



Payment Platform MontyPay

Integration API (Credit2Virtual)

version 1.1

Content

1	Introduction.....	3
2	Integration Process.....	4
2.1	Client Registration.....	4
2.2	Payment Platform Interaction.....	4
2.3	List of possible actions in Payment Platform.....	4
2.4	List of possible transaction results and statuses.....	4
3	Transactions requests.....	6
3.1	CREDIT2VIRTUAL request.....	6
3.1.1	Request Parameters.....	6
3.1.2	Response Parameters.....	6
3.1.3	Callback parameters.....	7
3.2	GET_TRANS_STATUS request.....	7
3.2.1	Request parameters.....	7
3.2.2	Response parameters.....	8
4	Errors.....	9
	Appendix A.....	10
	Appendix B.....	11

1 Introduction

This document describes integration procedures and CREDIT2VIRTUAL protocol usage for e-commerce merchants.

CREDIT2VIRTUAL protocol implements money transfers transactions to customer's account with using specific API interaction.

2 Integration Process

2.1 Client Registration

Before you get an account to access Payment Platform, you must provide the following data to the Payment Platform administrator.

Data	Description
IP List	List of your IP addresses, from which requests to Payment Platform will be sent.
Callback URL	URL which will be receiving the notifications of the processing results of your request to Payment Platform
Contact e-mail	Client's contact email

With all Payment Platform POST requests at Callback URL the Client must return the string "OK" if he/she successfully received data or return "ERROR".

You should get the following information from administrator to begin working with the Payment Platform.

Parameter	Description
CLIENT_KEY	Unique key to identify the account in Payment Platform (used as request parameter) In the administration platform this parameter corresponds to the "Public key" field
CLIENT_PASS	Password for Client authentication in Payment Platform (used for calculating hash parameter) In the administration platform this parameter corresponds to the "Public secret" field
PAYMENT_URL	URL to request the Payment Platform

2.2 Payment Platform Interaction

For the transaction you must send the **server to server HTTPS POST** request to the Payment Platform URL (PAYMENT_URL). In response Payment Platform will return the JSON (<http://json.org/>) encoded string.

2.3 List of possible actions in Payment Platform

When you make request to Payment Platform, you need to specify action that needs to be done. Possible actions are:

Action	Description
CREDIT2VIRTUAL	Creates CREDIT2VIRTUAL transaction
GET_TRANS_STATUS	Gets status of transaction in Payment Platform

2.4 List of possible transaction results and statuses

Result – value that system returns on request. Possible results are:

Result	Description
SUCCESS	Action was successfully completed in Payment Platform

Result	Description
DECLINED	Result of unsuccessful action in Payment Platform
ERROR	Request has errors and was not validated by Payment Platform

Status – actual status of transaction in Payment Platform. Possible statuses are:

Status	Description
SETTLED	Successful transaction
DECLINED	Not successful transaction
PREPARE	Transaction is accepted and awaiting processing

3 Transactions requests

3.1 CREDIT2VIRTUAL request

CREDIT2VIRTUAL request is used to create CREDIT2VIRTUAL transaction in Payment Platform.

This operation transfers money from merchants account to customer account.

If you want to send a payout for the specific sub-account (channel), you need to use *channel_id* that specified in your Payment Platform account settings.

This request is sent by POST in the background (eg, through PHP CURL).

3.1.1 Request Parameters

Parameter	Description	Values	Required
action	CREDIT2VIRTUAL	CREDIT2VIRTUAL	+
client_key	Unique client key (CLIENT_KEY)		+
channel_id	Payment channel (Sub-account)	String up to 16 characters	-
brand	Brand through which the transaction is performed	String up to 36 characters (Appendix B)	+
order_id	Order ID in the Clients system	String up to 255 characters	+
order_amount	The amount of the transaction	Numbers in the form XXXX.XX (without leading zeros)	+
order_currency	Currency	3-letter code	+
order_description	Description of the transaction (product name)	String up to 1024 characters	+
parameters	Array of the parameters for specific brand	(Appendix B)	+
hash	Special signature to validate your request to Payment Platform	* (Appendix A)	+

3.1.2 Response Parameters

You will get JSON encoded string with transaction result.

SUCCESSFUL RESPONSE

Parameter	Description
action	CREDIT2VIRTUAL
result	SUCCESS
status	SETTLED
order_id	Order ID in the Client's system
trans_id	Transaction ID in the Payment Platform
trans_date	Transaction date in the Payment Platform

UNSUCCESSFUL RESPONSE

Parameter	Description
-----------	-------------

Parameter	Description
action	CREDIT2VIRTUAL
result	DECLINED
status	DECLINED
order_id	Order ID in the Client's system
trans_id	Transaction ID in the Payment Platform
trans_date	Transaction date in the Payment Platform
decline_reason	The reason why the transaction was declined

3.1.3 Callback parameters

SUCCESSFUL RESPONSE

Parameter	Description
action	CREDIT2VIRTUAL
result	SUCCESS
status	SETTLED
order_id	Order ID in the Client's system
trans_id	Transaction ID in the Payment Platform
trans_date	Transaction date in the Payment Platform
hash	Special signature to validate callback. **(Appendix A)

UNSUCCESSFUL RESPONSE

Parameter	Description
action	CREDIT2VIRTUAL
result	DECLINED
order_id	Order ID in the Client's system
trans_id	Transaction ID in the Payment Platform
trans_date	Transaction ID in the Payment Platform
decline_reason	Description of the cancellation of the transaction
hash	Special signature to validate callback. **(Appendix A)

3.2 GET_TRANS_STATUS request

Gets order status from Payment Platform. This request is sent by POST in the background (eg, through PHP CURL).

3.2.1 Request parameters

Parameter	Description	Values	Required
action	GET_TRANS_STATUS	GET_TRANS_STATUS	+

Parameter	Description	Values	Required
client_key	Unique client key (CLIENT_KEY)		+
trans_id	Transaction ID in the Payment Platform	String up to 255 characters	+
hash	Special signature to validate your request to Payment Platform	***(Appendix A)	+

3.2.2 Response parameters

Parameter	Description
action	GET_TRANS_STATUS
result	SUCCESS
status	DECLINED/SETTLED
order_id	Order ID in the Client's system
trans_id	Transaction ID in the Payment Platform
decline_reason	Reason of transaction decline. It shows for the transactions with the "DECLINED" status

4 Errors

In case error you get synchronous response from Payment Platform:

Parameter	Description
result	ERROR
error_message	Error message

Appendix A

Hash is signature rule used either to validate your requests to payment platform or to validate callback from payment platform to your system.

It must be md5 encoded string calculated by rules below:

* hash is calculated by the formula:

```
$hash = md5(strtoupper(strrev($order_id . $order_amount . $order_currency)) . $CLIENT_PASS);
```

** hash is calculated by the formula:

```
$hash = md5(strtoupper(strrev($trans_id . $order_id . $status)) . $CLIENT_PASS);
```

*** hash is calculated by the formula:

```
$hash = md5(strtoupper(strrev($trans_id)) . $CLIENT_PASS);
```

Appendix B

You have to add to your CREDIT2VIRTUAL request specific list of parameters which is determined by the value for the “brand” parameter.

You should get additional information from account manager.

Paygate10 Netbanking/UPI

If you set the value “netbanking-upi” for the “brand” parameter you have to specify in your request the next parameters as well:

Parameter	Description	Values	Required
country_code	Customer’s country. Must be 2-letter code	String 2 characters	+
user_name	Merchant’s customer “user name”	String up to 100 characters	+
account_name	Beneficiary Bank Account Name	String up to 100 characters	+
account_number	Beneficiary Bank Account Number	String up to 20 characters	+
bank_name	Bank name	String up to 100 characters	+
bank_ifsc	Bank IFSC code	String up to 20 characters	+
bank_branch	Bank branch name	String up to 100 characters	+
bank_address	Bank address	String up to 250 characters	+

Sample curl request

```
curl -d "action=CREDIT2VIRTUAL&client_key=c2b8fb04-110f-11ea-bcd3-0242c0a85004&brand=netbanking-upi&order_id=ORDER12345&order_amount=1.99&order_currency=USD&order_description=Product&parameters[country_code]=IN&parameters[user_name]=test&parameters[account_name]=test&parameters[account_number]=1234567890&parameters[bank_name]=apnaBank&parameters[bank_ifsc]=ASBL00000007&parameters[bank_branch]=Mumbai&parameters[bank_address]=Sakinaka]&hash="a1a6de416405ada72bb47a49176471dc" https://test.apiurl.com -k
```