole centimetres.
- Weight; weight in kilo's.



An example of a shipping call:

```
http://www.myshop.com/  
notify.php?size=10x20x30&weight=1.0&countrycode=GE&tran  
sactionid=AX6723WHW
```

An example of a shipping call answer from the webshop is:

```
<shipping-info>  
  <shipping>  
    <shipping-name>TNT - verzending NL</shipping-name>  
    <shipping-cost currency="EUR">7.00</shipping-cost>  
  </shipping>  
</shipping-info>
```

An example of a shipping call without possibilities:

```
<shipping-info>  
</shipping-info>
```

#### 4.8 Update variables

The system enables to provide a transaction with all correct web shop data. Because the possibility exists that a transaction number is only available after completion of the payment. Additional data can be added. It allows the merchant to add own data to the transaction.

An example of an updated variables request:

```
<?xml version="1.0" encoding="UTF-8"?>  
<request>  
  <clientinfo>  
    <ipaddress>0.0.0.0</ipaddress>  
    <useragent>api</useragent>  
  </clientinfo>  
  <function>  
    <updatetransaction>  
      <merchanttransactionid>0000</merchanttransactionid>  
      <merchantaccountid>00000000</merchantaccountid>  
      <merchantsecurecode>000000</merchantsecurecode>  
      <merchantsubid>0000</merchantsubid>  
      <fields>  
        <id>111222333</id>  
        <invoiceid>444333</invoiceid>  
        <own_field>any specific info</own_field>  
      </fields>  
    </updatetransaction>  
  </function>  
</request>
```

## 4.9 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.

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,-

#### 4.10 Tax-tables

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.

```
<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%.

## 4.11 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.

It is also possible to enclose own custom fields. A field could contain the next content:

<b>name</b>	Tekst	Name of the field
<b>type</b>	Tekst	Type (textbox, selectbox, checkbox , radio)
<b>label</b>	Tekst	Name which will be shown
<b>default</b>	Tekst	Default value
<b>options</b>	Complex	List with options (applies to selectbox and radio)
<b>validation</b>	Complex	Validation of the field
<b>description-top</b>	Complex	Extra description above the field
<b>description-right</b>	Complex	Extra description on the right of the field
<b>description-bottom</b>	Complex	Extra description below the field
<b>savevalue</b>	true/false	Save the value of this customer. Data of this field will be shown upfront in a next transaction.
<b>field-restrictions</b>	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-bottom>
    <validation>
      <regex>[a-zA-Z0-9 ]+</regex>
      <error xml:lang="nl">Testvraag mag alleen letters en cijfers bevatten</error>
    </validation>
  </field>
</custom-fields>
```

## Talen

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.

<b>regex</b>	Tekst	Regular expression that meets the value
<b>error</b>	Tekst	Error if the validation didn't succeed

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

**TIP: Error notifications are possible in different languages. This allows you to send notifications to international customers**

## 4.12 Customer delivery

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
houzenumber	Text (10)	House number
zipcode	Text (10)	Postal code
city	Text (50)	City
country	AB	Country code (ISO 3166-1)
phone	Text (25)	Phone number

A successful fast checkout response has the following (XML) structure:

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

The consumer needs to be sent to the payment\_url, where the consumer can follow the transaction process.

## 4.13 Status

All order data and transaction data will be returned via the status request. A status request has the following (XML structure). The request is identical to the MultiSafepay standard status request.

status	(root node)
merchant	merchant details
transaction	
id	transaction ID as given in the transaction request

A fast checkout status response has the following (XML) structure:

status	(root node)
ewallet	MultiSafepay transaction details

customer	details / consumer's invoice address
<b>customer-delivery</b>	consumer's delivery address
transaction	merchant transaction details
paymentdetails	payment details
<b>checkoutdata</b>	order information (shopping-cart, shipping method, extra fields)

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

#### 4.14 Example XML request

```
<?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>
    <houenumber></houenumber>
    <zipcode></zipcode>
    <city></city>
    <state></state>
    <country></country>
    <phone></phone>
    <email></email>
  </customer>
  <customer-delivery>
    <firstname></firstname>
    <lastname></lastname>
    <address1></address1>
    <address2></address2>
    <houenumber></houenumber>
    <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>
  <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>
  </merchant-checkout-flow-support>
</checkout-flow-support>
<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>msp_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>
  <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>
```

## 4.15 XML transaction response

```
<?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>
```

## 4.16 XML error response

```
<?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>
```

## 4.17 XML cart update

During the fast checkout process it's possible to update the "shopping cart" by activating certain "custom fields". The addition of "custom fields" is previously described in this document. This feature can be used, for example, for adding the option gift service. This will cause positive as well negative price changes, as the total shopping cart will be renewed.

To add this feature, the following element needs to be added to the transaction request checkout settings:

```
<checkout-settings>
  <use-field-notification>true</use-field-notification>
</checkout-settings>
```

Then it's possible per "custom field" to add the element "notification" to create an update request. This request will be send to the notification url with the concerning field and filled value. The type of request, via the GET variable is "cart". Also the transaction ID of the Merchant will be enclosed.

```
<field>
<name>regularcostumer</name>
<type>selectbox</type>
<savevalue>>true</savevalue>
<label xml:lang="nl">Bent u vaste klant?</label>
<label xml:lang="en">Are you a regular customer?</label>
<notification>true</notification>
<options>
  <option>
    <value>1</value>
    <label xml:lang="nl">Ja</label>
    <label xml:lang="en">Yes</label>
  </option>
  <option>
    <value>0</value>
    <label xml:lang="nl">Nee</label>
    <label xml:lang="en">No</label>
  </option>
</options>
</field>
```

## 5.0 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 (not paid yet)
uncleared	initialized, but not cleared yet (possible with credit cards, debit payment)
void	void
declined	declined
refunded	refunded
expired	Expired

### 5.1 Fields

#### Merchant

	Waarde	Omschrijving
account	Numerical	Account ID
site_id	Numerical	Site ID
site_secure_id	Numerical	Site Code

#### Transaction

	Waarde	Omschrijving
id	Text (50)	Order ID

### 5.2 XML request

```
<?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>
```

## 6.0 Status response

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

### 6.1 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 adjustment

### 6.2 Customer

Customer's details (invoice address)

<b>firstname</b>	Text	First name
<b>lastname</b>	Text	Last name
<b>address1</b>	Text	Address line 1
<b>address2</b>	Text	Address line 2
<b>address3</b>	Text	Address line 3
<b>hounumber</b>	Text	House number
<b>city</b>	Text	City
<b>state</b>	Text	State
<b>country</b>	ISO 3166-1	Country code (for example NL)
<b>email</b>	Text	

### 6.3 Customer-delivery

Customer's delivery address.

<b>firstname</b>	Text	First name
<b>lastname</b>	Text	Last name
<b>address1</b>	Text	Address line 1
<b>address2</b>	Text	Address line 2
<b>address3</b>	Text	Address line 3
<b>hounumber</b>	Text	House number
<b>city</b>	Text	City
<b>state</b>	Text	State
<b>country</b>	ISO 3166-1	Country code (for example NL)

## 6.4 Transaction

Merchant transaction details

<b>id</b>	Numerical	Transactie ID zoals meegegeven
<b>currency</b>	Numerical	Currency
<b>amount</b>	Numerical	Amount
<b>description</b>	Text	Transaction description
<b>var1</b>	Text	Variable
<b>var2</b>	Text	Variable
<b>var3</b>	Text	Variable
<b>items</b>	HTML	Items in html format

## 6.5 Payment details

Payment details

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

## 6.6 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:

```
<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>
```

## 6.7 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>
  <misp_birthday>19-12-1980</ misp_birthday>
  <companyname>MultiSafepay</companyname>
</custom-fields>
```

## 6.8 Example XML status response

```
<?xml version="1.0" encoding="UTF-8"?>
<status result="ok">
  <ewallet>
    <id>11794</id>
    <status>completed</status>
    <fastcheckout>YES</fastcheckout>
    <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>
    <houzenumber>34</houzenumber>
    <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>
      </items>
    </shopping-cart>
  </checkoutdata>
</status>
```

```
<item-description/>
<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>
  <misp_birthday>yes</misp_birthday>
  <companyname>MultiSafepay</companyname>
</custom-fields>
</checkoutdata>
</status>
```

## 6.9 Example XML error

```
<?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>
```

## 7.0 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	Invalid 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	<i>Reserved</i>		
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	<i>Reserved</i>		
1019	<i>Reserved</i>		
9999	Unknown error		

**TIP: Error code 1014 often occurs due to the wrong combination of: Account ID, Site ID, Site Security Code in combination with a test or live account.**

## Questions?

If you have any questions during the implementation, don't hesitate to contact our technical support team. You can contact us via [techsupport@multisafepay.com](mailto:techsupport@multisafepay.com). You can also contact our technical support team by using the Live Chat function on our website.