

API en

Deposit of funds (API)



Warning!

This integration method is NOT recommended if you have not previously integrated similar payment solutions that require activation. [Recommended integration method](#)

When choosing integration via API, we recommend that you additionally make the following improvements:

1. Add hints in text form for each payment system (PS) to explain to a user all payment steps.
2. Add a visual explanation of the payment process - gifs showing the process and screenshots of the payment steps.
3. Screenshots and all the necessary minimum text information can be obtained directly from Paykassma TS.

Access to the shared API is implemented using a secret key. PLUGINAPI_SECRET - secret key string. It will need to be added to all requests in the form of the secret get parameter.

For example: https://api.{URL_Paykassma_Server}/api/v1/deposit-info?secret=pluginapi_secret

When you send a request without using these parameters, the error 404 'Not found' will be displayed

Attention: Before starting to work with the API for deposit, it is necessary to send the IP address from which requests will be sent to the Paykassma TS.

Obtaining available PSs

API url
<code>https://api.{URL_Paykassma_Server}/api/v1/payment-systems/available</code>

Request method: GET

Format: JSON

Parameters:

Parameter	Type	Required	Description
currency_code	string[] (3)	YES	An array consisting of currency codes. PS that work with the specified currencies will be displayed
label	string (190)	YES	Unique identifier (ID) of the user making the payment

Request
<pre>{ "currency_code": ["USD", "INR"], "label": "55" }</pre>

- Deposit of funds (API)
 - Obtaining available PSs
 - Getting mediahints
 - Creating a payment
 - 1. Payment creation request:
 - 2. Activation request:
 - For PS that do not require payment activation
 - Payment creation request:
- Information about savings
- Redirect addresses hashing
- Withdrawal of funds
 - Signature generation algorithm
 - Withdrawal status
- Information about the postback that was sent earlier
- Accounts' balance
- Order statuses

Response Success

```
{
  "status": "ok"
  "params": {
    "currency_data": [
      {
        "code": "USD",
        "symbol": "$"
      },
      {
        "code": "INR",
        "symbol": "INR"
      }
    ],
    "wallets": [
      {
        "type": "staging_b_ml",
        "currency_code": "INR",
        "order": 6,
        "logo": "https://api.staging.client-paykassma.com/storage/payment_system/logos/staging_b_ml.png",
        "name": "staging B ml",
        "is_manual": true,
        "uploading_payment_bill": true
      },
      {
        "type": "nagad",
        "currency_code": "USD",
        "order": 17,
        "logo": "https://api.staging.client-paykassma.com/storage/payment_system/logos/nagad_default.svg",
        "name": "Nagad",
        "is_manual": false,
        "uploading_payment_bill": false
      }
    ]
  }
}
```

Response Fail

```
{
  "status": "fail"
  "message": "MESSAGE"
}
```

Getting mediahints

To receive mediahints on how to make a deposit using the payment system, you need to send a request:

API url

`https://api.{URL_server_Paykassma}/wallet-type/images/`

Request method: GET

Format: JSON

Response Success

```
"status": "success",
"message": "",
"code": 20000,
"data": {
  "imps": {
    "id": 486,
    "url": "https://api.staging.client-paykassma.com/storage/guide-image/bkash.mp4"
  },
  "paytm": {
    "id": 500,
    "url": "https://api.staging.client-paykassma.com/storage/guide-image/Ghjkfre22.gif"
  },
  "upi": {
    "id": 506,
    "url": "https://api.staging.client-paykassma.com/storage/guide-image/upi_in.mp4"
  }
}
```

Response Fail

```
{
  "status": "fail"
  "message": "MESSAGE"
}
```

Creating a payment

Currently, there are two types of requests for creating a payment, depending on the PS.



Attention! To get a complete list of available PSs, you need to contact Paykassma TS. TS will provide complete examples of requests for currently available PSs. They will also determine the correct request, depending on the selected PS.

For PS that require payment activation

1. Payment creation request:

API url

```
https://api.{URL_Paykassma_Server}/api/v1/transaction/create/{wallet_type}
```

Example of request for PS PayTM

API url

```
https://api.{URL_Paykassma_Server}/api/v1/transaction/create/paytm
```

Request method: POST

Format: JSON

Parameters:

Parameter	Type	Required	Description
currency	string (3)	YES	Currency code
label	string (190)	YES	Unique identifier of the user making the payment

Request

```
{
  "currency": "INR",
  "label": "eligendi"
}
```

Response Success

```
{
  "status": "ok",
  "params": {
    "identifier": "6234234234"
  }
}
```

Response Fail

```
{
  "status": "fail"
  "message": "MESSAGE"
}
```

2. Activation request

Depending on the value of the **is_manual** parameter, there are two types of requests for payment activation:

1) if *is_manual* = false

API url

`https://api.{URL_Paykassma_Server}/api/v1/transaction/activate`

Request method: POST

Format: JSON

Parameters:

Parameter	Type	Required	Description
currency_code	string (3)	YES	Currency code
wallet_type	string (190)	YES	Value from the list of available PSs. Attention: To get a complete available list, contact Paykassma TS.
label	string (190)	YES	Unique identifier of the user making the payment
key1	string (190)	YES	A code on which activation is performed (Transiction_ID), received fro
amount	float	*required if	Transaction amount
custom_transaction_id	string (64)	YES	Unique transaction identifier in the Client's system <i>It is necessary to pass a non-repeating parameter value</i>

Request

```
{
  "currency_code": "USD",
  "wallet_type": "paytm",
  "label": "55",
  "key1": "123-ABC-456-DEF"
  "custom_transaction_id": "123465477897"
}
```

Response Success

```
{
  "status": "ok"
}
```

Response Fail

```
{
  "status": "fail"
  "message": "MESSAGE"
}
```

2) if *is_manual* = true

API url

`https://api.{URL_server_Paykassma}/api/v1/possible-transaction`

Request method: POST

Format: JSON

Parameters:

Name	Type	Required	Description
secret	string	yes	

Body:

Parameter	Type	Required	Description
currency_code	string (3)	YES	Currency code
wallet_type	string (190)	YES	Value from the list of available PSs. Attention: To get a complete available list, contact Paykassma TS.
label	string (190)	YES	Unique identifier of the user making the payment
key1	string (190)	YES	A code on which activation is performed (Transiction_ID), received from the user
amount	float	YES	Transaction amount
screenshot	string (190)	NO	Upload payment screenshot
custom_id	string (190)	YES	Transaction ID in the Client's system

Response Success

```
{
  "status": "success",
  "message": "",
  "code": 20000,
  "data": [],
  "paginate": {
    "offset": 0,
    "total": 1,
    "limit": 1
  }
}
```

Response Fail

```
{
  "status": "fail",
  "message": {
    "key1": [
      "Transaction field has wrong length"
    ]
  }
}
```

*With enabled functionality [Delayed automatic activation of transactions\(Possible transactions\)](#)

Response possible_transaction_created

```
{
  "status": "possible_transaction_created",
  "message": "Transaction not found. Possible transaction created"
}
```

**For some PS, it is also necessary to send the amount field in the request:

Payment system	currency_code	wallet_type	label	key1	amount	custom_transaction_id
UPI P2P	required	required	required	required	required	required
PhonePe	required	required	required	required	required	required
bKash	required	required	required	required	-	required
Nagad	required	required	required	required	-	required

***If you try to activate a suspicious transaction, you will see **error 400** 'Check of deposit status started. The operation may take longer than expected. Once checked the funds will credit to you account'.

For PS that do not require payment activation

Payment creation request:

API url

```
https://api.{URL_Paykassma_Server}/api/v1/payment/create/{payment_system}
```

Request method: POST

Format: JSON

Parameters:

Parameter	Type	Required	Description
language	string (2)	YES	Interface language
currency	string (3)	YES	Currency code
bank	string (3)	YES	The bank via which the payment is performed
amount	float	YES	Payment amount
label	string (190)	YES	Unique identifier of the user making the payment
redirect_url	url	YES	URL of the page for redirecting after payment
custom_transaction_id	string (64)	YES	Unique transaction identifier in the Client's system <i>It is necessary to pass a non-repeating parameter value</i>

Request

```
{
  "language": "en",
  "currency": "USD",
  "bank": "MBB",
  "amount": 1000.00,
  "label": "55",
  "redirect_url": "https:\\\\google.com"
  "custom_transaction_id": "123465477897"
}
```

Response Success

```
{
  "status": "ok"
}
```

Response Fail

```
{
  "status": "fail"
  "message": "MESSAGE"
}
```

Example of request for PS Bkash API**API url**

```
http://api.{ URL_server_Paykassma }/api/v1/payment/create/bkash_api
```

Request method: POST**Format:** JSON**Params:**

Name	Type	Required	Description
secret	string	YES	Access key

Body:

Name	Type	Required	Description
amount	float (limited by min and max. deposit)	YES	Payment amount
account_number	string (11)	YES	The eleven digit account number of the Costumer's user in Bkash, in the format 01 Validation: <ul style="list-style-type: none"> numbers: 0-9
currency	string (3)	YES	Currency code
label	string (190)	YES	Unique ID of the user making the payment
custom_transaction_id	string (64)	YES	Unique transaction identifier in the Client's system <i>It is necessary to pass a non-repeating parameter value</i>
success_url	string (unlimited)	YES	Link to redirect the user from the Bkash page in case of successful payment
failure_url	string (unlimited)	YES	Link to redirect the user from the Bkash page in case of a payment error
cancel_url	string (unlimited)	YES	Link to redirect the user from the Bkash page in case of payment cancellation

Request

```
{
  "amount": 1.00,
  "account_number": "01770618575",
  "currency": "BDT",
  "label": "test",
  "custom_transaction_id": "01abc0095689",
  "success_url": "https://static5.depositphotos.com/1029663/395/i/600/depositphotos_3955476-stock-photo-green-check-mark.jpg?status=success",
  "failure_url": "https://top-fon.com/uploads/posts/2021-07/1626870875_29-p-krasnii-krest-fon-32.png?status=failure",
  "cancel_url": "https://yandex.ru/"
}
```

Response Success

```
{
  "status": "ok",
  "data": {
    "orderId": "1cb4627a420a6e69",
    "bkashUrl": "https://sandbox.payment.bkash.com/redirect/tokenized/?paymentID=TR00114E1661778153815&hash=dcRq9geuCt jkN6rFpaGxdLAMrmyxIxL*kEN-p41SLL2M!xXy9m20TUTLwlAGMVWDKyE8xkOHj_N_AHaTxNGT*(V2LXyC5y4zpUsG1661778154005&mode=0011&apiVersion=v1.2.0-beta",
    "message": "Successful"
  }
}
```

Response Fail

```
{
  'status': 'fail'
  'message': 'MESSAGE'
}
```

Example of request for PS Nagad API

API url

`http://api.{ URL_server_Paykassma }/api/v1/payment/create/nagad_api`

Request method: POST

Format: JSON

Params:

Name	Type	Required	Description
secret	string	YES	Access key

Body:

Name	Type	Required	Description
amount	float (limited by min and max. deposit)	YES	Payment amount
currency	string (3)	YES	Currency code
label	string (190)	YES	Unique ID of the user making the payment
custom_transaction_id	string (64)	YES	Unique transaction identifier in the Client's system <i>It is necessary to pass a non-repeating parameter value</i>
success_url	string (unlimited)	YES	Link to redirect the user from the Nagad page in case of successful payment
fail_url	string (unlimited)	YES	Link to redirect the user from the Nagad page in case of a payment error
cancel_url	string (unlimited)	YES	Link to redirect the user from the Nagad page in case of payment cancellation

Request

```
{
  "custom_transaction_id": "custom_transaction_1",
  "amount": 100.00,
  "currency": "BDT",
  "label": "test_user",
  "success_url": "https://success_url.com",
  "fail_url": "https://fail_url.com",
  "cancel_url": "https://cancel_url.com"
}
```

Response Success

```
{
  "status": "ok",
  "payment_url": "callBackUrl"//link to the Nagad payment page, where
the user must enter all the necessary details and make a payment
}
```

Response Fail

```
{
  'status': 'fail'
  'message': 'MESSAGE'
}
```

Information about savings

To save information about transactions, the amount of which does not exceed the minimum deposit amount for a particular payment system, the functionality of savings has been implemented. The user's transactions (selected by the label parameter) are accumulated in the system and are not credited to the user's account until their amount reaches the minimum deposit amount of this payment system. The [postback](#) will be sent to the client's server only after the minimum deposit amount is reached. Until the [postback](#) is sent, the transaction is not considered credited to the account.

In the [postback](#) of deposit transactions, you can see the Stockpiling field, it contains the amount of transactions involved in the accumulation (one or more transactions) converted to all currencies. Conversion is carried out on the basis of open data on exchange rates. All currencies are transmitted with an accuracy of up to hundredths, cryptocurrencies have 8 digits in the fractional part (one hundred millionth accuracy). And in field **transactions** Information about completed transactions participating in the accumulation (one or more) is transmitted. Passed as an list.

Request to provide information about savings of a certain user

API url

```
https://api.{URL_Paykassma_Server}/api/v1/deposit-info
```

Request method: GET

Format: JSON

Parameters:

Parameter	Type	Required	Description
label	string (190)	YES	Unique identifier of the user making the payment
wallet_type	url	YES	Value from the list of available PSs. Attention: To get a complete available list, contact Paykassma TS.
currency	string (3)	YES	Currency code
code	string (190) or "null"	NO	Name of the manual payment system. It's filled in when wallet_type = manual

Request

```
{
  "label": "55",
  "wallet_type": "paytm",
  "currency": "UDS"
}
```

Response Success

```
{
  "status": "ok",
  "stockpiling_sum": 0,
  "lacks_sum": 0,
  "minimum_deposit": 148
}
```

Response Fail

```
{
  "status": "fail",
  "message": "MESSAGE"
}
```

Redirect addresses hashing

The widget code contains parameters for specifying redirect addresses. Addresses must be specified in hashed form. To hash addresses send the request ([i](#) access to the API is implemented using a secret key):

API url

```
https://api.{URL_server_Paykassma}/payment-plugin/hash-url?secret=pluginapi_secret
```

Request method: POST

Format: JSON

Parameters:

Parameter	Type	Required	Description
success_urls	array	YES	The list of redirect addresses when the payment is successful .
order	integer	*YES	The redirect address number in the list.
url	string	*YES	The redirect address when the payment is successful .
fail_urls	array	YES	The list of redirect addresses when the payment is failed .
order	integer	*YES	The redirect address number in the list.
url	string	*YES	The redirect address when the payment is failed .
pending_urls	array	YES	The list of redirect addresses while waiting for payment confirmation .
order	integer	If one of the properties is specified, then the second is required.	The redirect address number in the list.
url	string		The redirect addresses while waiting for payment confirmation .

**One of the success_urls or fail_urls arrays can be empty.*

Request

```
{
  "success_urls": [
    {
      "order": 2,
      "url": "https://test.com/success2.jpg"
    },
    {
      "order": 1,
      "url": "https://test.com/success1.jpg"
    }
  ],
  "fail_urls": [
    {
      "order": 1,
      "url": "https://test.com/fail1.jpg"
    },
    {
      "order": 2,
      "url": "https://test.com/fail2.jpg"
    }
  ],
  "pending_urls": [
    {
      "order": 3,
      "url": "https://test.com/pending1.jpg"
    },
    {
      "order": 4,
      "url": "https://test.com/pending2.jpg"
    }
  ]
}
```

Response Success

```
{
  "status": "ok",
  "urls": {
    "success_url": [
      {
        "order": 1,
        "url": "a93558ba9e4f04eef3b1f759ae4281305776559ce86ae2af5f6e6498d84b99be"
      },
      {
        "order": 2,
        "url": "7c51f9d6b35a80a8fcce9cf3f4f0ae56b818c2e375a8b3297cb3fafb41a921eb"
      }
    ],
    "fail_url": [
      {
        "order": 1,
        "url": "8b99db57e0bda63400b0e852d7fe73d8c6da52b8e32894987a8bd00bcadf8c66"
      },
      {
        "order": 2,
        "url": "e72ed31e2f3814563a3bc446f6a59f43c946c3a4339c7b2f5c29d12e8066b37c"
      }
    ],
    "pending_url": [
      {
        "order": 1,
        "url": "8b99db57e0bda63400b0e852d7fe73d8c6da52b8e32894987a8bd00bcadf9v69"
      },
      {
        "order": 2,
        "url": "e72ed31e2f3814563a3bc446f6a59f43c946c3a4339c7b2f5c29d12e8066b37f"
      }
    ]
  }
}
```

Response Fail

```
{
  "status": "fail"
  "message": "MESSAGE"
}
```

The received address hashes must be specified in the parameters of the widget code.



Attention!

The hash lifetime is 1 day. After the expiration date resend the request to get hash for the address if necessary.

[< Home](#)

Withdrawal of funds

To withdraw funds by the user via payment systems available to the Client, it is necessary to:

1. Once send Paykassma TS the URL of the server where [postbacks](#) with the withdrawal status will be sent.
2. Send withdrawal requests (POST) to the URL, without using "secret = pluginapi_secret":

API url

`https://api.{URL_Paykassma_Server}/v2/withdrawal/create`



Attention: When transmitting data, there *should be no spaces* at the beginning and end

Each withdrawal request is signed by **signature**, it is formed as follows:

Signature generation algorithm

1. Your private key is initialized.
2. An array with data that will be hashed is formed. The array can be multidimensional.
3. Array elements are sorted by key using the `ksort()` function
4. The array elements are recursively concatenated with each other via ":" string.
5. Then md5 hash of the resulting string is allocated.
6. The private key is concatenated with a hash string.
7. Then sha1 hash is allocated from the resulting string.

An example of forming a signature in php

```
$data = [
    ...
];

$data['signature'] = generateSignature($data); // return
fa0ee1e2acf7f898635ec417491381c0a4f9d35c




//send $data like json ...

function generateSignature(array $data) {
    $privateKey = 'yourprivatekey';
    ksort($data);
    $implode = self::multiImplode( ':', $data);
    return sha1($privateKey . md5($implode));
}

function multiImplode($glue, $array): string
{
    $finArray = [];
    foreach ($array as $val) {
        $finArray[] = is_array($val) ? self::multiImplode($glue, $val) :
$val;
    }
    return implode($glue, $finArray);
}
```

Required parameters for transmission in POST request:

Parameter	Type	Required	Description
withdrawal_id	string(1-36)	optional	Withdrawal ID, if th
payment_system	string	required	Payment system

amount	int	required	Additional req 1. amount >= Tt 2. amount <= Th <div> Multiplici For PS P:</div>
currency_code	string(3)	required	Currency code
label	string	required	Unique identifier of
is_test	boolean	required	The is_test param true - if Withdrawal false - if Withdrawal
account_number	string	*required_if PS = bkash,imps_ib, upi_ib,paytm,....	Number of the bank <div> Input ma For PS P: For PS UI For PS BI For PS R:</div>
account_name	string	*required_if PS = imps_ib, upi_ib,....	Name of the bank :
account_email	string	*required_if PS = bkash,....	Email of the person
payments_details.payments_method	string	*required_if	Payment method
payments_details.payments_provider	string	optional	Supplier who make
bank_details.bank_code	string	*required_if	Unique ID assigned <div> Input ma For PS IM dashes.. I</div>
bank_details.branch_code	string	*required_if	Bank branch code
bank_details.bank_code_in_payments_system	string	*required_if	Bank code in paym
account_type	string	*required_if	Account type
document_type	string	*required_if	Document type
document_id	string	*required_if	Recipient docu
account_digit	string	*required_if	Account digit
iban	string	*required_if	IBAN



Information

*The withdrawal request contains fields that are required only for some payment systems (required_if), to get a complete list of required fields for a specific payment system, you need to contact Paykassma technical support.

Required fields for PS Bkash, PayTm, IMPS IB/UPI IB:

Payment system	account_number	account_name	account_email	payments_details		bank_details		
				payments_method	payments_provider	bank_code	branch_code	bank_code_in
Bkash	required	-	required	-	-	-	-	

IMPS IB	required	required	-	-	-	required	-	
Paytm	required	-	-	-	-	-	-	

Request:

Example request

```
{
  "withdrawal_id": "1234567",
  "payment_system": "paytm",
  "amount": 1000,
  "currency_code": "INR",
  "label": "55",
  "is_test": false,
  "comment": "withdrawal",
  "account_number": "11111111",
  "payments_details": {
    "payments_provider": "56",
    "payments_method": "MOBILE_MONEY"
  },
  "bank_details": {
    "bank_code": "MAHB0001821",
    "branch_code": "000254",
    "bank_code_in_payments_system": "9393"
  },
  "signature": "fae51673a8a9c85f724317214bd0bcc665ebc799"
}
```

Response Success

```
{
  "status": "ok"
}
```

Response Fail

```
{
  "status": "fail"
  "message": "MESSAGE"
}
```

*If, when trying to create a withdrawal request, the amount of account balances in the currency of the request is less than the total amount of new and still incomplete applications, the **error message** "Insufficient balance" will be displayed (to configure this functionality, contact the TP Paykassma).

Response Fail

```
{
  "status": "fail",
  "message": {
    "withdrawals_sum_amount": [
      "Insufficient balance"
    ]
  }
}
```

After processing the withdrawal, Paykassma will send a [postback](#) about the result of the withdrawal with a status, signed with a signature.

Withdrawal status

Status	Description
PROCESSED (1)	Uccessful
REJECTED (5)	Unsuccessful

Information about the postback that was sent earlier

Request to provide information about the postback that was sent earlier.

PLUGINAPI_SECRET - secret key string. It will need to be added to all requests in the form of the secret get parameter.


For example: https://api.{URL__Paykassma}/api/v1/postback?secret=pluginapi_secret

API url
<code>https://api.{URL_server_Paykassma}/api/v1/postback</code>

Request method: GET

Format: JSON

Parameters:

Parameter	Type	Required	Description
<code>type</code>	string	required	Type of postback. Possible values: <ul style="list-style-type: none"> • <code>deposit</code> • <code>withdrawal</code>
<code>transaction_id</code>	string	One of the parameters is required if <code>type = deposit</code>	Transaction number
<code>custom_id</code>	string		Order ID <div>  If Paykassma contains several transactions with the same Order ID, only one of them will be selected. For this reason, the uniqueness of <code>custom_id</code> must be controlled on the Client's system. </div>
<code>withdrawal_id</code>	string	Is required if <code>type = withdrawal</code>	Withdrawal ID

Response Success (type=deposit)
<pre>{ "status": "ok", "postback": { "transaction_id": "453535345", "label": "435", "stockpiling": { "USD": 4610062.05, "INR": 345345345, "EUR": 3983190.43, "IDR": 65837909699.54, "MYR": 19134062.55, "VND": 104893839449.17, "SGD": 6221984.08, "THB": 153749990.51, "NGN": 1898193050.29, </pre>

"TRY": 43873352.03,
"AED": 16933679.93,
"CAD": 5706403.96,
"AUD": 6134591.13,
"BDT": 394891290.79,
"AFN": 418156964.72,
"ALL": 488772456.91,
"AMD": 2202143869.02,
"AOA": 2752207045.6,
"ARS": 459519460.28,
"AWG": 8300416.73,
"AZN": 7840816.59,
"BAM": 7789732.49,
"BBD": 9220124.11,
"BGN": 7790566.91,
"BHD": 1737744.45,
"BIF": 9169348711.83,
"BMD": 4610062.05,
"BND": 6216396.68,
"BOB": 31801277.58,
"BRL": 26023403.82,
"BSD": 4610062.05,
"BTC": 74.54470339,
"BTN": 345299119.91,
"BWP": 52763690.74,
"BYN": 11244314.76,
"BZD": 9240952.37,
"CDF": 9143497429.28,
"CHF": 4208184.5,
"CLF": 135913.85,
"CLP": 3750285480.05,
"CNH": 29495130.91,
"CNY": 29495177.01,
"COP": 17331717342.01,
"CRC": 2939207744.86,
"CUC": 4610062.05,
"CUP": 118709097.86,
"CVE": 436572876.41,
"CZK": 102052399.68,
"DJF": 820509207.6,
"DKK": 29632944.11,
"DOP": 260174176.58,
"DZD": 634030335.09,
"EGP": 72425457.87,
"ERN": 69153789.03,
"ETB": 218717216.23,
"FJD": 9542828.45,
"FKP": 3369190.09,
"GBP": 3369190.09,
"GEL": 14521695.47,
"GGP": 3369190.09,
"GHS": 28121143.41,
"GIP": 3369190.09,
"GMD": 239723226.75,
"GNF": 44360498102.61,
"GTQ": 35672982.87,
"GYD": 964839822.52,
"HKD": 35879882.45,
"HNL": 111443474.11,
"HRK": 29978196.27,
"HTG": 452476986.58,
"HUF": 1435716207.54,
"ILS": 14476908.71,
"IMP": 3369190.09,
"IQD": 6725977357.42,
"IRR": 194775122390.71,
"ISK": 598293853.23,
"JEP": 3369190.09,
"JMD": 712559367.6,
"JOD": 3268534,
"JPY": 526637353.71,

"KES": 513053805.87,
"KGS": 390935115.33,
"KHR": 18835091650.05,
"KMF": 1942681734.96,
"KPW": 4149055847.63,
"KRW": 5430630260.1,
"KWD": 1391132.33,
"KYD": 3840638.09,
"KZT": 1971237418.21,
"LAK": 47201661108.42,
"LBP": 6965551762.15,
"LKR": 930998873.7,
"LRD": 698424377.97,
"LSL": 70723847.09,
"LYD": 21015783.85,
"MAD": 41858708.81,
"MDL": 80357078.85,
"MGA": 18336748035.86,
"MKD": 245401836.65,
"MMK": 8326080171.86,
"MNT": 13147861127.1,
"MOP": 36946291.23,
"MRO": 1645791359.96,
"MRU": 166749005.54,
"MUR": 198463171.38,
"MVR": 71225458.72,
"MWK": 3763215975.72,
"MXN": 95382589.56,
"MZN": 294260260.84,
"NAD": 69704138.24,
"NIO": 162371876.09,
"NOK": 38796622.62,
"NPR": 552477876.37,
"NZD": 6414537.15,
"OMR": 1774767.86,
"PAB": 4610062.05,
"PEN": 18422162.94,
"PGK": 16182622.45,
"PHP": 233052471.57,
"PKR": 792501444.06,
"PLN": 18414837.55,
"PYG": 31817465481.06,
"QAR": 16780842.55,
"RON": 19712487.04,
"RSD": 468340620.4,
"RUB": 327660160.41,
"RWF": 4695797248.61,
"SAR": 17293947.06,
"SBD": 37028825.17,
"SCR": 63024218.26,
"SDG": 2030732334.31,
"SEK": 39500786.54,
"SHP": 3369190.09,
"SLL": 49746487532.16,
"SOS": 2666103446.02,
"SRD": 99148604.57,
"SSP": 600506683.01,
"STD": 96610491131.87,
"STN": 98424824.83,
"SVC": 40330104.44,
"SYP": 5794559138.65,
"SZL": 70735399.9,
"TJS": 51758475.93,
"TMT": 16158267.5,
"TND": 12965799.52,
"TOP": 10312565.9,
"TTD": 31286296.76,
"TWD": 128400374.92,
"TZS": 10616972907.89,
"UAH": 121246397.65,
"UGX": 16393718610.02,

```
"UYU": 202666561.42,
"UZS": 49202252751,
"VEF": 1145543450116.04,
"VES": 19910627.5,
"VUV": 512750542.16,
"WST": 11839063.48,
"XAF": 2612802535.54,
"XAG": 192248.81,
"XAU": 2577.02,
"XCD": 12458923.2,
"XDR": 3272982.71,
"XOF": 2612802535.54,
"XPD": 2235.88,
"XPF": 475321217.85,
"XPT": 4398,
"YER": 1153668019.52,
"ZAR": 70640049.99,
"ZMW": 79757295.95,
"ZWL": 1484439981.04,
"ETH": 1054.27509088,
"LTC": 110641.4892702,
"DOGE": 16950326.08316667,
"DASH": 65854.73642603,
"BCH": 18901.25441699,
"ZEC": 126163.56820239,
"ETC": 878106.17959294,
"XRP": 24711011.35819838,
"TRX": 45776463.64719361,
"XLM": 98741900.28788687,
"WAVES": 172375.10682464
},
"stockpiling_id": 1585,
"transactions": [
  {
    "amount": 345345345,
    "currency_code": "INR",
    "wallet_type": "phone_pe",
    "transaction_id": "453535345",
    "transaction_type": 0,
    "from": null,
    "created_datetime": "2021-10-21 13:34:12",
    "activated_datetime": "2021-11-01 14:16:51",
    "custom_id": null
  }
]
}
```

Response Success (type=withdrawal)

```
{
  "status": "ok",
  "postback": {
    "withdrawal_id": "2298635",
    "status": 1,
    "comment": "test",
    "payment_system": "paytm",
    "amount": 500,
    "currency_code": "INR",
    "label": "55",
    "account_number": "5422222225",
    "account_name": "5645646",
    "account_email": null,
    "payments_details": {
      "payments_provider": null
    },
    "bank_details": {
      "bank_code": null,
      "branch_code": null
    }
  }
}
```

Response Fail

```
{
  "status": "fail"
  "message": "MESSAGE"
}
```

[< Home](#)

Accounts' balance

A request to provide the information about the balance of the Client's accounts (except for the Settlement balance) is formed as follows:

API url

```
https://api.{URL_server_Paykassma}/api/v1/accounts-balance?secret=
{{pluginapi.secret}}
```

Request method: GET

Format: JSON

Parameters:

Name	Type	Required	Description
secret	string	yes	Access key

Response Success

```
{
  "status": "success",
  "message": "",
  "code": 20000,
  "data": {
    "balances": [
      {
        "wallet_type": "paytm",
        "currency_code": "IDR",
        "amount": -324324.00
      },
      {
        "wallet_type": "paytm",
        "currency_code": "USD",
        "amount": -10800.00
      },
      {
        "wallet_type": "bkash",
        "currency_code": "VND",
        "amount": 155.00
      },
      {
        "wallet_type": "bkash",
        "currency_code": "BRL",
        "amount": 187239.195
      },
      {
        "wallet_type": "bkash",
        "currency_code": "EUR",
        "amount": 23.9375
      },
      {
        "wallet_type": "nagad",
        "currency_code": "CUC",
        "amount": 471.00
      },
      {
        "wallet_type": "upi_ib",
        "currency_code": "SAR",
        "amount": 499.00
      }
    ]
  }
}
```

Response Fail

```
{
  "status": "error",
  "code": 30228,
  "errors": {
    "target": "secret",
    "code": 36205,
    "message": "Wrong secret"
  },
  "error_message": "Data validation error"
}
```

Order statuses

A request to provide the information about the statuses of the Client's orders is formed as follows:

API url
<code>https://api.{URL_server_Paykassma}/api/v1/pullpostbacks?secret={{pluginapi.secret}}</code>

Request method: GET

Format: JSON

Parameters:

Name	Type	Required	Description
secret	string	yes	Access key

Name	Type	Required	Description
<code>type</code>	string	YES	Possible values: <ul style="list-style-type: none">• deposit• withdrawal
<code>plugin_custom_order_id</code>	array	YES if <i>type = withdrawal</i>	User transaction ID/Order ID
<code>withdrawal_id</code>	array	YES if <i>type = plugin_custom_order_id</i>	Withdrawal ID

Response body:

Name	Type	Required	Description
status	string	YES	Request status
message	string	To be filled in if the response is successful	Message on successful response
code	int	YES	Response code
data	array /object	To be filled in if the response is successful	
transactions	array	YES <i>if type = deposit</i>	Deposit information
plugin_custom_order_id	string	YES	User transaction ID/Order ID
message	string	YES <i>if the transaction is not found</i>	A message if the transaction was not found
postback	array	YES <i>if a transaction is found</i>	Postback information
withdrawals	array	YES <i>if type = withdrawal</i>	Withdrawal information
withdrawal_id	string	YES	Withdrawal ID
message	string	YES <i>if the withdrawal is not found</i>	A message if the transaction was not found
postback	array	YES <i>if the withdrawal is not found</i>	Postback information

Response Success (type = deposit)

```
{
  "status": "success",
  "message": "",
  "code": 20000,
  "data": {
    "transactions": [
      {
        "plugin_custom_order_id": "autotest51191116971521",
        "message": "Postback not found"
      },
      {
        "plugin_custom_order_id": "autotest51191472141726",
        "postback": {
          "label": "autotest51191472141726",
          "stockpiling": {
            "USD": 3.6,
            "EUR": 3.37,
            "IDR": 55405.29,
            "MYR": 16.92,
            "VND": 87878.53,
            "SGD": 4.91,
            "THB": 129.67,
            "NGN": 2844.58,
            "TRY": 97.39,
            "AED": 13.23,
            "CAD": 4.85,
          },
          "stockpiling_id": 269866,
          "transactions": [
            {
              "amount": 300.00,
              "currency_code": "INR",
              "wallet_type": "upi_1",
              "transaction_id": "770425808190",
              "transaction_type": 0,
              "from": "autotest51190375323527",
              "created_datetime": "2023-09-19 10:25:47",
              "activated_datetime": "2023-09-19 10:25:48",
              "custom_id": "autotest51191472141726"
            }
          ]
        }
      }
    ]
  },
  "paginate": {
    "offset": 0,
    "total": 2,
    "limit": 2
  }
}
```


Response Success (type = withdrawal)

```
{
  "status": "success",
  "message": "",
  "code": 20000,
  "data": {
    "withdrawals": [
      {
        "withdrawal_id": "autotest51180853781767",
        "postback": {
          "withdrawal_id": "autotest51180853781767",
          "status": 1,
          "comment": "autotest approve by file",
          "payment_system": "upi_idb",
          "amount": 330,
          "currency_code": "INR",
          "label": "autotest95118085378184",
          "account_number": "1950478885",
          "account_name": "autotest95117145615335",
          "account_email": "",
          "payments_details": {
            "payments_provider": null
          },
          "bank_details": {
            "bank_code": null,
            "branch_code": null
          }
        }
      }
    ]
  },
  "paginate": {
    "offset": 0,
    "total": 1,
    "limit": 1
  }
}
```

Response Fail

```
{
  "status": "fail",
  "message": {
    "plugin_custom_order_id": [
      "The plugin custom order id field is required when type is
      deposit."
    ]
  }
}
```