



LIPAPAY

LIPAPAY TRANSACTION API

Overview

This describes the resource that make up the LipaPay Transaction API v1.4.

The API access is over HTTPS, and accessed from the <https://www.lipapay.com/api/excashier.html>

Interface content

Send Message Endpoint

Product ENV

```
https://www.lipapay.com/api/excashier.html
```

Sandbox ENV

```
Order checkout url: https://sandbox.lipapay.com/api/excashier.html
```

```
Merchant url: https://sandbox.lipapay.com
```

```
Merchant id: test
```

Password: 1234567890

Key: Gw416RCMO8tD5MSUg5dok5uQGvR3rPpx

Request Method

A request to the checkout API should carry the following parameters with a Content-Type header `application/x-www-form-urlencoded` using **POST** method.

Request Parameters

Field Name	Type	Required	Description
version	string	Y	Interface version 1.4, Not in the signature field
merchantId	string	Y	It's created by LipaPay, Please contact via info@lipapay.com
signType	string	Y	Sign Type. (Supports only MD5)
sign	string	Y	See "Signature Rule"
notifyUrl	string	Y	A URL of merchant system that can accepts notifications from LipaPay system.
returnUrl	string	Y	Jump address of payment result, But Payment Method is 'OL'. This field is required.
merchantOrderNo	string	Y	The merchant platform order ID, It must be unique. And the length must less than 50.

amount	number	Y	The amount of transaction (unit cent).
sellerId	string	N	Seller User ID
sellerAccount	string	N	Seller User Name
buyerId	string	N	Buyer User ID
buyerAccount	string	N	Buyer User Name
goods[n].goodsId	string	N	Goods ID (max length 32)
goods[n].goodsName	string	Y	Goods Name (max length 60)
goods[n].goodsQuantity	string	Y	Goods Quantity (positive whole number)
goods[n].goodsPrice	string	Y	Goods Price (unit cent)
goods[n].goodsInfo	string	N	Goods Info (max length 2000)
goods[n].goodsType	string	Y	Goods Type {1 is physical goods, 2 is virtual goods}
goods[n].goodsUrl	string	N	Goods Info URL

customerIP	string	N	Customer's IP Address
expirationTime	string	Y	Expiry Time
sourceType	string	Y	Terminal source, APP:A, Browser: B
channels	string	N	Limit the channel that the payer can use, please contact our business manager if you need to add the designated channel. Do not add to it at will
paymentMethod	string	N	OL: online payment, OF: offline payment
email	string	N	Buyer's email
mobile	string	N	Buyer's mobile phone number
currency	string	Y	Currency defined by ISO 4217
p1	string	N	Customer's parameter (max length 200)
p2	string	N	Customer's parameter (max length 200)
p3	string	N	Customer's parameter (max length 200)

Examples:

```
# curl
curl -X POST 'https://sandbox.lipapay.com/api/excashier.html' \
-d 'version=1.4' \
```

```
-d 'merchantId=your merchant id' \  
  
-d 'signType=MD5' \  
  
-d 'sign=sign string' \  
  
-d 'notifyUrl=https://www.xxxx.com' \  
  
-d 'returnUrl=https://www.xxxx.com' \  
  
-d 'merchantOrderNo=your order No.' \  
  
-d 'amount=87500' \  
  
-d 'sellerId=your seller id' \  
  
-d 'sellerAccount=your seller account' \  
  
-d 'buyerId=your buyer id' \  
  
-d 'buyerAccount=your buyer account' \  
  
-d 'goods[0].goodsId=1801191319467600584' \  
  
-d 'goods[0].goodsName=goodsName1' \  
  
-d 'goods[0].goodsQuantity=20' \  
  
-d 'goods[0].goodsPrice=3250' \  
  
-d 'goods[0].goodsInfo=goodsInfo1' \  
  
-d 'goods[0].goodsType=1' \  
  
-d 'goods[0].goodsUrl=http://www.xxxx.com' \  
  
-d 'goods[1].goodsId=1801191319467600585' \  
  
-d 'goods[1].goodsName=goodsName2' \  
  
-d 'goods[1].goodsQuantity=10' \  
  
-d 'goods[1].goodsPrice=2250' \  
  
-d 'goods[1].goodsInfo=goodsInfo2' \  
  
-d 'goods[1].goodsType=2' \  

```

```

-d 'goods[1].goodsUrl=http://www.xxxx.com' \
-d 'customerIP=10.0.0.140' \
-d 'expirationTime=1000000' \
-d 'sourceType=B' \
-d 'currency=KES' \
-d 'channels=' \
-d 'paymentMethod=' \
-d 'email=customer@xxx.com' \
-d 'mobile=your customer phone No.'

```

Synchro Response

If payment Method is 'OF', LipaPay transaction api will return the payment code related information synchronously.

Response Data Type

application/json

Response Parameters

Field Name	Type	IsNull	Description
bizContent	string	N	
channels	list	N	Payment Channels
orgCode	string	N	Payment Organization Code

payBill	string	N	Pay Bill Number
paySN	string	N	Payment Account Number
errorCode	string	Y	
merchantId	string	Y	
merchantOrderNo	string	Y	
orderId	string	Y	LipaPay SN.
paymentMethod	string	Y	Payment Method
sign	string	Y	Sign Content
signType	string	Y	Sign Type,(Supports only MD5)
status	string	Y	SUCCESS

Examples:

```
{
  "bizContent":
    {"channels":
      [
        {"orgCode":"MPESA","payBill":"XXX","paySN":"643BUQKK"},
        {"orgCode":"MTN","payBill":"XXX","paySN":"643BUQKK"},
      ]
    }
}
```

```

    {"orgCode":"Airtel","payBill":"XXX","paySN":"643BUQKK"},

    {"orgCode":"Diamond","payBill":"XXX","paySN":"643BUQKK" }

]

},

"errorCode":"00000",

"merchantId":"XXXXXX",

"merchantOrderNo":"XXXXXX",

"orderId":"XXXXXX",

"paymentMethod":"OF",

"sign":"XXXXXXXX",

"signType":"MD5",

"status":"SUCCESS"

}

```

Page Callback Endpoint

After the user clicks the Lipapay page button, the parameters returned by the page call-back address.

Field Name	Type	Required	Description
merchantId	string	Y	Merchant ID
merchantOrderNo	string	Y	The merchant platform order number
orderId	string	Y	LipaPay SN.

status	enum	Y	The status {SUCCESS,UNKNOWN}
p1	string	N	Customer's p1 parameter
p2	string	N	Customer's p2 parameter
p3	string	N	Customer's p3 parameter
signType	enum	Y	Sign Type,(Supports only MD5)
sign	string	Y	Sign Content. See "Signature Rule"

Examples:

`https://merchant-return-url/xxx?merchantId=xxx&merchantOrderNo=xxx&orderId=xxx&status=xxx&signType=xxx&sign=xxx`

Service Call-back Endpoint

If buyer paid successful, LipaPay service will notify the 'notifyUrl' of the payment results.

Request Parameters

Field Name	Type	Required	Description
amount	string	Y	Order Amount
merchantId	string	Y	Merchant ID

merchantOrderNo	string	Y	Merchant Order No.
orderId	string	Y	LipaPay SN.
orgTransId	string	Y	Transaction ID from bank
paymentChannel	string	Y	Payment Channel
paymentMethod	string	Y	Payment Method
sign	string	Y	Sign Content. See "Signature Rule"
signType	string	Y	Sign Type,(Supports only MD5)
status	string	Y	The transaction status
p1	string	N	Customer's p1 parameter
p2	string	N	Customer's p2 parameter
p3	string	N	Customer's p3 parameter

Examples:

POST <https://merchant-notify-url/xxx>

```
amount=1000000&merchantId=20160511120&merchantOrderNo=PQOISCBGZBD864KO&orderId=K170503120044078901
3&orgTransId=1493784054&paymentChannel=MPESA&paymentMethod=OL&sign=d8b53e07a184f3968110de63f747f48f&
signType=MD5&status=SUCCESS
```

Response Parameters

Merchant system should respond message with the following format.

If LipaPay cannot get the response from merchant or received failed message, the notification message will be dropped after the number of retries is reached.

Field Name	Type	Required	Description
status	string	Y	Status {SUCCESS, FAILURE}
errorCode	string	Y	100 is SUCCESS others is FAILURE
merchantId	string	Y	Merchant ID
signType	string	Y	Sign Type,(Supports only MD5)
merchantOrderNo	string	Y	Merchant Order No.
orderId	string	Y	LipaPay SN.
sign	string	Y	Sign Content. See "Signature Rule"

Examples:

```
{  
  
  "status": "SUCCESS",  
  
  "errorCode": "100",  
  
  "merchantId": "test",  
  
  "signType": "MD5"  
  
  "merchantOrderNo": "f087786a-3f4d-c3ae-2cd5-59a7dbcedb84",
```

```
"orderId": "K1708310947491101622",  
  
"sign": "c5a08e4e3c612bfcf01d240e0c29810f"  
  
}
```

Signature Rule

All fields in the message will be protected by digital signature, the "sign" field will be obeyed the following rules.

1. Collect and Sort: collect all message fields excluding the empty value fields and the "sign" field, and sort them by field name (key) with ASCII ascending order. Examples:

```
key1=value1  
  
key2=value2  
  
.....
```

2. Concat: Concat all ordered fields in the format like "k1=v1&k2=v2" as the plaintext variable. Examples:

```
var plainText = "k1=v1&k2=v2"
```

3. Signature: Append the signKey to the tail of plainText, then do signature using MD5, and output it in hex format. Examples:

```
plainText = plainText + "signKey"  
  
var sign = md5(plainText) // e.g. ec33eed433e570c11fc19e9a07c8f1eb
```

Error Codes

There is some details of error codes in "Transaction Checkout API".

Code	Description Code	Description
00000	SUCCESS	Successful
200100	REQUEST_IS_NULL	Request parameters is null
200200	MERCHANT_ID_IS_NULL	Merchant ID is null
200300	MERCHANT_ORDERNO_IS_NULL	Merchant Order No. is null
200401	AMOUNT_FORMAT_ERROR	Amount Format Error
200402	AMOUNT_LENGTH_ERROR	Amount Length Error
200404	CURRENCY_IS_NULL	Currency Code is null
200405	NOT_SUPPORT_CURRENCY	Currency Code not support
200704	NOTIFY_URL_LENGTH_ERROR	Notification URL Length Error
200800	PAGE_URL_IS_NULL	Page URL is null
200801	PAGE_URL_FORMAT_ERROR	Page URL Format Error
200900	SERVER_URL_IS_NULL	Server URL is null

200901	SERVER_URL_FORMAT_ERROR	Server URL format error
201000	SIGN_IS_NULL	Sign Content IS NULL
201001	VALIDATE_SIGN_ERROR	Validate sign error
201002	SIGN_ERROR	Sign Error
201100	SIGN_TYPE_IS_NULL	Sign Type is null
201101	SIGN_TYPE_ERROR	Sign Type Error
201201	IP_FORMAT_ERROR	IP Address format error
201400	EXPIRE_TIME_IS_NULL	Expire time is null
201401	EXPIRE_TIME_FORMAT_ERROR	Expire time format error
201402	EXPIRE_TIME_LENGTH_ERROR	Expire time length error
201403	EXPIRE_TIME_ERROR	Expire time error
201500	SOURCE_TYPE_IS_NULL	Source type is null
201501	SOURCE_TYPE_FORMAT_ERROR	Source type format error
201601	PAY_METHOD_FORMAT_ERROR	Payment method format error

201702	SELLER_ID_LENGTH_ERROR	Seller ID length error
201802	SELLER_ACCOUNT_LENGTH_ERROR	Seller account length error
201902	BUYER_ID_LENGTH_ERROR	Buyer ID length error
202002	BUYER_ACCOUNT_LENGTH_ERROR	Buyer Account length error
202102	GOODS_NAME_LENGTH_ERROR	Goods name length error
202201	GOODS_TYPE_FORMAT_ERROR	Goods type format error
202301	GOOD_URL_FORMAT_ERROR	Goods URL format error
202302	GOOD_URL_LENGTH_ERROR	Goods URL length error
202602	GOODS_INFO_LENGTH_ERROR	Goods info length error
202700	VERSION_ERROR	Version is error
202600	EXTEND_PARAM_LENGTH_ERROR	Extend param length error
203101	GOODS_QUANTITY_FORMAT_ERROR	Goods quantity format error
203201	GOODS_PRICE_FORMAT_ERROR	Goods price format error