



Fxbit Platform Interface Document

v1.0.3

1. Specification

1.1 Communication Protocol

HTTP/HTTPS

1.2 Request Method

All interfaces support only Post or Get requests.

1.3 Encoding Mode

All encoding formats use UTF-8, it includes the following terms:

- Http protocol commutation
- BASE64 encoding of the message

1.4 Message Specification

1. The message specification is divided into request message and response message.
 - The request message is initiated by the initiator.
 - The response message is responded by the receiver.

2. Interface Directory:

- 1、 Get the exchange rate

- 2、 Get transaction details
- 3、 Get wallet address
- 4、 Create a trading
- 5、 Round to find the transaction status

Interface List:

1、 Get the exchange rate

Interface function description:

Get the latest exchange rate

Request URL:

<https://merchant-api.fxbit.io/v1/api/getExchangeRate>

Request Mode:

GET

Request parameter format:

N/A

Response parameter format:

application/json

Request parameters:

N/A

Response parameters:

Parameter	Type	Remark
from_currency_code	Object	Currency before exchange
to_currency_code	Object	Currency after exchange
buyRate	String	Buying rate
sellRate	String	Selling rate
updated_timestamp	String	Exchange rate update time(timestamp)

Sample request:

```
curl -X GET "https://merchant-api.fxbit.io/v1/api/getExchangeRate" -H "accept: application/json" -H "x-api-key: HIX00hYqStkQ2G2N1D4JoNR1YLBLPF0DMYqt7uka"
```

Successful response sample:

```
{
  "BTC": {
    "HKD": {
      "updated_timestamp": "2019-04-24T06:33:01.000Z",
      "buyRate": "48337.3",
      "sellRate": "41745.85"
    },
    "JPY": {
      "updated_timestamp": "2019-04-24T06:33:01.000Z",
      "buyRate": "689032.3",
      "sellRate": "595073.35"
    },
    "CNY": {
      "updated_timestamp": "2019-04-24T06:33:01.000Z",
      "buyRate": "41445.8",
      "sellRate": "35794.1"
    },
    "USD": {
      "updated_timestamp": "2019-04-24T06:33:01.000Z",
      "buyRate": "5744.47925",
      "sellRate": "5464.26075"
    },
    "ETH": {
      "updated_timestamp": "2019-04-24T06:33:01.000Z",
      "buyRate": "36.23674186",
      "sellRate": "31.29536797"
    },
    "USDT": {
      "updated_timestamp": "2019-04-24T06:33:01.000Z",
      "buyRate": "6129.15807658",
      "sellRate": "5293.36379341"
    },
    "USDC": {
      "updated_timestamp": "2019-04-24T06:33:01.000Z",
      "buyRate": "6170.76180864",
      "sellRate": "5329.29428928"
    },
    "TUSD": {
      "updated_timestamp": "2019-04-24T06:33:01.000Z",
      "buyRate": "6164.19165032",

```

```
        "sellRate": "5323.62006164"
    },
    "PAX": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "6172.14678487",
        "sellRate": "5330.490405115"
    },
    "GUSD": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "6168.33959511",
        "sellRate": "5327.202377595"
    }
},
"ETH": {
    "HKD": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "1466.784",
        "sellRate": "1266.768"
    },
    "JPY": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "20914.355",
        "sellRate": "18062.3975"
    },
    "CNY": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "1257.674",
        "sellRate": "1086.173"
    },
    "USD": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "176.02245",
        "sellRate": "164.11755"
    },
    "USDT": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "186.05909916",
        "sellRate": "160.68740382"
    },
    "USDC": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "187.32236303",
        "sellRate": "161.778404435"
    },
    "TUSD": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
```

```

        "buyRate": "187.12288364",
        "sellRate": "161.60612678"
    },
    "PAX": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "187.32651014",
        "sellRate": "161.78198603"
    },
    "GUSD": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "187.21651586",
        "sellRate": "161.68699097"
    }
},
"USDT": {
    "HKD": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "8.668",
        "sellRate": "7.486"
    },
    "JPY": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "123.651",
        "sellRate": "106.7895"
    },
    "CNY": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "7.436",
        "sellRate": "6.422"
    },
    "USD": {
        "updated_timestamp": "2019-04-24T06:33:01.000Z",
        "buyRate": "1.02515",
        "sellRate": "0.99485"
    }
}
}

```

Failed response sample:

```

{
  "error": {
    "code": "API_KEY_MISSING",
    "message": "No api_key was supplied. Get one at https://demo-api.fx
bit.io"
  }
}

```

2、 Get transaction details

Interface function description:

Get transaction details

Request URL:

<https://merchant-api.fxbit.io/v1/api/getTransactionDetail>

Request Mode:

GET

Request parameter format:

N/A

Response parameter format:

application/json

Request parameters:

Parameters	Required field?	Type	Remark
receipt_token	Y	String	Transaction code

Response parameters:

Parameters	Type	Remark
error	Number	0 - There is no error
receipt_token	String	Transaction code (Corresponding to the Payment ID)
shop_name	String	Name of shop
marked_currency	String	Transaction currency
marked_price	Number	Transaction price
billed_currency	String	The billing currency
billed_price	Number	The bill price
received_amount	Number	Amount received
payment_status	Number	Transaction status: 0:new, 1:confirming, 2:confirmed, 3:failed, 4:insufficient, 5:expired, 6: Incorrect_amount
net	Number	Net gain
fee	Number	Procedure fee

Sample request:

```
curl -X GET "https://merchant-api.fxbit.io/v1/api/getTransactionDetail?receipt_token=1552455439430" -H "accept: application/json" -H "x-api-key: HIX00hYqStkQ2G2N1D4JoNR1YLBPF0DMYqt7uka"
```

Successful response sample:

```
{
  "error": 0,
  "receipt_token": "1552455439430",
  "shop_name": "Yuen_BitPay",
  "marked_currency": "ETH",
  "marked_price": 0.05,
  "billed_currency": "ETH",
  "billed_price": 0.05,
  "received_amount": 0.05,
  "payment_status": 2,
  "net":0.04,
  "fee":0.01
}
```

Failed response sample:

```
{
  "error": 1
}
```

3、 Get wallet address

Interface function description:

In response to the transaction, get the wallet address and payment amount.

Request URL:

<https://merchant-api.fxbit.io/v1/api/getWalletAddress>

Request Mode:

POST

Request parameter format:

application/json

Response parameter format:

application/json

Request parameters:

Parameters	Required field?	Type	Remark
receipt_token	Y	String	Transaction code

billed_currency	Y	String	The billing currency
-----------------	---	--------	----------------------

Response parameters:

Parameters	Type	Remark
public_address	String	Payment address for cryptocurrency
billed_price	Number	The bill price
index	Number	The index

Sample request:

```
curl -X POST "https://merchant-api.fxbit.io/v1/api/getWalletAddress" -H
"accept: application/json" -H "x-api-key: HIX00hYqStkQ2G2N1D4JoNR1YLBL
PF0DMYqt7uka" -H "Content-Type: application/json" -d "{ \"receipt_token
\": \"1552483134660\", \"billed_currency\": \"BTC\"}"
```

Successful response sample:

```
{
  "public_address": "1CjoVrwpNBLMCdvYAAAYEU4keahJLsiY1C7",
  "billed_price": "0.02671",
  "index": 99
}
```

Failed response sample:

```
{
  "error": 2,
  "error_message": "Shop or receipt does not exist."
}
```

4、 Create a trading

Interface function description:

Create a new trading (No currency has been chosen)

Request URL:

https://merchant-api.fxbit.io/v1/api/getReceiptToken

Request Mode:

POST

Request parameter format:

application/json

Response parameter format:

application/json

Request parameters:

Parameters	Required Field?	Type	Remark
marked_currency	Y	String	The billing currency
marked_price	Y	Number	The bill price
remark	Y	String	The remark
callback_url	Y	String	The callback address for notification

Response parameters:

Parameters	Type	Remark
error	Number	0 - There is no error
shop_identifier	String	The identifier of shop
receipt_token	String	Transaction code (Corresponding to the Payment ID)

Sample request:

```
curl -X POST "https://merchant-api.fxbit.io/v1/api/getReceiptToken" -H "accept: application/json" -H "x-api-key: HIX00hYqStkQ2G2N1D4JoNR1YLB L P F0DMYqt7uka" -H "Content-Type: application/json" -d "{ \"marked_currency\": \"USD\", \"marked_price\": \"100\", \"remark\": \"remark\", \"callback_url\": \"https://example.com/callback\"}"
```

Successful response sample:

```
{  "error": 0,  "receipt_token": 1552483134660,  "shop_identifier": "39cf4fb7d461d2fd73a4baf6585b7c21"}
```

Failed response sample:

```
{  "err": "Invalid Currency"}
```

Callback response parameters:

Parameters	Type	Remark
receipt_token	String	Transaction code (Corresponding to the Payment ID)
billed_price	Number	The bill price
received_amount	Number	Amount received

status	Number	Transaction status: 0:new, 1:confirming, 2:confirmed, 3:failed, 4:insufficient, 5:expired, 6: Incorrect_amount
signature	String	

Callback response sample:

```
{
  "receipt_token": "1552455439430",
  "billed_price": 0.05,
  "received_amount": 0.05,
  "status": 2
}
```

Signature:

Signature mechanism

In order to ensure the authenticity and integrity of the data transmission process, the callback data is returned to the signature data, and the merchants verify after receiving the callback data.

The signature is generated by splicing the values according to the ascending order of key parameter letters. Finally, the original string is generated by adding the merchant API Key, and the signature result is calculated according to the sha265 encryption method.

Unsigned string

1. Except for the signature field, all parameters are spliced in ascending order of key letters.
2. In the string to be signed, the original values are used for the field values.

Example

As the field signature in the callback response example mentioned above, the fields of JSON data in the above-mentioned list are first arranged in ascending order of keys and then stitched together. Finally, the merchant API Key is added to get the following result of the string to be signed.

Billed_price = 0.05 & receipt_token = 1552455439430 & received_amount = 0.05 & status = 2 & your_api_key

Then the above signature string is encrypted by sha265 algorithm and the signature data is finally obtained.

The following nodejs signature code:

```
Signature = crypto. createHash ('sha256'). update ('billed_price = 0.05  
& receipt_token = 1552455439430 & received_amount = 0.05 & status = 2  
& your_api_key'). digest ('hex');
```

Verify signatures in response to data

When the callback response data is obtained, the parameters need to be signed to obtain 64-bit signature data. You need to verify that the signature data is identical to the value in the signature field of the callback data result to determine whether the signature is validated or not.

5、 Round to find the transaction status

Interface function description:

Round to find the transaction status

Request URL:

<https://merchant-api.fxbit.io/v1/api/pollPaymentStatus>

Request Mode:

POST

Request parameter format:

application/json

Response parameter format:

application/json

Request parameters:

Parameters	Required field?	Type	Remark
address	Y	String	Transaction currency address

Response parameters:

Parameters	Type	Remark
status	Number	Transaction status: 0: There are no entries 1: Monitoring money into this address

Sample request:

```
curl -X POST "https://merchant-api.fxbit.io/v1/api/pollPaymentStatus" -H "accept: application/json" -H "x-api-key: HIX00hYqStkQ2G2N1D4JoNR1YLB LPF0DMYqt7uka" -H "Content-Type: application/json" -d "{\"address\": \"1CjoVrwpNBLMCdvYAAYEU4keahJLsiY1C7\"}"
```

Successful response sample:

```
{  
  "status": 1  
}
```

Failed response sample:

```
{  
  "status": 0  
}
```

3 Appendix A - Response code specification

Response Code	Remark
200	Successful
301	Parse message error
302	Invalid call credentials
303	Incorrect parameter
500	Internal system error

4 Appendix B - Currency

Currency	Remark
CNY	China Yuan
HKD	Hong Kong dollar
JPY	JAPANESE YEN
TWD	New Taiwan Currency

USD	American dollar
-----	-----------------