Checkout Integration Guide
Payment API

Revision History

DATE	DESCRIPTON
22/May/2014	Overview

Copyright

UNIVERSO ONLINE S/A. Address: Avenida Faria Lima, 1384, 6° Floor, São Paulo/SP, ZIP 01452-002, Brazil.
The PagSeguro services are not, nor are intended to be, comparable to financial services offered by financial institutions or credit card companies, and consist simply of a way to enable and monitor the implementation of e-commerce transactions through the management of payments. Any transaction made through PagSeguro is subject to and must comply with the laws of the Federative Republic of Brazil.
We recommend that you read the terms and conditions carefully.

Disclaimer

UOL offers no guarantees of any kind [implicit, implied or statutory] with respect to the information contained therein. UOL does not undertake any liability for loss or damage (direct or indirect), caused by error or omission, or resulting from the use of this document or the information contained herein or resulting from the application or use of a product or service described in this document. UOL is entitled to make any changes to any information herein without prior notice.

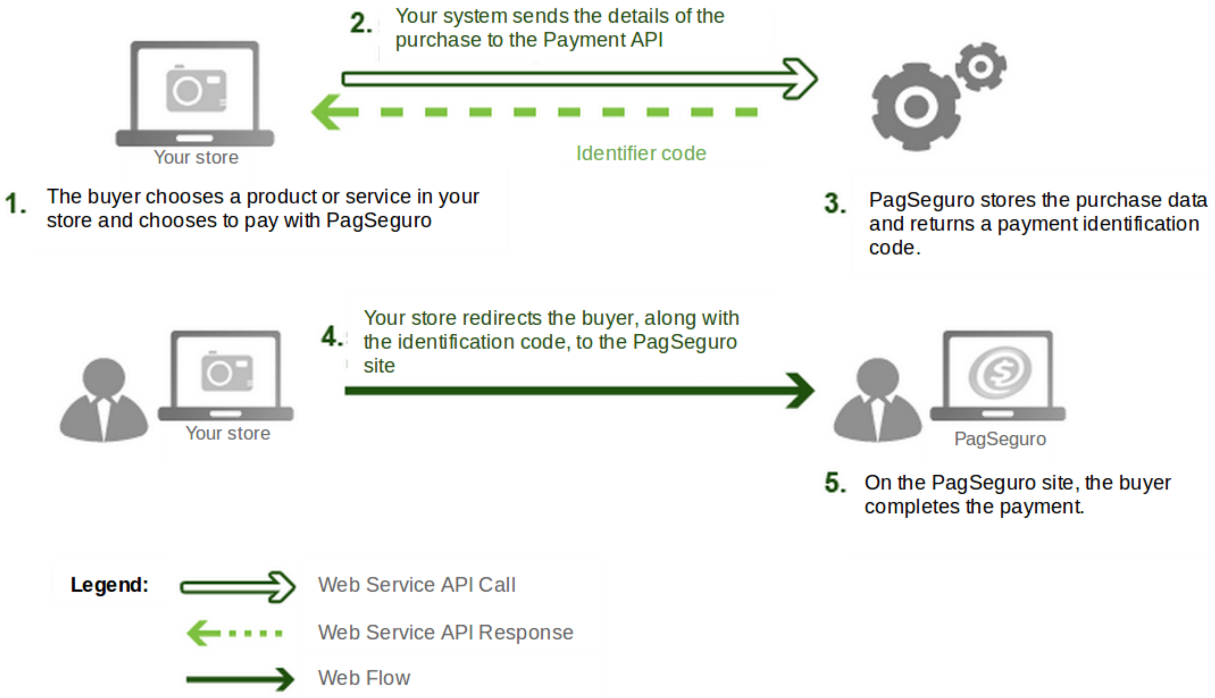
Contents

Payment API

What you'll find here:

- Description of the data in HTTP parameters
- Description of the data in XML format
- Description of the parameters in the Payment API
- Description of the Payment API Responses
- Sending the buyer into the payment flow

Using the payment API offers greater control and flexibility over the payment process, however, it requires programming knowledge or the use of an integration plugin or library.



1. The buyer chooses a product or service in your store and chooses to pay with PagSeguro

2. Your system sends the details of the purchase to the Payment API

3. PagSeguro stores the purchase data and returns a payment identification code.

4. Your store redirects the buyer, along with the identification code, to the PagSeguro site

5. On the PagSeguro site, the buyer completes the payment.

To initiate the payment process, you need to make a call to the PagSeguro Payment API sending the payment details. This request creates an identifier code, and with this code, you can direct the buyer to the PagSeguro website, where he/she can complete the payment. When calling the Payment API, you can provide the payment data in two ways: as HTTP parameters or in XML format.

Sending data as HTTP parameters

One way to make a request to the Payment API is to send the payment data directly as HTTP parameters using the **POST** method.
This format is very similar to how payments via HTML, the difference being that the call is made to the Payment API instead of the web page endpoint.

URL of the Payment API of PagSeguro:

```
POST https://ws.sandbox.pagseguro.uol.com.br/v2/checkout
```

The **Content-Type** header must be informed as shown below:

```
Content-Type: application/xml; charset=ISO-8859-1
```

Note: If your application or store does not use a set of **ISO-8859-1** characters, i.e. (**UTF-8**), you will need to set the charset parameter in the example above accordingly.
Every payment transactions can contain one or more items. Every item represents a product, item or service that is being purchased. The parameter names associated with items end in a number.
For example: parameters **itemId1**, **itemDescription1**, **itemAmount1** and **itemQuantity1** refer to the first item of payment, while parameters itemId2, itemDescription2, itemAmount2 and itemQuantity2 refer to the second item of payment.
See below a complete example of a request made to the Payment API of PagSeguro using HTTP parameters (the content has been broken into lines for easier reading).

```
curl https://ws.sandbox.pagseguro.uol.com.br/v2/checkout/ -d\
         "email=suporte@lojamodelo.com.br\
         &token=95112EE828D94278BD394E91C4388F20\
         &currency=BRL\
         &itemId1=0001\
         &itemDescription1=Notebook Prata\
         &itemAmount1=24300.00\
         &itemQuantity1=1\
         &itemWeight1=1000\
         &itemId2=0002\
         &itemDescription2=Notebook Rosa\
         &itemAmount2=25600.00\
         &itemQuantity2=2\
         &itemWeight2=750\
         &reference=REF1234\
         &senderName=Jose Comprador\
         &senderAreaCode=11\
         &senderPhone=56273440\
         &senderEmail=comprador@uol.com.br\
         &shippingType=1\
         &shippingAddressStreet=Av. Brig. Faria Lima\
         &shippingAddressNumber=1384\
         &shippingAddressComplement=5o andar\
         &shippingAddressDistrict=Jardim Paulistano\
         &shippingAddressPostalCode=01452002\
         &shippingAddressCity=Sao Paulo\
         &shippingAddressState=SP\
        &shippingAddressCountry=BRA"
```

Sending data as XML

Another way to make the request to the Payment API is to provide the payment data as XML. This format may be preferable if your system already works with XML.

In this format you send the XML content of an HTTP request using the POST.

An advantage of using the XML format is that the data is sent in a more structured way.

See below the URL of PagSeguro's Payment API:

```
https://ws.sandbox.pagseguro.uol.com.br/v2/checkout
```

See below the required **HTTP headers**

The **Content-Type** header must be informed as shown below:

```
Content-Type: application/xml; charset=ISO-8859-1
```

Note: If your application or store does not use the ISO-8859-1 character set, i.e. (UTF-8), you will need to set the charset parameter in the example above accordingly.

When making the request, it is necessary to provide the e-mail and the account token of your store or application as query string parameters in the URL as shown below:

```
email=suporte@lojamodelo.com.br&token=95112EE828D94278BD394E91C4388F20
```

See example below an example of structured XML for payment data:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
<checkout>
    <currency>BRL</currency>
    <items>
        <item>
            <id>0001</id>
            <description>Notebook Prata</description>
            <amount>24300.00</amount>
            <quantity>1</quantity>
            <weight>1000</weight>
        </item>
        <item>
            <id>0002</id>
            <description>Notebook Rosa</description>
            <amount>25600.00</amount>
            <quantity>2</quantity>
            <weight>750</weight>
```

```xml
        </item>
    </items>
    <reference>REF1234</reference>
    <sender>
        <name>José Comprador</name>
        <email>comprador@uol.com.br</email>
        <phone>
            <areaCode>11</areaCode>
            <number>56273440</number>
        </phone>
    </sender>
    <shipping>
        <type>1</type>
        <address>
            <street>Av. Brig. Faria Lima</street>
            <number>1384</number>
            <complement>5o andar</complement>
            <district>Jardim Paulistano</district>
            <postalCode>01452002</postalCode>
            <city>Sao Paulo</city>
            <state>SP</state>
            <country>BRA</country>
        </address>
    </shipping>
</checkout>
```

Description of Payment API parameters

The parameters expected by the Payment API are described in the table below.

PARAMETER	DESCRIPTION
HTTP header: **charset**	**Character encoding.** Specifies the character encoding used in the parameters sent. **Presence:** Optional. **Type:** Text. **Format:** Accepted values are **ISO-8859-1** and **UTF-8**.
HTTP parameter: **email**	**E-mail of the account that calls the API.** Specifies the e-mail address associated with the PagSeguro account that is making the call to API. **Presence:** Required. **Type:** Text. **Format:** a valid email (e.g., usuario@site.com.br), with a maximum of 60 characters.
HTTP parameter: **token**	**Token of the account that calls the API.**

PARAMETER	DESCRIPTION
	Specifies the token that corresponds to the PagSeguro account that is making the call to API. To create a token for your PagSeguro account, access the payment setup page. **Presence:** Required. **Type:** Text. **Format:** a sequence of 32 characters.
XML element: **<checkout>**	**This field is the root of the XML file and includes the payment details.**
HTTP parameter: **receiverEmail** XML Element: <checkout> <receiver> **<email>**	**Specifies the email that should appear on the payment screen.** **Presence:** Optional. **Type:** Text. **Format:** a valid email (e.g., usuario@site.com.br), with a maximum of 60 characters. The e-mail provided must be linked to the PagSeguro account that is making the call to API.
HTTP parameter: **currency** XML Element: <checkout> **<currency>**	**Currency used.** Indicates the currency in which payment will be made. At the moment, the only option available is **BRL** (Real). **Presence:** Required. **Type:** Text. **Format:** Case-sensitive Only the **BRL** value is accepted.
XML element: <checkout> **<items>**	**List of items contained in the payment.**
XML element: <checkout> <items> **<item>**	**Represents a payment item.**
HTTP parameter: **itemId1, itemId2, etc.** XML element: <checkout> <items> <item> **<id>**	**Item identifiers.** Identify the items being paid. You can choose codes that have a meaning in your system and provide them in these parameters. PagSeguro does not perform any validation on these identifiers. They may repeat within the same payment. **Presence:** Required. **Type:** Text. **Format:** Free-form, restricted to 100 characters.

PARAMETER	DESCRIPTION
HTTP parameter: **itemDescription1, itemDescription2, etc..** XML element: <checkout> <items> <item> **<description>**	**Item descriptions.** Describes the items being paid for. The description is the text that PagSeguro displays on each line-item when the buyer is finalizing the payment, therefore, it is important for it to be clear and explanatory. **Presence:** Required. **Type:** Text. **Format:** Free-form, restricted to 100 characters.
HTTP parameter: **itemAmount1, itemAmount2, etc.** XML element: <checkout> <items> <item> **<amount>**	**Unit values of items.** Represent the unit prices for each item being paid for. Besides being able to contain multiple items, the payment data may also contain several units of the same item. This parameter represents the value of a single item unit, which will be multiplied by the quantity to obtain the total value of the line-item. **Presence:** Required. **Type:** Number. **Format:** Decimal with two decimal places separated by a period (e.g., 1234.56), greater than 0.00 and less than or equal to 9999999.00.
HTTP parameter: **itemQuantity1, itemQuantity2, etc.** XML element: <checkout> <items> <item> **<quantity>**	**Quantity of items.** Represents the quantity of each item being paid for. Besides being able to contain multiple items, the payment may also contain several units of the same item. This parameter represents the number of units being purchased, which will be multiplied by the unit value to obtain the total value of this line-item. **Presence:** Required. **Type:** Number. **Format:** A whole number greater than or equal to 1 and less than or equal to 999.
HTTP Parameter: **itemShippingCost1, itemShippingCost2, etc.** XML Element: <checkout> <items> <item> **<shippingCost>**	**Custos de frete dos itens.** Represents the cost of freight for each line item. If this cost is specified PagSeguro will assume that this is the cost of shipping for this item and will not make any calculations based on weight. **Presence:** Optional. **Type:** Number. **Format:** Decimal, with two decimal places separated by a period (e.g., 1234.56), greater than 0.00 and less than or equal to 9999999.00.

PARAMETER	DESCRIPTION
HTTP Parameter: **itemWeight1, itemWeight2, etc.** XML Element: <checkout> <items> <item> **<weight>**	**Item weight.** The weight in grams of each item being purchased. PagSeguro uses the weight of the item to perform a calculation of the shipping cost according to the postal service, unless the shipping cost of the item is already specified directly. See the Shipping Calculation Rules section for more information. **Presence:** Optional. **Type:** Number. **Format:** A whole number corresponding to the weight in grams of the item. The sum of the weight of all products cannot exceed 30000 g (30 kg).
HTTP Parameter: **reference** XML Element: <checkout> **<reference>**	**Reference code.** Defines a reference code for the payment. This code is linked to the transaction that is created and is useful to link the PagSeguro transactions with a registered sale in your system. **Presence:** Optional. **Type:** Text. **Format:** Free-form restricted to 200 characters.
XML Element: <checkout> **<sender>**	**Buyer data.**
HTTP Parameter: **senderEmail** XML Element: <checkout> <sender> **<email>**	**E-mail of buyer.** Specifies the email address of the buyer who is making the payment. This field is optional and can be sent if the buyer’s data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro. **Presence:** Optional. **Type:** Text. **Format:** Valid email (e.g., user@site.com.br), with no more than 60 characters.
HTTP Parameter: **senderName** XML Element: <checkout> <sender> **<name>**	**Full name of buyer.** Specifies the full name of the buyer who is making the payment. This field is optional and can be sent if the buyer’s data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro.

PARAMETER	DESCRIPTION
	Presence: Optional. **Type:** Text. **Format:** At least two sequences of characters , with a total limit of 50 characters.
XML Element: <checkout> <sender> **<phone>**	**Buyer's telephone number data.**
HTTP Parameter: **senderAreaCode** XML Element: <checkout> <sender> <phone> **<areaCode>**	**Area code of buyer.** Specifies the area code (DDD) of the buyer who is making the payment. This field is optional and can be sent if the buyer's data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro. **Presence:** Optional. **Type:** Number. **Format:** A two-digit number corresponding to a valid area code.
HTTP Parameter: **senderPhone** XML Element: <checkout> <sender> <phone> **<number>**	**Telephone number of buyer.** Specifies the telephone number of the buyer who is making the payment. This field is optional and can be sent if the buyer's data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro. **Presence:** Optional. **Type:** Number. **Format:** A number with 7 to 9 digits.
XML Element: <checkout> <sender> **<documents>**	**List of documents of buyer.**
XML Element: <checkout> <sender> <documents> **<document>**	**Represents a document of buyer.**
XML Element: <checkout> <sender> <documents> <document>	**Type of document of buyer.** At this point only the CPF document type is supported. Specifies the type of document of the buyer who is making the payment. This field is optional and can be sent if the buyer's data

<table>
<tr><th>PARAMETER</th><th>DESCRIPTION</th></tr>
<tr><td>**<type>**</td><td>has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro.

Presence: Optional.
Type: Text.
Format: Case sensitive. Only the value **CPF** is accepted.</td></tr>
<tr><td>HTTP Parameter:
senderCPF

XML Element:
<checkout>
<sender>
<documents>
<document>
<value></td><td>**Document number of buyer.**

Specifies the document number of the buyer who is making the payment. This field is optional and can be sent if the buyer's data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro.

Presence: Optional.
Type: Number.
Format: one 11-digit number.</td></tr>
<tr><td>HTTP Parameter:
senderBornDate

XML Element:
<checkout>
<sender>
<bornDate></td><td>**Birth date of buyer.**

Specifies the birth date of the buyer who is making the payment. This field is optional and can be sent if the buyer's data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro.

Presence: Optional.
Type: Data.
Format:dd/MM/yyyy (day/month/year).</td></tr>
<tr><td>XML Element:
<checkout>
<shipping></td><td>**Shipping data.**</td></tr>
<tr><td>HTTP Parameter:
shippingType

XML Element:
<checkout>
<shipping>
<type></td><td>**Shipping type.**

The shipping method to be used to send the product. This information is used by PagSeguro to calculate, according to the postal service, the shipping cost based on the weight of the items. The table below describes the accepted types and their meanings:

<table>
<tr><th>Code</th><th>Meaning</th></tr>
<tr><td>**1**</td><td>Normal mail (PAC).</td></tr>
<tr><td>**2**</td><td>SEDEX special mail</td></tr>
</table></td></tr>
</table>

PARAMETER	DESCRIPTION
	3 Unspecified delivery method. **Presence:** Optional. **Type:** Number. **Format:** One of the codes in the table above.
HTTP Parameter: **shippingCost** XML Element: <checkout> <shipping> **<cost>**	**Total shipping cost.** Specifies the total shipping cost of the order. If this value is specified, PagSeguro will use this as the shipping cost and will not make any calculations based on the weight and value of the delivery items. **Presence:** Optional. **Type:** Number. **Format:** Decimal with two decimal places separated by a period (e.g., 1234.56), greater than 0.00 and less than or equal to 9999999.00.
XML Element: <checkout> <shipping> **<address>**	**Shipping address data.**
HTTP Parameter: **shippingAddressCountry** XML Element: <checkout> <shipping> <address> **<country>**	**Shipping address country.** Specifies the country part of the shipping address. This field is optional and can be sent if the buyer's data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro. **Presence:** Optional. **Type:** Text. **Format:** Currently, only **BRA** is allowed.
HTTP Parameter: **shippingAddressState** XML Element: <checkout> <shipping> <address> **<state>**	**Shipping address state.** Specifies the state part of the shipping address. This field is optional and can be sent if the buyer's data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro. **Presence:** Optional. **Type:** Text. **Format:** Two letters, upper case, representing the abbreviation of the corresponding Brazilian state such as SP, RJ, MG, BA, etc.

PARAMETER	DESCRIPTION
HTTP Parameter: **shippingAddressCity** XML Element: <checkout> <shipping> <address> **<city>**	**Shipping address city.** Specifies the city part of the shipping address. This field is optional and can be sent if the buyer's data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro. **Presence:** Optional. **Type:** Text. **Format:** Free-form. Must be a valid Brazilian city name, with at least 2 and no more than 60 characters.
HTTP Parameter: **shippingAddressPostalCode** XML Element: **<checkout>** <shipping> <address> **<postalCode>**	**Shipping address postal code.** Specifies the postal code part of the shipping address. This field is optional and can be sent if the buyer's data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro. Currently only Brazilian postal codes (CEP) are supported. **Presence:** Optional. **Type:** Number. **Format:** 8 digit number.
HTTP Parameter: **shippingAddressDistrict** XML Element: <checkout> <shipping> <address> **<district>**	**Shipping address district.** Specifies the district or neighborhood part of the shipping address. This field is optional and can be sent if the buyer's data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro. **Presence:** Optional. **Type:** Text. **Format:** Free-form, restricted to 60 characters.
HTTP Parameter: **shippingAddressStreet** XML Element: <checkout> <shipping> <address> **<street>**	**Shipping address street.** Specifies the street part of the shipping address. This field is optional and can be sent if the buyer's data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro. **Presence:** Optional. **Type:** Text. **Format:** Free-form, restricted to 80 characters.
HTTP Parameter: **shippingAddressNumber**	**Shipping address street number.**

PARAMETER	DESCRIPTION
XML Element: <checkout> <shipping> <address> **<number>**	Specifies the street number of the shipping address. This field is optional and can be sent if the buyer's data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro. **Presence:** Optional. **Type:** Text. **Format:** Free-form, restricted to 20 characters.
HTTP Parameter: **shippingAddressComplement** XML Element: <checkout> <shipping> <address> **<complement>**	**Shipping address compliment.** Specifies the complement (block, apartment, etc.) part of the shipping address. This field is optional and can be sent if the buyer's data has already been inserted into your system and you wish to prevent him/her from filling out this information again in PagSeguro. **Presence:** Optional. **Type:** Text. **Format:** Free-form, restricted to 40 characters.
HTTP Parameter: **extraAmount** XML Element: <checkout> **<extraAmount>**	**Extra value.** Specifies an extra value to be added or subtracted from the total payment. This value may represent an additional payment to be charged or a discount to be applied if the value is negative. **Presence:** Optional. **Type:** Number. **Format:** Decimal (positive or negative), with two decimal places separated by a period (e.g., 1234.56, -1234.56), greater than or equal to -9999999.00 and less than or equal to 9999999.00. When negative, this value cannot be greater than or equal to the sum of the products.
HTTP Parameter: **redirectURL** XML Element: <checkout> **<redirectURL>**	**Redirect URL after payment.** Defines the URL to which the buyer will be redirected after the end of the payment flow. This parameter may include a specific address for every payment made. See more in Redirecting the buyer to a dynamic address. **Presence:** Optional. **Type:** Text. **Format:** A valid URL, restricted to 255 characters.
HTTP Parameter: **notificationURL**	**URL to send payment notifications.**

<table>
<tr><th>PARAMETER</th><th>DESCRIPTION</th></tr>
<tr><td>XML Element:
<checkout>
<b>&lt;notificationURL&gt;</b></td><td>Defines the URL to which PagSeguro will send the notification codes related to the payment. Whenever there is a change in the status of the transaction and that requires your attention, a new notification will be sent to this address.
<b>Presence:</b> Optional.
<b>Type:</b> Text.
<b>Format:</b> A valid URL, restricted to 255 characters.</td></tr>
<tr><td>HTTP Parameter:
<b>maxUses</b>
XML Element:
<checkout>
<b>&lt;maxUses&gt;</b></td><td><b>Maximum number of uses for payment code.</b>
Defines the maximum number of times the payment code created by the Payment API call can be used. This parameter can be used as a safety measure.
<b>Presence:</b> Optional.
<b>Type:</b> Number.
<b>Format:</b> A whole number greater than 0 and less or equal to 999.</td></tr>
<tr><td>HTTP Parameter:
<b>maxAge</b>
XML Element:
<checkout>
<b>&lt;maxAge&gt;</b></td><td><b>Validity of payment code.</b>
Defines the time (in seconds) during which the payment code created by the Payment API call can be used. This parameter can be used as a safety measure.
<b>Presence:</b> Optional.
<b>Type:</b> Number.
<b>Format:</b> A whole number greater or equal to 30 and less than or equal to 999999999.</td></tr>
<tr><td>HTTP Parameter:
<b>metadataItemKey1, metadataItemKey2, etc.</b>
XML Element:
<checkout>
<metadata>
<item&gt;
<b>&lt;key&gt;</b></td><td><b>Metadata of transaction.</b>
Allows extra information, grouped or not, to be added to your payment request.
<b>Presence:</b> Optional.
<b>Type:</b> Text.
<b>Format:</b> Only the values described below are accepted.
<b>Note:</b> Using HTTP there is a restriction of up to 100 keys per post.
<table>
<tr><th>Value</th><th>Format</th><th>Description</th></tr>
<tr><td>PASSENGER_CPF</td><td>[0-9]{11}</td><td>CPF taxpayer ID no. of passenger</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>PARAMETER</th><th>DESCRIPTION</th></tr>
<tr><td></td><td>
<table>
<tr><td>PASSENGER_PASSPORT</td><td>.+</td><td>Passport of passenger</td></tr>
<tr><td>ORIGIN_CITY</td><td>.+</td><td>City of origin</td></tr>
<tr><td>DESTINATION_CITY</td><td>.+</td><td>City of destination</td></tr>
<tr><td>ORIGIN_AIRPORT_CODE</td><td>.+</td><td>Airport code of origin</td></tr>
<tr><td>DESTINATION_AIRPORT_CODE</td><td>.+</td><td>Airport code of destination</td></tr>
<tr><td>GAME_NAME</td><td>.+</td><td>Name of game</td></tr>
<tr><td>PLAYER_ID</td><td>.+</td><td>ID of player</td></tr>
<tr><td>TIME_IN_GAME_DAYS</td><td>[0-9]+</td><td>Time in game days</td></tr>
<tr><td>MOBILE_NUMBER</td><td>([0-9]{2})?([0-9]{2})([0-9]{4,5}[0-9]{4})</td><td>Mobile phone to be recharged</td></tr>
<tr><td>PASSENGER_NAME</td><td>.+</td><td>Name of passenger</td></tr>
</table>
</td></tr>
<tr><td>HTTP Parameter:
<b>metadataItemValue1, metadataItemValue2, etc.</b>

XML Element:
<checkout>
<metadata>
<item&gt;
<b>&lt;value&gt;</b></td><td><b>Meta data value.</b>

Meta data value for a metadata item.

<b>Presence:</b> Optional.
<b>Type:</b> Text.
<b>Format:</b> Free form. Restricted to 100 characters.
<b>Note:</b> Using HTTP there is a restriction of up to 100 values per post.</td></tr>
</table>

PARAMETER	DESCRIPTION
HTTP Parameter: **metadataItemGroup1, metadataItemGroup2, etc.** XML Element: <checkout> <metadata> <item> **<group>**	**Groups of meta data.** Provides a mechanism for grouping multiple meta data items, such as CPF and name of the same passenger, by specifying a group ID. Other items with the same value in this field would belong to the same meta data group. **Presence:** Optional. **Type:** Number. **Format:** A whole number larger than 0. **Note:** Using HTTP there is a restriction of up to 100 groups per post.

Response of Payment API

The response from the Payment API is in XML. The example below shows a successful response to a Payment API call.

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<checkout>
    <code>8CF4BE7DCECEF0F004A6DFA0A8243412</code>
    <date>2010-12-02T10:11:28.000-02:00</date>
</checkout>
```

The table below describes the elements present in a success response from the Payment API.

PARAMETER	DESCRIPTION
<checkout>	Root of the XML response.
<checkout> **<code>**	**Identification code of payment.** This code must be used to redirect the buyer into the payment flow **Type:** Text. **Format:** A string of 32 characters.
<checkout> **<date>**	**Creation date of payment code.** **Type:** Date / time. **Format: YYYY-MM-DDThh:mm:ss.sTZD**, the official W3C format for dates. See more about formatting dates.

If an error occurs on a Payment API call, an error response will be returned as follows. It will contain all errors that occurred on the call.

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<errors>
    <error>
        <code>11004</code>
        <message>Currency is required.</message>
    </error>
    <error>
        <code>11005</code>
        <message>Currency invalid value: 100</message>
    </error>
</errors>
```

The table below describes the elements present in an error response from the Payment API.

ELEMENT	DESCRIPTION
<errors>	Root of error response XML. Contains a list of errors found.
<errors> **<error>**	Describes a single error.
<errors> <error> **<code>**	**Error code.** Identifies the error so the system can handle the condition accordingly. **Type:** Number. **Format:** See the error table of the Payment API.
<errors> <error> **<message>**	**Description of the error.** Provides a description of the error encountered. Can be useful for debugging. **Type:** Text. **Format:** See the error table of the Payment API.

Directing the buyer to the payment flow

After making a successful call to the Payment API, you must redirect the buyer to the payment flow using the returned payment code. The example below shows a fully constructed URL that will initiate a payment flow for the user.

```
https://pagseguro.sandbox.uol.com.br/v2/checkout/payment.html?code=8CF4BE7DCECEF0F004
A6DFA0A8243412
```

Note that the base URL of the payment flow is the same as that used by the HTML form on the Payment page. It takes a single parameter described below.

PARAMETER	DESCRIPTION
code	**Payment code returned by the call to the payment API of PagSeguro.** **Presence:** Required. **Type:** Text. **Format:** A string of 32 characters.