EmbeddedWalletAPI Documentation v1.0

Getting Started

- Body of requests must contain valid JSON and content type equal to **application/json**
- All requests will result in a 200 Ok response unless there is a server or infrastructure error. The API result will be wrapped in a JSON Result object, where the **errorCode** field indicates response status.
- All requests must have signed with HMACSHA256 sign

Request Signing

All calls to the API are sent via HTTP using POST and must contain the following headers:

RP-CompanyId - Your company unique identifier. Provided by Runpay.
RP-Timestamp - The UNIX timestamp in milliseconds (https://en.wikipedia.org/wiki/Unix_time) **RP-Sign** - The HMACSHA250 hash (as a hex string) created by API Key secret and the message body concatenated with CompanyId and Timestamp.

EXAMPLE:
RP-Sign = ToHEX(HMACSHA512(“{REQUEST_BODY}{RP-CompanyId}{RP-Timestamp}”, Key)) ,where
ToHEX() - a function that represents byte array to a hexadecimal string
**HMACSHA512(**) - hash function (https://en.wikipedia.org/wiki/HMAC)
RP-CompanyId - the value of the RP-CompanyId request header
RP-Timestamp - the value of the RP-Timestamp request header
REQUEST_BODY - body of the request (https://en.wikipedia.org/wiki/HTTP_message_body) **Key** - Your private secret key provided by Runpay.

API methods

1. Account registration

Description: use this method for registration new user account.

URL: [BASE_URL]/account/registration **Request**

example:

```
{
    "clientId": "22222",
    "accountType": "person",
    "currency": "LYD",
    "endUserIP": "12.12.12.12",      "clientData":
    {
        "name": "John",
        "surname": "Wallet",
        "middlename": "Optional",
        "phone": "218941112222",
        "email": "email@email.com",
        "docType": "NID",
        "docNumber": "119752345678",
        "docImage1": "QeThYlndHdcwPRJdb4+7wWNmZdevItnJkSsj+tvJSOZwGzKWA==",
        "docImage2": "",
        "docImage3": "",
        "docImage4": "",
        "docImage5": ""
    }
}
```

Parameter	Description	Type
clientId	Client identifier.	string
accountType	Type of the account	enum [“person”, “entityt”]
currency	Currency of the account	string
endUserIP	IP address of the user that retrieves account registration	string
personalInfo	Personal information of the user	object
/name	Name (and Surname) of the user	string
/phone	Phone number of the user	string
/email	Email of the user	string
/docType	Document of the user	enum [NID, Passport, MF-Matricule fiscale]
/docNumber	Document number	string
/docImage1	Document image number 1	BASE64 string (size no more 500Kb)
/docImage2	Document image number 2	BASE64 string (size no more 500Kb)
/docImage3	Document image number 3	BASE64 string (size no more 500Kb)
/docImage4	Document image number 4	BASE64 string (size no more 500Kb)
/docImage5	Document image number 5	BASE64 string (size no more 500Kb)

Response example:

```
{
    "errorCode": 0,
    "errorText": "success",
    "status": "Registered"
}
```

Parameter	Description	Type
errorCode	Response result code.	int
errorText	Description of the result code.	string
status	Status of the registration	enum [“accepted”, “declined”, “registered”]

2. Account information

Description: use this method to retreive user account information.

URL: [BASE_URL]/account/info

Sign message: “{id}{companyId}{clientId}” **Request example:**

```
{
    "clientId": "22222"
}
```

Parameter	Description	Type
clientId	Client identifier.	string

Response example:

```
{
    "status": "registered",
    "account": {
        "number": "1205",
        "code": "75",
        "iban": "1200000000012",
        "currency": "LYD"
    },
    "maxLimit": 1000000.0,
    "outDailyLimit": 50000.0,
    "balance": 123.56,
    "createdUTC": "2021-08-10T19:22:00.2803+03:00",
    "lastIP": "127.0.0.1",
    "lastDateTimeUTC": "2021-08-09T19:22:00.2818656+03:00",
    "errorCode": 0,
    "errorText": "success"
}
```

Parameter	Description	Type
status	Status of the registration	enum [“accepted”, “declined”, “registered”]
account	Main account information	object
/number	Number of account	string
/code	Code of account	string
/iban	Bank number of account	string
/currency	Currency of account	string
maxLimit	Maximum limit of account balance	decimal
outDayLimit	Outgoing daily limit	decimal
balance	Account current balance	decimal
createdUTC	Date and time of creating account	datetime
lastIP	IP address of last operation	string

Parameter	Description	Type
lastDateTimeUTC	Date and time of last operation	datetime
errorCode	Response result code.	int
errorText	Description of the result code.	string

3. Account topup

Description: use this method to topup an account balance.

URL: [BASE_URL]/transaction/topup **Request**

example:

```
{
    "clientId": "22222",
    "amount": 50.23,
    "currency": "LYD",
    "endUserIP": "12.12.12.12",
}
```

Parameter	Description	Type
clientId	Client identifier.	string
amount	Amount of transfer	decimal
currency	Currency of transfer	string
endUserIP	IP address of the user that retrieves account registration	string

Response example:

```
{
    "errorCode": 0,
    "errorText": "success"
}
```

Parameter	Description	Type
errorCode	Response result code.	int
errorText	Description of the result code.	int

4. Do payment

Description: use this method to do payment transaction.

URL: [BASE_URL]/transaction/transfer **Request**

example:

```
{
        "companyTranId": "123abc",
        "clientId": "22222",
        "recipientId": 33333,
        "amount": 50.23,
        "currency": "LYD",
        "comment": "test",
        "endUserIP": "12.12.12.12"
}
```

Parameter	Description	Type
companyTranId	Company unique transaction idetifier	string
clientId	Client identifier	string
recipientId	Recipient identifier	string
amount	Amount of transfer	decimal
currency	Currency of transfer	string
endUserIP	IP address of the user that retrieves account registration	string
comment	transaction description	string

Response example:

```
{
  "operation": {
    "runpayTranId": 782,
    "emoneyTranId": 3539900,
    "companyTranId": 1652261199,
    "amount": 1.0,
    "currency": "LYD",
    "commission": 0.0,
    "dateTimeUTC": "2022-05-11T09:26:39.5215479Z",
    "comment": "transfer_transfer",
    "recipientId": "1652260540276",
    "clientId": "1652195124863",
    "tranTypeId": 2,
    "endUserIP": "189.56.12.44",
    "clientAccountId": 506,
    "recipientAccountId": 507
  },
  "errorCode": 0,
  "errorText": "success"
}
```

Parameter	Description	Type
operation	Details of operation	object
/runpayTranId	Operation identifier	long
/emoneyTranId	Emoney unique transaction idetifier	long
/companyTranId	Company unique transaction idetifier	long
/amount	Amount of operation	decimal
/currency	Currency of operation	string
/commission	Commission of operation	decimal
/dateTimeUTC	Date and time of operation	datetime

/comment	Comment of operation	string
Parameter	**Description**	**Type**
/recipientId	Recepient of operation	string
/clientId	Payer of operation	string
/tranTypeId	Is 2	int
/endUserIP	IP address of the user that retrieves account registration	string
/clientAccountId	Operation client's account	int
/recipientAccountId	Operation recipient's account	int
errorCode	Response result code.	int
errorText	Description of the result code.	string

5. Operations history

Description: use this method to retreive history of the account operations.

URL: [BASE_URL]/transaction/history **Request**

example:

```
{
    "clientId": "22222",
    "dateFrom": "2018-06-14T12:13:02",
    "dateTo": "2021-08-09T12:21:36"
}
```

Parameter	**Description**	**Type**
clientId	Client identifier	string
dateFrom	Period from	datetime
dateTo	Period to	datetime

Response example:

```
{
    "errorCode": 0,
    "errorText": "success",
"transactions":
    [
        {
            "id": 123,
            "companyTranId": "123abc",
            "amount": 555.13,
            "currency": "LYD",
            "commission": 2.4,
            "dateTime": "2021-07-10T19:39:30.4284049+03:00",
            "comment": "test",
            "clientId": "22222",
            "recipientId": 33333,
            "endUserIP": "12.12.12.12",
        },

        {
            "id": 123,
            "companyTranId": "123abc",
            "amount": 555.13,
            "currency": "LYD",
            "commission": 2.4,
            "dateTime": "2021-07-10T19:39:30.4284049+03:00",
            "comment": "test",
            "clientId": "22222",
            "recipientId": 33333,
            "endUserIP": "12.12.12.12",
        }
    ]
```

Parameter	Description	Type
errorCode	Response result code.	int
errorText	Description of the result code.	int
transactions	List of operations	object
/id	Operation identifier	long
/companyTranId	Company unique transaction idetifier	string
/amount	Amount of operation	decimal
/currency	Currency of operation	string
/commission	Commission of operation	decimal
/dateTime	Date and time of operation	datetime
/comment	Comment of operation	string
/recipientId	Recepient of operation	string
/clientId	Payer of operation	string
/endUserIP	IP address of the user that initiated operation	string

6. Create payment token

Description: use this method to generate payment QR-token.

URL: [BASE_URL]/token/create **Request**

example:

```
{
    "clientId": "22222",
    "maxAmount": 50.23,
    "currency": "LYD"
}
```

Parameter	Description	Type
clientId	Client identifier.	string
maxAmount	Maximum amount of transfer by token 0 - unlimited	decimal
currency	Currency of token	string

Response example:

```
{
    "errorCode": 0,
    "errorText": "success",
    "token": "3315409543133266457",
    "dateExpiredUTC": 300
}
```

Parameter	Description	Type
errorCode	Response result code.	int
errorText	Description of the result code.	int
token	Payment token	long
dateExpiredUTC	Token expired UTC date and time	datetime

7. Do payment with token

Description: use this method to do payment with QR-token.

URL: [BASE_URL]/token/withdraw **Request**

example:

```
{
    "companyTranId": "1050",
    "amount": 50.23,
    "currency": "LYD",
    "token": "3326285140063360701620",
    "toAccount": "10003453450004500004",
    "comment": "Rkeeper order 007888 KIPA MASALA 56 LYD"
}
```

Parameter	Description	Type
companyTranId	Unique identifier for transaction	long
amount	Amount of transfer	decimal
currency	Currency of transfer	string
token	Payment token	string
toAccount	Merchant's account to which the pay is made	string
comment	Operation comment	string

Response example:

```
{
    "errorCode": 0,
    "errorText": "success",
}
```

Parameter	Description	Type
errorCode	Response result code.	int
errorText	Description of the result code.	int

8. Get company info

Description: use this method to do payment with QR-token.

URL: [BASE_URL]/company/info **HTTP**
Method: GET

Response example:

```
{
    "errorCode": 0,
    "errorText": "success",
    "companyName": "Company name",
    "accountBalance": 125.21
}
```

Parameter	Description	Type
errorCode	Response result code.	int
errorText	Description of the result code.	int
companyName	Name of the company	string
accountBalance	Balance of the company account	int

Error codes

In order to check the status of the requested operation, you have to check *errorCode* attribute of the response. *errorCode = 0* indicates the success of the requested operation.

All error codes will be described soon…

Test environment

For testing, use this URLs:

Moldova - https://mddev.runpay.com:5443/EmbeddedWalletAPI/status

Tunis - https://tndev.runpay.com/EmbeddedWalletAPI/status

Postman collection - https://www.getpostman.com/collections/95bd8b94e1f0f74bea18

From:
https://wiki.runpay.com/ - **Wiki**

Permanent link:
https://wiki.runpay.com/doku.php?id=public:embeddedwalletapi_documentat

Last update: **2022/05/11 11:02**