



Standard eCommerce Documentation

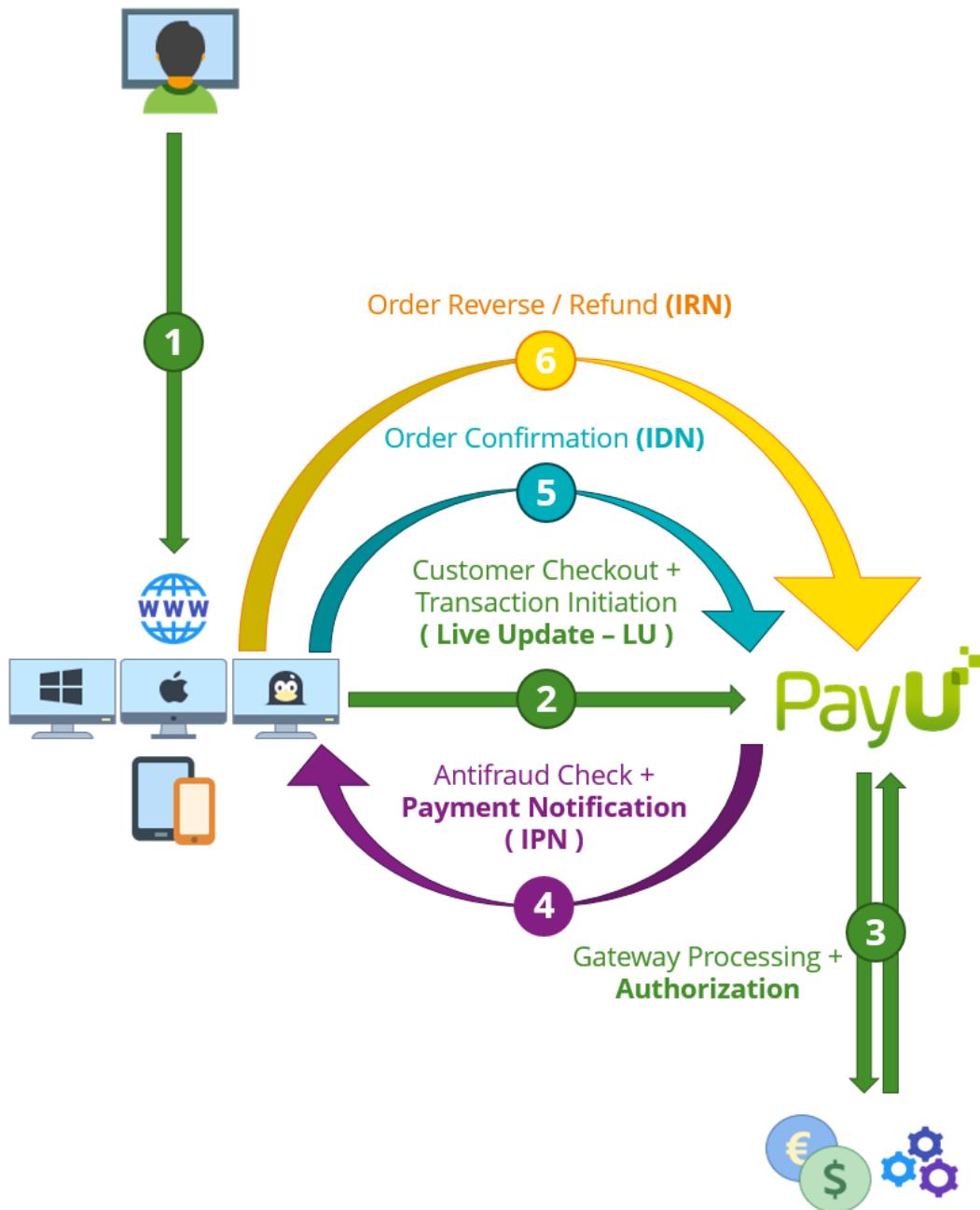
Last Update: 17th of April 2019



Contents

1. Live Update Protocol (LU)	5
How does it work?.....	5
Live Update Data.....	5
General Data.....	5
Order Details (Shopping Cart Contents).....	5
Customer Information.....	7
Additional Info.....	8
Authenticating the transaction (HMAC MD5 Signature).....	9
Going further - Including Billing and Delivery Information	12
Tips and Tricks	13
Advanced - Redirect to merchant website	13
Test Orders	13
Error Messages.....	14
2. Instant Payment Notification Protocol (IPN)	15
How does it work?.....	15
Notes:.....	15
The IPN Format.....	16
General Data.....	16
Billing Data	17
Delivery Information	17
Ordered Products.....	18
Other Details	19
How do I validate receiving the notification?	19
How to calculate the signature (HASH) on a data series.....	20
3. Instant Delivery Notification Protocol (IDN)	22
How does it work?.....	22
Notes:.....	22
What data needs to be sent to PayU?.....	23
How to build the HASH signature?.....	23
How do I receive the confirmation from PayU?.....	24
How is the response HMAC MD5 signature calculated?.....	24
4. Instant Reverse/Refund Notification Protocol (IRN)	26

How does it work?.....	26
What is the data that must be sent to PayU?.....	26
What is the difference between REVERSE and REFUND?.....	27
How is the data structured to be sent to PayU?.....	27
How do I receive the PayU response?.....	28
What response do I receive from PayU?.....	28
5. Instant Order Search Protocol (IOS).....	29
Authentication	29
Answer	29



1. Live Update Protocol (LU)

How does it work?

For each customer that clicks the "Checkout" button in the merchant's website or mobile application, an HTTP POST must be initiated to <https://secure.payu.ro/order/lu.php> for the transaction to get started.

The customer is then redirected to the PayU payment page, where he must enter the data required to authorize the online payment.

Live Update Data

The Live Update API is product-based, meaning that any order must contain at least one product, **uniquely identified** in the merchant's account through its product code. The PayU server expects the following information:

General Data

PARAMETER	DESCRIPTION	Required
MERCHANT	Your Merchant ID, available in cPanel (Account management / Account settings – APIs Settings box).	Yes
ORDER_REF	The order number from your website or application (for easier order identification/reconciliation).	Yes
ORDER_DATE	The date when the order is initiated in the system, in YYYY-MM-DD HH:MM:SS format (ex: "2018-05-01 21:15:45").	Yes

Order Details (Shopping Cart Contents)

PARAMETER	DESCRIPTION	Required
ORDER_PNAME[]	Array with the product names (maximum length: 155 characters per product name).	Yes
ORDER_PCODE[]	Array containing unique identification / product codes associated with each product available on your website or in your application. If multiple products are sent (in the same or subsequent transactions) with the same product code, PayU will update the product with the corresponding ORDER_PCODE, overwriting all the other product information – name, price, taxes. Assigning the same product code to multiple products will lead to errors or payments processed at a wrong value.	Yes

ORDER_PINFO[]	Array with additional product information.	No
ORDER_PRICE[]	Array with the product prices, positive number, with "." as a decimal separator. Prices must include VAT and cannot be equal to zero.	Yes
ORDER_QTY[]	Array with the quantities for each product (value should be a positive integer).	Yes
ORDER_VAT[]	Array with VAT percentage values for each product in the order (value should be a positive integer). Ex: 0, 9, 19.	Yes
ORDER_PRICE_TYPE[]	Array that specifies that the ORDER_PRICE[] includes VAT. Value is " GROSS " (case sensitive). The parameter is mandatory. Not specifying it will lead to errors or payments processed at a wrong value.	Yes
PRICES_CURRENCY	The currency in which the prices, taxes costs, shipping fees and discounts are expressed. Ex: RON, EUR, USD. <i>*To transact in a different currency than the one in which the prices are specified, use the CURRENCY parameter.</i>	Yes
ORDER_SHIPPING	Shipping costs for the order, positive number, with ":" as a decimal separator (optional).	No
DISCOUNT	The discount value for the order, positive number, with "." as a decimal separator (optional).	No
PAY_METHOD	<p>The payment method for the transaction. If the parameter is specified, the customer will not be able to change its value in the PayU payment pages. If the parameter is not specified, the payment methods active* on the account will be displayed.</p> <p><i>*If you want to activate some of the payment methods presented below please contact your account manager.</i></p> <p>Possible values</p> <ul style="list-style-type: none"> • Card payments: <ul style="list-style-type: none"> ◦ CCVISAMC - VISA/MasterCard. • Masterpass payments: <ul style="list-style-type: none"> ◦ MASTERPASS - by MasterCard. • Installments option: <ul style="list-style-type: none"> ◦ ALPHABANK_INSTALLMENTS – payments with Alpha Banks installments cards; ◦ BCR_INSTALLMENTS – payments with Banca Comerciala Romana installment cards; 	Yes

	<ul style="list-style-type: none"> ○ BRD_INSTALLMENTS – payments with BRD installment cards; ○ BRDF – payments with BRD Finance installment cards; ○ CARD_AVANTAJ – payments with Credit Europe's Card Avantaj installment cards; ○ GARANTI_RO – payments with Garanti Bank installment cards; ○ OPTIMO – payments with Credit Europe's Optimo installment cards; ○ STAR_BT – payments with Banca Transilvania Star BT installment cards. ○ ING_INSTALLMENTS – payments with ING Credit Card installments cards. ● Internet banking (iTransfer): <ul style="list-style-type: none"> ○ ITRANSFER_BCR – Internet banking payment with Click 24 Banking BCR; ○ ITRANSFER_BT – iTransfer via BT24. ● Other: <ul style="list-style-type: none"> ○ ZEBRA_PAY – cash collection at Zebra Pay Self Service Terminals; ○ PAYPAL – PayPal Payments; ○ WIRE – Bank Wire 	
ORDER_HASH	HMAC_MDS signature for the sent data (HMAC is defined in RFC 2104).	Yes

Customer Information

PARAMETER	DESCRIPTION	Required
BILL_FNAME	Customer's first name	Yes
BILL_LNAME	Customer's last name	Yes
BILL_EMAIL	Customer's email address	Yes
BILL_PHONE	Phone number	Yes
BILL_COUNTRYCODE	2 letter country code (ex: RO for Romania)	Yes
BILL_COMPANY	Legal company name for billing. <i>* Required only for company orders.</i>	Yes
BILL_FISCALCODE	Company's Fiscal Code (CUI/VAT ID). <i>* Required only for company orders.</i>	Yes
BILL_REGNUMBER	Company's Registration Number at the Commerce Registry. <i>* Required only for company orders.</i>	Yes

Additional Info

PARAMETER	DESCRIPTION	Required
LANGUAGE	<p>Allows setting a specific language for the payment interface and email notifications sent by PayU to your customers.</p> <p>Possible values for all PayU payment pages:</p> <ul style="list-style-type: none"> • RO – Romanian • EN – English • HU – Hungarian • BG – Bulgarian <p>Possible values available only in the old PayU payment page:</p> <ul style="list-style-type: none"> • DE – German • FR – French • IT – Italian • ES – Spanish • PL – Polish 	Yes
BACK_REF	<p>URL used to redirect the customer back to your website or application after the payment has been successfully authorized by the issuing bank.</p> <p>The redirect will not be made for failed payments.</p>	No
TESTORDER	<p>Optional parameter, used to initiate transactions in TEST MODE.</p> <p>Possible values: "TRUE" or "FALSE" (case sensitive).</p> <p>If the parameter is active, the PayU payment form will be pre-filled with test payment details (you don't need any credit card test numbers).</p>	No
AUTOMODE	<p>Boolean parameter ("0" or "1") used to automatically redirect the user to the final step of the payment process.</p> <p>This only works with the old PayU payment page, if all the required information has been correctly submitted in the Live Update protocol.</p>	No
ORDER_TIMEOUT	<p>Number of seconds in which the card details can be filled into the PayU page. Value must be a positive integer number.</p>	No
TIMEOUT_URL	<p>Sets the URL for the redirect of the customer, in case the ORDER_TIMEOUT expired (optional).</p>	No

Authenticating the transaction (HMAC MD5 Signature)

Let's construct a Live Update request, for the following transaction information:

Information	Parameters and values	Value Byte Length
General Data	MERCHANT = PAYUDEMO	8
	ORDER_REF = 789456123	9
	ORDER_DATE = 2016-04-10 17:30:56	19
Cart Contents	ORDER_PNAME[] = CD Player	9
	ORDERPCODE[] = SKU_04891	9
	ORDER_PINFO[] = Extended Warranty - 5 Years	27
	ORDER_PRICE[] = 82.3	4
	ORDER_PRICE_TYPE[] = GROSS	5
	ORDER_QTY[] = 7	1
	ORDER_VAT[] = 19	2
	ORDER_PNAME[] = Mobile Phone	12
	ORDERPCODE[] = SKU_07409	9
	ORDER_PINFO[] = Dual SIM	8
	ORDER_PRICE[] = 1945.75	7
	ORDER_PRICE_TYPE[] = GROSS	5
	ORDER_QTY[] = 3	1
	ORDER_VAT[] = 19	2
Customer Information	ORDER_PNAME[] = Laptop	6
	ORDERPCODE[] = SKU_04965	9
	ORDER_PINFO[] = 17" Display	11
	ORDER_PRICE[] = 5230	4
	ORDER_PRICE_TYPE[] = GROSS	5
Company Information	ORDER_QTY[] = 1	1
	ORDER_VAT[] = 19	2
	PRICES_CURRENCY = RON	3
	ORDER_SHIPPING =	0
	DISCOUNT = 55	2
Additional Info	PAY_METHOD = CCVISA/MC	8
	BILL_FNAME = Average	n/a
Customer Information	BILL_LNAME = Joes	n/a
	BILL_EMAIL = average.joes@email.com	n/a
	BILL_PHONE = 0700000000	n/a
	BILL_COUNTRYCODE = RO	n/a
	BILL_COMPANY = SC COMPANY SRL	n/a
Company Information	BILL_FISCALCODE = J40/1234	n/a
	BILL_REGNUMBER = R01234	n/a
Additional Info	TESTORDER = TRUE	4
	LANGUAGE = RO	n/a

*Parameters that have n/a in the value length field are not included in the string based on which the signature is calculated. More details below.

To validate this information when the order is placed, an HMAC MD5 signature is required, as a value for the ORDER_HASH parameter. The calculation is made using the Secret Key of the merchant account, available in your PayU cPanel, by accessing [Account Management/Account Settings – APIs Settings](#) section.

The signature is calculated based on a **string**, composed of each of the **values** above, with their corresponding **byte lengths** prepended (in UTF-8 bytes). The parameters that compose the string are the following, in this exact same order:

- MERCHANT
- ORDER_REF
- ORDER_DATE
- ORDER_PNAME[]
- ORDER_PCODE[]
- ORDER_PINFO[]
- ORDER_PRICE[]
- ORDER_QTY[]
- ORDER_VAT[]
- ORDER_SHIPPING
- PRICES_CURRENCY
- DISCOUNT
- PAY_METHOD
- ORDER_PRICE_TYPE[]
- TESTORDER

Given the Live Update request containing the parameters from the above table, the resulting string will be:

8PAYUDEMO9789456123192016-04-10 17:30:569CD Player12Mobile
Phone6Laptop9SKU_048919SKU_074099SKU_0496527Extended Warranty - 5 Years8Dual
SIM1117"Display482.371945.75452301111121921921903RON2558CCVISAMC5GROSS5GROSS5GRO
SS4TRUE

Note! For the order shipping, there was no value for the ORDER_SHIPPING parameter, but because the parameter was specified in the request form, we must consider its byte length (0) in the string composition. This applies to any parameter that must be included in the hash and is included in the request without any value.

The data must be transported in UTF-8 format, to correctly calculate the byte lengths on both ends (your server and the PayU platform).

If the Secret Key for the PayU account with Merchant ID: PAYUDEMO is **1231234567890123**, the resulting HMAC MD5 and value for the ORDER_HASH parameter is:
1f2393078b4d365083927e41fea507d9.

The resulting Live Update HTTP POST request is:

```
<form action="https://secure.payu.ro/order/lu.php" method="post">
    <input type="hidden" name="MERCHANT" id="MERCHANT" value="PAYUDEMO"/>
    <input type="hidden" name="ORDER_REF" id="ORDER_REF" value="789456123"/>
    <input type="hidden" name="ORDER_DATE" id="ORDER_DATE" value="2016-04-10 17:30:56"/>
    <input type="hidden" name="ORDER_PNAME[]" id="ORDER_PNAME[]" value="CD Player"/>
    <input type="hidden" name="ORDER_PCODE[]" id="ORDER_PCODE[]" value="SKU_04891"/>
    <input type="hidden" name="ORDER_PINFO[]" id="ORDER_PINFO[]" value="Extended Warranty - 5 Years"/>
    <input type="hidden" name="ORDER_PRICE[]" id="ORDER_PRICE[]" value="82.3"/>
    <input type="hidden" name="ORDER_PRICE_TYPE[]" id="ORDER_PRICE_TYPE[]" value="GROSS"/>
    <input type="hidden" name="ORDER_QTY[]" id="ORDER_QTY[]" value="1"/>
    <input type="hidden" name="ORDER_VAT[]" id="ORDER_VAT[]" value="19"/>
    <input type="hidden" name="ORDER_PNAME[]" id="ORDER_PNAME[]" value="Mobile Phone"/>
    <input type="hidden" name="ORDER_PCODE[]" id="ORDER_PCODE[]" value="SKU_07409"/>
    <input type="hidden" name="ORDER_PINFO[]" id="ORDER_PINFO[]" value="Dual SIM"/>
    <input type="hidden" name="ORDER_PRICE[]" id="ORDER_PRICE[]" value="1945.75"/>
    <input type="hidden" name="ORDER_PRICE_TYPE[]" id="ORDER_PRICE_TYPE[]" value="GROSS"/>
    <input type="hidden" name="ORDER_QTY[]" id="ORDER_QTY[]" value="1"/>
    <input type="hidden" name="ORDER_VAT[]" id="ORDER_VAT[]" value="19"/>
    <input type="hidden" name="ORDER_PNAME[]" id="ORDER_PNAME[]" value="Laptop"/>
    <input type="hidden" name="ORDER_PCODE[]" id="ORDER_PCODE[]" value="SKU_04965"/>
    <input type="hidden" name="ORDER_PINFO[]" id="ORDER_PINFO[]" value="17" Display"/>
    <input type="hidden" name="ORDER_PRICE[]" id="ORDER_PRICE[]" value="5230"/>
    <input type="hidden" name="ORDER_PRICE_TYPE[]" id="ORDER_PRICE_TYPE[]" value="GROSS"/>
    <input type="hidden" name="ORDER_QTY[]" id="ORDER_QTY[]" value="1"/>
    <input type="hidden" name="ORDER_VAT[]" id="ORDER_VAT[]" value="19"/>
    <input type="hidden" name="PRICES_CURRENCY" id="PRICES_CURRENCY" value="RON"/>
    <input type="hidden" name="ORDER_SHIPPING" id="ORDER_SHIPPING" value=""/>
    <input type="hidden" name="DISCOUNT" id="DISCOUNT" value="55"/>
    <input type="hidden" name="LANGUAGE" id="LANGUAGE" value="RO"/>
    <input type="hidden" name="PAY_METHOD" id="PAY_METHOD" value="CCVISAMC"/>
    <input type="hidden" name="TESTORDER" id="TESTORDER" value="TRUE"/>
    <input type="hidden" name="BILL_FNAME" id="BILL_FNAME" value="Average"/>
    <input type="hidden" name="BILL_LNAME" id="BILL_LNAME" value="Joes"/>
    <input type="hidden" name="BILL_COMPANY" id="BILL_COMPANY" value="SC COMPANY SRL"/>
    <input type="hidden" name="BILL_FISCALCODE" id="BILL_FISCALCODE" value="J40/1234"/>
    <input type="hidden" name="BILL_REGNUMBER" id="BILL_REGNUMBER" value="RO1234"/>
    <input type="hidden" name="BILL_EMAIL" id="BILL_EMAIL" value="average.joes@email.com"/>
    <input type="hidden" name="BILL_PHONE" id="BILL_PHONE" value="0700000000"/>
    <input type="hidden" name="BILL_COUNTRYCODE" id="BILL_COUNTRYCODE" value="RO"/>
    <input type="hidden" name="ORDER_HASH" id="ORDER_HASH" value="lf2393078b4d365083927e4fea507d9"/>
    <input type="submit" value="PAYU LiveUpdate"/>
</form>
```

Going further – Including Billing and Delivery Information

You have the opportunity of including the billing and delivery information in your Live Update requests. By sending full billing and delivery details, you can help PayU provide the best Anti-Fraud filtering possible for your orders.

The billing and delivery parameters are optional and are not included in the HMAC MD5 hash calculation detailed above.

PARAMETER	DESCRIPTION	Required
BILL_CISERIAL	ID Card Series (for RO residents)	No
BILL_CINUMBER	ID Card Number (for RO residents)	No
BILL_CISSLER	ID Card Issuer (for RO residents)	No
BILL_CNP	Numeric Personal Code (for RO residents)	No
BILL_BANK	Company's bank – only for company orders.	No
BILL_BANKACCOUNT	Company's bank account – only for company orders.	No
BILL_FAX	Fax number	No
BILL_ADDRESS	Customer's/Company's address	No
BILL_ADDRESS2	Customer's/Company's address (additional)	No
BILL_ZIPCODE	Customer's/Company's ZIP/Postal Code	No
BILL_CITY	City	No
BILL_STATE	State/County	No
DELIVERY_FNAME	Customer's first name	No
DELIVERY_LNAME	Customer's last name	No
DELIVERY_COMPANY	Company Legal Name	No
DELIVERY_PHONE	Phone number	No
DELIVERY_ADDRESS	Customer's/Company's Address	No
DELIVERY_ADDRESS2	Customer's/Company's Address (additional)	No
DELIVERY_ZIPCODE	Customer's/Company's ZIP/Postal Code	No
DELIVERY_CITY	City	No
DELIVERY_STATE	State/County	No
DELIVERY_COUNTRYCODE	Country Code (RO for Romania)	No

Tips and Tricks

How to: Charging in another currency than the one in which prices are specified.

If you have more than one currency activated for your account, the customers can choose in the payment form the currency in which they will make the payment.

If you want to force the order's collecting using another currency than the one in which the prices are set (PRICES_CURRENCY), you can send the optional CURRENCY parameter:

```
<input name="CURRENCY" value="USD" type="hidden" />
```

Advanced - Redirect to merchant website

To redirect the customer back to the website (his account or a customized, dynamic thank you page), you can use the **BACK_REF** parameter with an URL. The **BACK_REF** redirect is, by default, made only if the used payment method is one with *instant authorization* (e.g. will be made for CCVISAMC, but not for regular WIRE transfer) and if the payment has been successful.

You can use GET parameters to specify an order or customer identifier, as below:

```
<input name="BACK_REF" value="http://domain.com/process.php?order=123456" type="hidden">
```

To make sure that the redirect comes from PayU, a control variable is attached to the URL to which the redirect is made. The control variable (a GET parameter, **ctrl**) is also HMAC MD5, calculated on a string composed from the **URL (with all parameters)** to which PayU redirects, with the **byte length** of that URL (parameters included) **prepended**.

Sample:

<http://www.yourdomain.com/process.php?order=123456&ctrl=741fcf35a297e256f4090c4ddf0ed65>

The source string for the ctrl HMAC MD5 calculation:

50<http://www.yourdomain.com/process.php?order=123456>

Test Orders

The purpose of placing a test order is to verify and validate the integration between your website and the PayU platform.

On production accounts, you will be able to generate an order that will have the TEST status using the TESTORDER = TRUE parameter in your Live Update request, as detailed above.

On test accounts, any order placed by you (regardless of the presence of the TESTORDER parameter, in the LU request) will be automatically marked by the PayU system with the TEST status.

By generating test orders, you will be able to verify:

- If the Live Update request is done according to this document;
- How the transaction details (customer information, shopping card contents, billing or delivery addresses) are displayed in the PayU Control Panel;

- If the PayU payment page template is loading properly (only available on accounts that use the old PayU payment page);
- If the new PayU payment page (Page V2) is triggered correctly.
- If the email notifications that PayU automatically sends to your customers are according to your requirements (the email templates can be personalized, upon request);

If the IPN has been implemented according to the documentation and if the algorithm is working properly, on your servers.

Error Messages

If you get errors when trying to transmit your data, see the list below for problem descriptions for each type of error:

Error	Description
ACCES DENIED	Your access to the PayU interface is not allowed. Please contact the PayU support team.
Invalid account	The MERCHANT parameter is incorrect or not specified.
Access not permitted	You access to the Live Update feature is restricted. You should contact your PayU Account Manager.
Invalid Data	The data you have transmitted is not correctly formed. Please check the arrays.
Invalid product code	The ORDER_PCODE[] array is incorrectly formed.
Invalid product name	The ORDER_PNAME[] array is incorrectly formed.
Invalid product group	The ORDER_PGROUP[] array is incorrectly formed.
Invalid Price	The ORDER_PRICE[] array is incorrectly formed.
Invalid VAT	The ORDER_VAT[] array is incorrectly formed.
Invalid price	The calculated total is incorrect. Check the DISCOUNT parameters.
Invalid Signature	The HMAC_MD5 signature is incorrectly calculated for the sent data.

2. Instant Payment Notification Protocol (IPN)

The IPN makes possible the automated processing of each authorized order in the online payment system, being a link between the PayU servers and your servers. This notification method will allow the retrieval of transaction data to be processed in your own order management system, with the purpose of helping you update the order status into your own platform.

How does it work?

After an order gets authorized by the issuing bank and Anti-Fraud approved by PayU system, the PayU server will send a data structure containing all the order related info to a preset URL (endpoint) belonging to your system. The data is sent through HTTP POST and it will also contain a signature (hash) to validate the information. The signature is obtained by applying an HMAC_MDS function with a common PayU/Vendor key, on the entire dataset (HMAC is defined at RFC 2104).

Notes:

- The URL (endpoint) where you want PayU to send notifications must be configured in your PayU cPanel account, by accessing [Account Management / Account Settings / IPN Settings](#). An HTTP GET request will be placed by PayU when saving your URL, to validate the server-to-server communication.
- In case there is no confirmation on the correct receival of the sent data, the PayU server will keep retrying to sending the IPN, until a valid response is obtained. The amount of time between each retry grows exponentially and the PayU system will stop retrying to send IPNs for the same order when it has reached the maximum of 50 retries.
- The IPN communication is independent of your customer's active connection and cannot be linked in any way to his session.
- You will have to add to your firewall / whitelist the following IP addresses belonging to the PayU servers, to ensure a proper communication between PayU and your own server:
 - 185.68.12.10
 - 185.68.12.11
 - 185.68.12.12
 - 185.68.12.26
 - 185.68.12.27
 - 185.68.12.28
- To study or test IPN notifications, we recommend placing a TEST order (using the TESTORDER = TRUE parameter, as detailed [above](#)). Test orders can be placed on both production or test accounts, opened in the PayU platform.
- You should **always check the ORDERSTATUS** (more details below) when receiving IPNs from PayU, and always make sure that Test orders will not be, by any means, confirmed or delivered like a real transaction should be (this is particularly applicable if your business involves online delivery of products, like software licenses, e-books, vouchers, prepay phone recharge codes, vignettes etc).

- You can resend an IPN notification as many times as you wish, using the following steps:
 - Find the order in the cPanel, at the [Orders & Reports](#) section, using the available filters;
 - Click on the order's reference number to open the pop-up with the order details;
 - In this pop-up, click on the "**Re-send notification**" link;
 - check the "**Debug IPN?**" option and click the button "Continue". This way, you can see both the contents of the IPN request, and the response of your script.

The IPN Format

The data sent via IPN to your server is described in the table below:

General Data

SALEDATE	The order placement date, in the following format: Y-m-d H:i:s (2016-04-26 14:59:35) .
REFNO	The PayU reference number for the order (maximum length: 9 characters).
REFNOEXT	The external reference - your own order number (maximum length: 100 characters), previously sent in the LU request as a value of the ORDER_REF parameter.
ORDERNO	The order count number (maximum length: 6 characters).
ORDERSTATUS	Current order status. Possible values: <ul style="list-style-type: none">• PAYMENTAUTHORIZED – card payment authorized & approved• PAYMENTRECEIVED – bank transfer payment collected• TEST – test order• COMPLETE – complete (finished) payment• REVERSED – reversed (cancelled) payment• REFUND – refunded payment (fully or partially)
PAYMETHOD	Customer's used payment method (maxim 40 characters). Ex: "Visa/MasterCard/Eurocard" or "Visa/MasterCard/Eurocard MasterPass".
PAYMETHOD_CODE	The code of the payment method used. Ex: "CCVISAMC" or "CCVISAMC MASTERPASS".

Billing Data

FIRSTNAME	First name (maximum length: 40 characters)
LASTNAME	Name (maximum length: 40 characters)
IDENTITY_NO	Customer ID Card series and number (Series/Number – available only for Romanian customers)
IDENTITY_ISSUER	IDENTITY_NO ID Card issuer authority
IDENTITY_CNP	Customer's personal numeric code, available only for Romanian customers.
COMPANY	Company name (maximum length: 40 characters)
REGISTRATIONNUMBER	Company's Commerce Registry registration number (maximum length: 40 characters)
FISCALCODE	Unique Registration Number / VAT ID (maximum length: 40 characters)
CBANKNAME	Company's Bank (maximum length: 40 characters)
CBANKACCOUNT	Company's Bank Account (maximum length: 50 characters)
ADDRESS1	Address (maximum length: 100 characters)
ADDRESS2	Additional Address info (maximum length: 100 characters)
CITY	City (maximum length: 30 characters)
STATE	State/Sector/County (maximum length: 30 characters)
ZIPCODE	ZIP/Postal Code (maximum length: 20 characters)
COUNTRY	Country (maximum length: 50 characters)
PHONE	Phone number (maximum length: 40 characters)
FAX	Fax number (maximum length: 40 characters)
CUSTOMEREMAIL	Customer's e-mail address (maximum length: 40 characters)

Delivery Information

FIRSTNAME_D	First name (maximum length: 40 characters)
LASTNAME_D	Last Name (maximum length: 40 characters)
COMPANY_D	Company (maximum length: 50 characters)
ADDRESS1_D	Address (maximum length: 100 characters)

ADDRESS2_D	Additional address info (maximum length: 100 characters)
CITY_D	City (maximum length: 30 characters)
STATE_D	State/Sector/County (maximum length: 30 characters)
ZIPCODE_D	ZIP/Postal Code (maximum length: 20 characters)
COUNTRY_D	Country (maximum length: 50 characters)
PHONE_D	Phone number (maximum length: 40 characters)
IPADDRESS	Client's IP Address (maximum length: 250 characters)

Ordered Products

CURRENCY	The currency in which the order has been processed. Possible values: RON, USD, EUR.
IPN_PID[]	Array with the ID Codes of the ordered products, in the PayU database (PayU reference)
IPN_PNAME[]	Array with product names
IPN_PCODE[]	Array with the product codes assigned by the vendor in the system (vendor reference)
IPN_INFO[]	Array with additional information sent for each ordered product (if they have been sent to PayU).
IPN_QTY[]	Array with the product quantities
IPN_PRICE[]	Array with unit prices per product (without VAT), in RON, with period/full-stop (.) as decimal place separator
IPN_VAT[]	Array with VAT values per product, with period "." as decimal place separator
IPN_VER[]	Array with product versions (maximum length: 50 characters)
IPN_DISCOUNT[]	Array with the amounts with which there has been made a discount in a promotion. Including VAT.
IPN_PROMONAME[]	Array with the names of the promotions in which the discounts specified above have been made.
IPN_DELIVEREDCODES[]	Array with the codes delivered to the clients, if the PayU contract contains this feature. Each element in the array is represented by a string, having comma (,) as a separator for each sent code, in case the ordered quantity is greater than 1.

IPN_TOTAL[]	Partial total on order line (including VAT), cu period/full-stop (.) as a decimal place separator
IPN_TOTALGENERAL	Total transaction amount, including VAT costs, with period/full-stop (.) as a decimal place separator
IPN_DATE	IPN POST's sending date in the following format: YmdHis (ex.: 20120426145935)
HASH	Request signature (MD5 HMAC on all the fields above)

Other Details

PAYMENTDATE	The order authorization date, in the following format: Y-m-d H:i:s (2012-04-26 15:01:25) .
COMPLETE_DATE	The order completion date, in the following format: Y-m-d H:i:s (2012-04-26 15:02:28) .
CARD_TYPE	Used credit card type. Ex: "Visa" or "MasterCard".
TOKEN_HASH	Value of the generated token. Used for Pay By Click or Pay On Time payment methods.
IPN_CC_MASK	Last four digits of the used credit card. Used for Pay By Click or Pay On Time payment methods. Ex: xxxx-xxxx-xxxx-1111 .
IPN_CC_EXP_DATE	Generated token's expiration date. Used for Pay By Click or Pay On Time payment methods. Ex: 2019-08-31 .

Note! If you require some of the parameters above and you notice that the IPN received from PayU does not contain them, please check your PayU cPanel ([Account Management / Account Settings / IPN Settings](#)) and verify if these parameters are active in your settings. You can also choose to deactivate some of these parameters, if you don't need to receive them from PayU.

How do I validate receiving the notification?

PayU expects an **answer inline** (anywhere in the page) in the following format:
 <EPAYMENT>DATE|HASH</EPAYMENT>

where:

DATE	The date of the answer return, in the YmdHis format (ex: 20120426145935)
HASH	Answer signature (MD5 HMAC on the initial fields IPN_PID[0], IPN_PNAME[0], IPN_DATE and DATE - previous field)

The HMAC calculation fields for the answer are:

IPN_PID[0]	Echo from the original IPN message – the ID of the first product.
IPN_PNAME[0]	Echo from the original IPN message – The name of the first product.
IPN_DATE	Echo from the original IPN message – The IPN date in the YmdHis format (20130101120001).
DATE	Date of the answer (your server's hour) in the YmdHis format (20130201120003).

How to calculate the signature (HASH) on a data series

Lets assume to have the following information:

Field name	Length	Field value
SALEDATE	19	2013-01-01 12:00:01
REFNO	7	1000037
REFNOEXT	0	
ORDERNO	2	13
ORDERSTATUS	8	AUTHRECEIVED
PAYMETHOD	8	CCVISAMC
FIRSTNAME	4	Test
LASTNAME	4	PayU
COMPANY	0	
REGISTRATIONNUMBER	0	
FISCALCODE	0	
CBANKNAME	0	
CBANKACCOUNT	0	
ADDRESS1	14	Some Street 21
ADDRESS2	0	
CITY	8	Bucharest
STATE	8	Bucharest
ZIPCODE	5	90210
COUNTRY	7	Romania

PHONE	11	0722.111.111
FAX	0	
CUSTOMEREMAIL	13	test@payu.com
FIRSTNAME_D	4	Test
LASTNAME_D	4	PayU
COMPANY_D	0	
ADDRESS1_D	14	Some Street 21
ADDRESS2_D	0	
CITY_D	6	Bucharest
STATE_D	6	Bucharest
ZIPCODE_D	5	90210
COUNTRY_D	6	Romania
PHONE_D	11	0268/121212
IPADDRESS	6	node11
CURRENCY	3	RON
IPN_PID[0]	1	123
IPN_PNAME[0]	25	Apple MacBook Air 13 inch
IPN_PCODE[0]	7	AMBA13I
IPN_INFO[0]	0	
IPN_QTY[0]	1	1
IPN_PRICE[0]	8	5000.00
IPN_VAT[0]	7	1200.00
IPN_VER[0]	0	
IPN_DISCOUNT[0]	4	0.00
IPN_PROMONAME[0]	0	
IPN_DELIVEREDCODES[0]	0	
IPN_TOTAL[0]	8	59500.00
IPN_TOTALGENERAL	8	6200.00

IPN_DATE	14	20130101120001
-----------------	----	----------------

The source string for the response is composed using the following data:

Field name	Length	Field value
IPN_PID[0]	3	123
IPN_PNAME[0]	25	Apple MacBook Air 13 inch
IPN_DATE	14	20130101120001
DATE	14	20130101120002

Resulting string:

312325Apple MacBook Air 13 inch**14**20130101120001**14**20130101120002

If the Secret Key for the PayU account with Merchant ID: PAYUDEMO is **1231234567890123**, the resulting HMAC MD5 and value for the ORDER_HASH parameter is:

2aad1b916c9c4c584ef1aa4a87d2c9ea

The HASH field characters can be lowercase/uppercase (hexadecimal string).

The response from the server to which the notification has been sent must be:

<EPAYMENT>20130101120002|2aad1b916c9c4c584ef1aa4a87d2c9ea</EPAYMENT>

In case the response is not valid, the notification is not confirmed. PayU will automatically resend the notification in a few minutes.

3. Instant Delivery Notification Protocol (IDN)

How does it work?

The IDN is a protocol that you will need to use for capturing (finishing) a payment that is *In Processing*, which means it has been authorized by the issuing bank and Anti-Fraud approved by PayU. To capture a payment, an HTTP POST request containing the identification data for that particular order must be sent from your administration system to the following PayU endpoint: <https://secure.payu.ro/order/idn.php> .

Notes:

- A separate HTTP POST needs to be sent for each order than needs to be captured by the PayU system.
- Each HTTP POST will be authenticated by using an HMAC_MD5 signature, based on the identification data contained in the POST, by using your PayU account's Merchant ID and Secret Key.

What data needs to be sent to PayU?

The identification data contained in the HTTP POST is described in the following table and must be sent in the following exact order:

Code	Description
MERCHANT	Your PayU Merchant ID
ORDER_REF	The PayU reference number for that payment (received via the IPN).
ORDER_AMOUNT	The total value of that payment.
ORDER_CURRENCY	The currency in which the order was made.
IDN_DATE	The date & time of the moment you're sending the IDN request, in YYYY-MM-DD HH:MM:SS format (ex: "2018-05-01 21:15:45").
ORDER_HASH	The requests signature. This signature is a HMAC_MD5 type signature built from all fields above (1-5).
REF_URL*	The URL address where the response will be sent with the GET method. The URL address must begin with the <> syntax. <i>* This field is optional. If this parameter is not sent or if it's empty, the response will be INLINE.</i>

How to build the HASH signature?

This chapter shows how the HASH signature is calculated.

Input data example:

Filed Name	Length	Field's Value
MERCHANT	8	PAYUDEMO
ORDER_REF	7	1000500
ORDER_AMOUNT	4	1645
ORDER_CURRENCY	3	EUR
IDN_DATE	19	2012-04-26 17:46:56

The source string for the MAC calculation is given by adding the string length at the beginning of the field. Giving the input data, the source string is:

8PAYUDEMO71000500416453EUR192012-04-26 17:46:56

The Secret Key for this example is: **1231234567890123**.

The final MD5 value will be: **f8108f20662200683117798598ee8e2d**

How do I receive the confirmation from PayU?

The PayU response is set to be INLINE (in the same page) and it has the following format (anywhere in the page):

```
<EPAYMENT>ORDER_REF|RESPONSE_CODE|RESPONSE_MSG|IDN_DATE|ORDER_HASH</EPAYMENT>
```

The meaning of the reply codes and their messages are as follows:

Response Code	Response Message
1	Confirmed.
2	ORDER_REF missing or incorrect.
3	ORDER_AMOUNT missing or incorrect.
4	ORDER_CURRENCY is missing or incorrect.
5	IDN_DATE is not in the correct format.
6	Error confirming order.
7	Order already confirmed.
8	Unknown error.
9	Invalid ORDER_REF.
10	Invalid ORDER_AMOUNT.
11	Invalid ORDER_CURRENCY.

Note! In case of an invalid reply from PayU, the order is not confirmed.

How is the response HMAC MD5 signature calculated?

The response HASH signature for the HTTP POST request is calculated by using the following data:

Filed Name	Length	Field's Value
ORDER_REF	7	1000500
RESPONSE_CODE	1	1
RESPONSE_MSG	9	Confirmed
IDN_DATE	19	2012-04-26 17:46:58

For more details on the information presented in this document, you can contact the team in charge of technical support at techsupport@payu.ro.



Results:

String:	71000500119Confirmed192012-04-27 17:46:58
MDS HASH Value:	6f8dfe9da81d6ea51e8f5d63341f4902

Note! The HASH fields can contain both lowercase and uppercase characters (hexadecimal string).

4. Instant Reverse/Refund Notification Protocol (IRN)

How does it work?

The IRN makes it possible for you to automate the sending of reverse/refund requests for orders paid through PayU, directly from your order management application/platform. The process requires the sending of the order information through HTTP POST, validated with a HMAC_MDS, to the following URL: <https://secure.payu.ro/order/irn.php>. This process must be executed for each order where a payment reversal, partial refund or total refund of the paid amount is required.

What is the data that must be sent to PayU?

The PayU server expects the data packed in the following structure, strictly adhering to the below specified order:

MERCHANT	Your Merchant ID.
ORDER_REF	The PayU reference number.
ORDER_AMOUNT	The total value of the original payment.
ORDER_CURRENCY	The currency in which the order's amount was specified.
IRN_DATE	The date when the reverse/refund request is issued, in the following format: Y-m-d H:i:s (Ex: 2016-04-26 14:30:56).
ORDER_HASH	HMAC_MDS signature for the sent data (HMAC defined at: RFC 2104)
AMOUNT	The amount that must be reversed or refunded. If the AMOUNT is smaller than the ORDER_AMOUNT, a partial refund will be requested. If the AMOUNT is equal to the value of the ORDER_AMOUNT, a total refund will be issued.

Sample array sent through POST:

```
$irn = array(  
    'MERCHANT'=>MERCHANT,  
    'ORDER_REF' => 3954142,  
    'ORDER_AMOUNT'=> 39.99,  
    'ORDER_CURRENCY'=> 'USD',  
    'AMOUNT'=> '12.56',  
    'IRN_DATE'=> date( 'Y-m-d H:i:s' ),  
    'ORDER_HASH'=> 603ced7568f22d656937e8bf0b1b44a9);
```

What is the difference between REVERSE and REFUND?

REVERSE is the procedure to cancel an order still *In Processing*, before it has been captured via the IDN request and the products have been delivered to the final customer.

In case of a REVERSE, the transaction amount that has been locked after the payment authorization will get unlocked by PayU, after approval by the financial department of the REVERSE procedure. The transaction will get a REVERSE status in the PayU administration interface and in the payment notifications (IPN).

A REFUND is the procedure through which a *Complete* order is refunded (after it has been captured via the IDN request).

Once the REVERSE/REFUND procedure has been approved for a specific payment, an IPN/email confirmation will be sent to the merchant, containing the status of the order that has been cancelled (e.g. REVERSE or REFUND) and the total amount that has been cancelled or refunded, displayed with a negative value.

How is the data structured to be sent to PayU?

We assume to have the following data:

Field name	Length (in bytes)	Field value
MERCHANT	8	PAYUDEMO
ORDER_REF	7	1000500
ORDER_AMOUNT	4	22.5
ORDER_CURRENCY	3	RON
IRN_DATE	19	2012-04-26 14:30:56
AMOUNT	5	12.56

To validate the sent data, a HMAC_MDS signature has to be calculated, that will be encoded with the Secret Key attached to your account.

The source string for the HMAC_MDS signature is creating by adding the field length at the beginning of each field value, without "new line" characters (for UTF-8 characters, the string length in bytes may be larger than the number of characters), in the exact same parameter order as found in the table above.

For the data above, the source string is:

8PAYUDEMO71000500422.53RON192012-04-26 14:30:56512.56

The secret key for data validation is: **1231234567890123** .

The HMAC_MDS signature calculated for the data above is:
059a9e53d6b3c55c07c8517445368e2f

How do I receive the PayU response?

PayU validates the successful receive of the information you sent by putting an answer inline in the page that receives the information, like below:

<EPAYMENT>ORDER_REF|RESPONSE_CODE|RESPONSE_MSG|IRN_DATE|ORDER_HASH </EPAYMENT>

The parameters in the validation response sent by PayU are:

ORDER_REF	Order reference in the PayU system received by IRN
RESPONSE_CODE	Response code for the reverse/refund request
RESPONSE_MSG	Response message for the reverse/refund request
IRN_DATE	The date when the order reverse/refund request response has been sent, in the following format: Y-m-d H:i:s (Ex: 2012-04-26 14:30:56)
ORDER_HASH	HMAC_MDS signature for data validation

What response do I receive from PayU?

The answer codes meaning and messages are:

Response Code	Response Message
1	OK.
2	ORDER_REF missing or incorrect.
3	ORDER_AMOUNT missing or incorrect.
4	ORDER_CURRENCY is missing or incorrect.
5	IRN_DATE is not in the correct format.
6	Error confirming order.
7	Order already cancelled.
8	Unknown error.
9	Invalid ORDER_REF.
10	Invalid ORDER_AMOUNT.
11	Invalid ORDER_CURRENCY.

5. Instant Order Search Protocol (IOS)

The IOS protocol makes it possible to query the PayU platform for the status of an order. An IOS request can be triggered through HTTP POST or HTTP GET by accessing:
<https://secure.payu.ro/order/ios.php> .

The query of the PayU database will be made individually, for one order at a time. To query the PayU system for the status of a payment, the following parameters are required:

MERCHANT	Your Merchant ID.
REFNOEXT	The external reference - your own order number, previously sent in the LU request as a value of the ORDER_REF parameter.
HASH	Request signature (HMAC MD5 over a string composed through the fields above)

Authentication

The request is authenticated with a HMAC_MD5 signature calculated on the sent data and your Secret Key. The string on which the signature will be created is composed like this: parameter byte length + parameter value.

Sample:

MERCHANT: PAYUDEMO

REFNOEXT: 1042556

String on which the signature is calculated: **8PAYUDEMO71042556**

Secret key: **1231234567890123**

HASH: **f7ebd84a227d62f89b9f2d93bc5d665f**

Answer

The PayU system will respond **inline**, in XML format, with the following values:

ORDER_DATE	Order date in the format: Y-m-d H:i:s (Ex: 2012-04-26 14:30:56).
REFNO	Global PayU reference number for the order (max. 9 characters).
REFNOEXT	The external reference - your own order number.

ORDERSTATUS	<p>Current order status. Possible values:</p> <ul style="list-style-type: none"> • NOT_FOUND – not existing/unfinished order • WAITING_PAYMENT – the order has been placed and payment is waiting • CARD_NOTAUTHORIZED – the card used for payment has not been authorized • IN_PROGRESS – payment has been authorized, the order is in approval process • PAYMENT_AUTHORIZED – payment authorized, order approved • COMPLETE – finished order (charged/delivered) • FRAUD – fraud suspect order • INVALID – invalid data entered by the customer • TEST – test order • REVERSED – order reversed, money unlocked in the customer account • REFUND – order refund, money wire transferred to the customer account.
PAYMETHOD	Used payment method (maxim 40 characters).
HASH	Request signature (HMAC MD5 on all the fields above).

Sample:

```
<!--?xml version="1.0" ?-->
<order>
    <order_date>2006-10-26 10:15:00</order_date>
    <refno>1074992</refno>
    <refnoext>EPAY10425</refnoext>
    <order_status>PAYMENT_AUTHORIZED</order_status>
    <paymethod>Credit/debit card (Visa/MasterCard)</paymethod>
</order>
```

Note! In case you have sent more than one order with the same order number (external reference), the query will return information for the most recent transaction found in the database for that reference.

 SECURE PAYMENTS BY



Thank you!

