In-Store API - QR Code Picture

[To check HTML verion online](#)

This API will return `QR code picture` and order information in [#Payment Interface](#Payment-Interface), perfect for EFTPOS. This document is only suitable for the following `in-store` situation.

- The customer loads E-commerce website with a `mobile browser` and pays through `Alipay app`.
- The customer loads E-commerce website with `Alipay app's browser` and pays through `Alipay app`.
- The customer scans `Alipay payment QR code` through `Alipay app`.
- The customer scans `Wechat payment QR code` through `Wechat app`.

How it works?
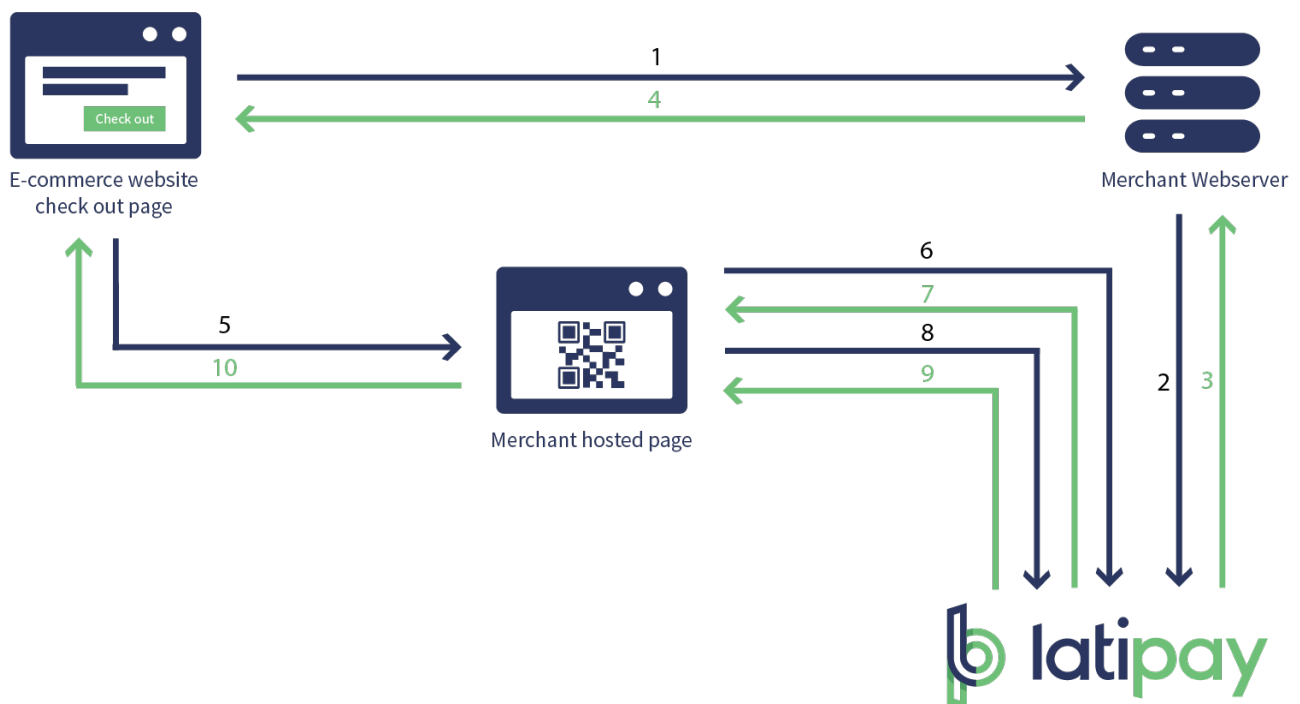
1. To process a transaction, customers select Latipay Wechat pay or Alipay as a payment method.

2. **Merchant** request to **Latipay** [#Transaction Interface](#) with the authentication and order information.

3. **Latipay** responds to **Merchant** with a `host_url` and `nonce`.

4. **Merchant** responds with a unique URI for an SSL secure payments page.

5. The shopping cart uses the returned URI to redirect the customer to the secure **Merchant** hosted payments page.

6. **Merchant** uses the `host_url` and `nonce` to request a QR code. #Payment Interface.

7. **Latipay** responds a QR code Url, then the QR code will display in **Merchant** hosted page. The customer will be prompted to `scan the QR code` and complete the payment.

8. **Merchant** query the payment status from **Latipay** #Payment Result Interface.

9. **Latipay** updates the payment status.

10. The payment result is displayed and the browser is redirected back to the website.
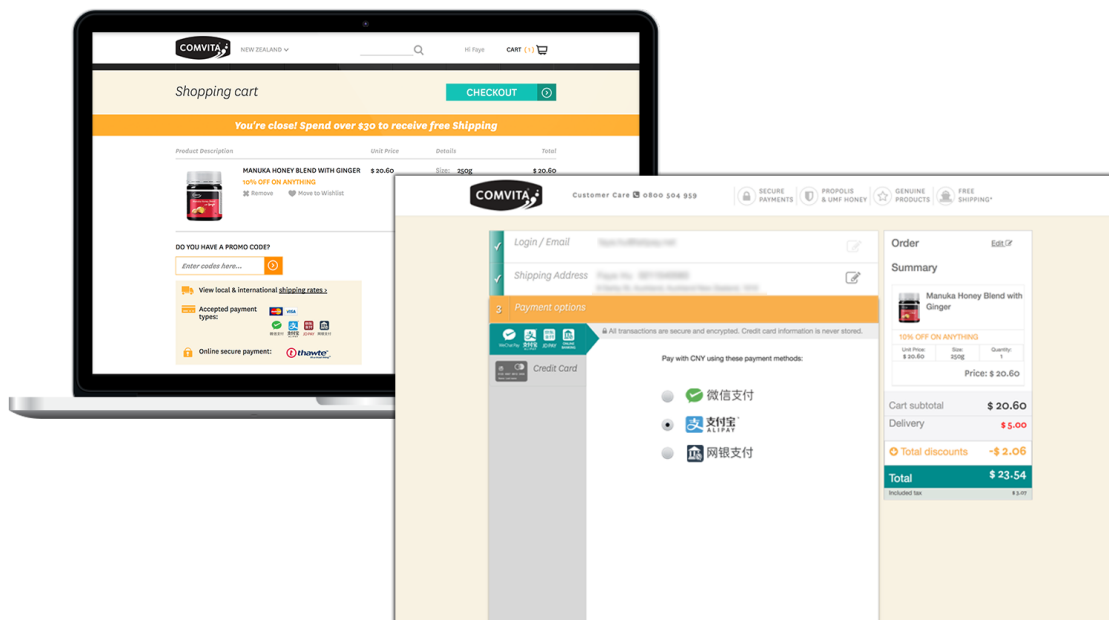
Payment Scenarios

Alipay
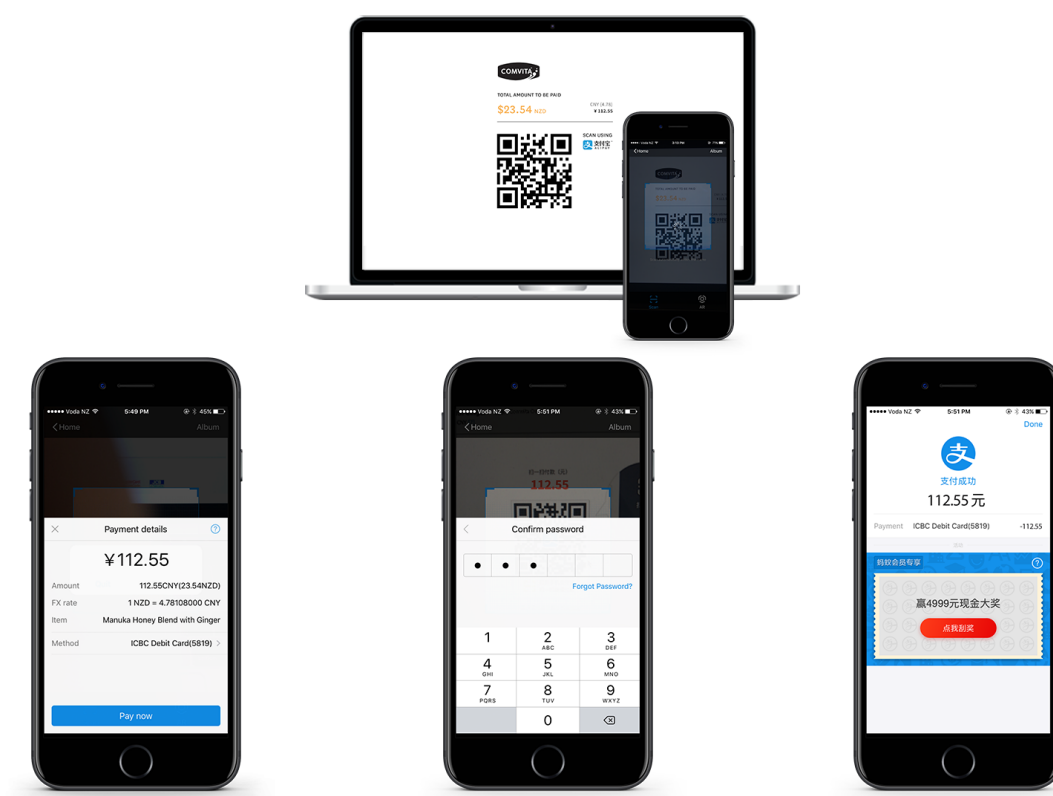
WeChat Pay
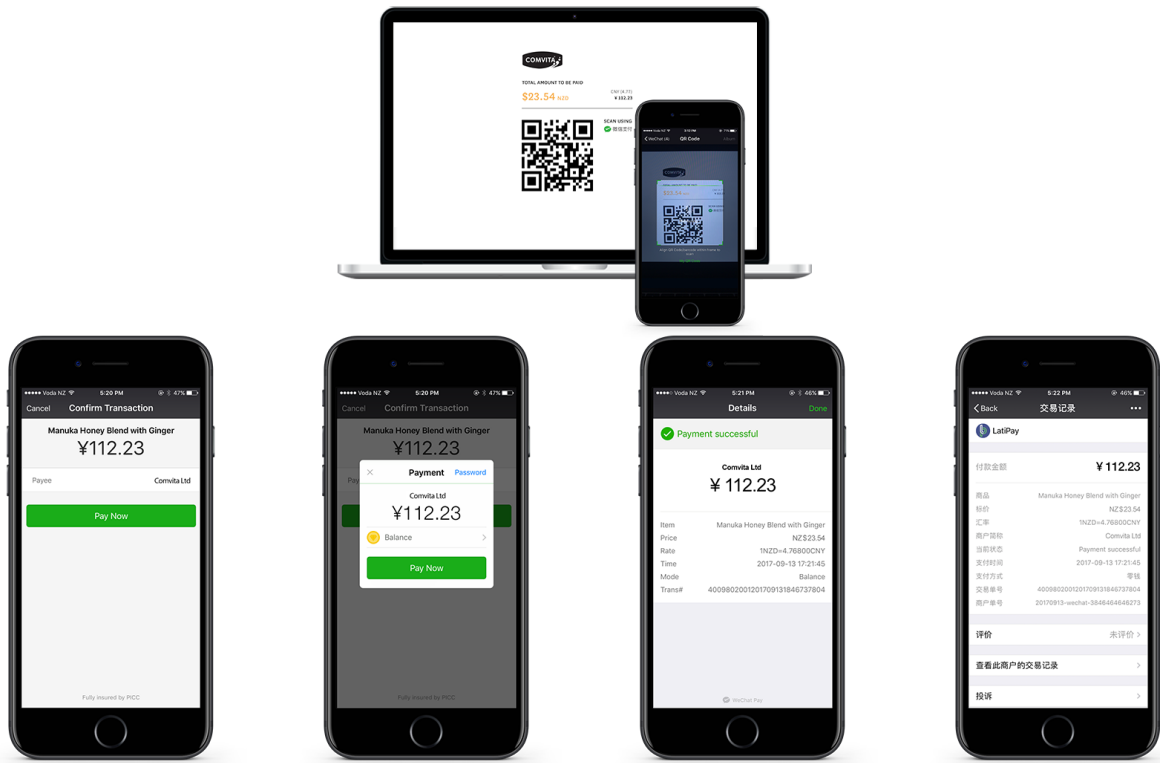
Website check-out page

API List

Preparation

Before using the following API, please make sure you have `user_id`, `wallet_id` and `api_key` on hand. If you don't have them, we would like to help you. Contact Us.

Transaction Interface

Create a latipay transaction is the first step for using alipay or wechat pay.

```
POST https://api.latipay.net/v2/transaction
```

Parameters:

Name	Type	Description	Nullable
user_id	String	The user account you want to use to process the transaction.	No
wallet_id	String	The ID of the wallet you want to use.	No
amount	String	A decimal amount.	No
merchant_reference	String	A field for identifying your transaction.	No
payment_method	String	Payment method options are `alipay` and `wechat`.	No
return_url	String	The URL of the landing page where Latipay will return the customer after payment, but it doesn't work in `in-store` situation.	No
callback_url	String	The URL of the callback address the transaction notification is sent after payment.	No
ip	String	The IP address of the customer.	No
version	String	The latest version of the platform. must be `"2.0"`	No
product_name	String	The name of the product or service being sold.	No
host_type	String	must be `"1"`, for #Payment Interface to return `QR code picture` in base64	No
signature	String	The SHA-256 HMAC API signature.	No

Example Parameters

```
{
  "signature": "14d5b06a2a5a2ec509a148277ed4cbeb3c43301b239f080a3467ff0aba4070e3",
  "wallet_id": "W00000001",
  "amount": "120.00",
  "user_id": "U000334333",
  "merchant_reference": "dsi39ej430sks03",
  "return_url": "https://merchantsite.com/checkout",
  "callback_url": "https://merchantsite.com/confirm",
  "ip": "122.122.122.1",
  "version": "2.0",
  "product_name": "Pinot Noir, Otago",
  "payment_method": "alipay",
  "host_type" : "1"
}
```

SHA-256 HMAC Signature

```
message: user_id + wallet_id + amount + payment_method + return_url + callback_url
secret: api_key
```

Example Signature **Try your signature online**

```
message: U000000001W000000010.01alipayhttp://merchant.com/returnhttp://merchant.com/callback
secret: 111222333

signature: 2367bcd9e9a2f9a547c85d7545d1217702a574b8084bbb7ae33b45a03a89983
```

Response

```
{
  "host_url": "https://api.latipay.net/merchanthosted/gatewaydata",
  "nonce": "7d5a88119354301ad3fc250404493bd27abf4467283a061d1ed11860a46e1bf3"
  "signature": "14d5b06a2a5a2ec509a148277ed4cbeb3c43301b239f080a3467ff0aba4070e3",
}
```

Name	Type	Description
host_url	String	The url for request a QR code picture. The nonce should be appended to the end.
nonce	String	The transaction nonce must be appended to the host_url URL.
signature	String	The SHA-256 HMAC API signature.

Signature in Response

For security reasons, we highly recommend you verify the signature in the response.

```
message: nonce + host_url
secret: api_key
```

Payment Interface

```
POST {host_url}/{nonce}
```

Example

```
https://api.latipay.net/merchanthosted/gatewaydata/7d5a88119354301ad3fc250404493bd
27abf4467283a061d1ed11860a46e1bf3
```

Response

Name	Type	Description
code	Integer	The response code of payment, `0` or [Error Code](#), 0 means no error happened.
message	String	The response message of payment interface.
data	Object	Contains `QR code picture` and order information.

the `data` object

Name	Type	Description
order_id	String	A unique transaction identifier generated by Latipay.
nonce	String	A unique transaction nonce generated by Latipay.
payment_method	String	Payment method options are alipay and wechat.
amount	String	A decimal amount.
amount_cny	String	A decimal amount.
currency	String	The currency code of the transaction.
product_name	String	The name of the product or service being sold.
organisation_id	Integer	The merchant id which is used to process the transaction.
org_name	String	The merchant's name
user_id	String	The user account which is used to process the transaction.
user_name	String	The user account's name
wallet_id	String	The user wallet account which is used to process the transaction.
wallet_name	String	The wallet account's name
qr_code	String	The QR code picture
qr*code*url	String	The QR code text version, you can generate QR code picture with it.
currency_rate	String	The rate which is used to the transaction.
merchant_reference	String	A field for identifying your transaction.
signature	String	The `SHA-256 HMAC` API signature.

Example

```json
{
    "code": 0,
    "message": "SUCCESS",
    "data": {
        "order_id": "20180112000003",
        "nonce": "fd270820180112103849a19380c7300c425898780b3d0b1fbc",
        "payment_method": "wechat",
        "amount": 0.01,
        "amount_cny": 0.05,
        "currency": "AUD",
        "product_name": "产品中文",
        "organisation_id": 695,
        "organisation_name": "126Emai",
        "user_id": "U000000266",
        "user_name": "04_oliver",
        "wallet_id": "W000000329",
        "wallet_name": "AUD_01",
        "qr_code": "data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAMgAAADICAYAAACt
WK6eAAAEzUlEQVR42u3dS27cQAwE0Ln/pZMTGDBgDVlFPQLZaWBJzadFpT+ff0qpH+vjFSgFiFKAKAWIUo
AoBYhS3UA+n0/Mv18/0C9/2/hsf3nev7yDrd8mvHtAAAEEEAAAAQQQQAABJBoICPJwR9e9MSAPP1sE8+7
dS+NfQUIIIAAgggAAFIN5Olm27ru6SZPSrEm7iUpaQQEEEAAAQQQQAABBBBBAAAEkAMiV6SJJCV
06akAAAQQQAABBBBBBAAAEEEECKUqzGaSAT0za2ki1AAAEEEEAAAQQQQAABBBBBAvgjkbS9w4u+mY7UeBBBBA
AAEEEEEAAAQQQQAABJABI0lQO1925ztajgLgOEEBcBwgrgMEEcB8uIzCtPXYEw821bKVtUngAACCCAAA
IIIIAAgggbUAm1iRcnlay1ZTpiWTChwgQQABBBBBAAEEEEEAASQaSORBhO/3RrM9CMWJhK6b94LIIAA
AggggAACCCCA1AC5stlB+ll8SSnblaTsTMwLCCCAAAIIIIAAAggggADSkjCl7/jXCH0rGdxCAwgggAACJLY0Fs79F1pyq0mb0/UAAEEEE
AAAQQQQAABBBBBAXgVkK4naevlX1mUkJXTfvD9AAAEEEEAAAQQQQAABBJAYIIYIeTisvJVuN7OXM+CCCAAAII
IIIIAAAggggAACyCSQLUjjpzTa9Y9bcB24yc070TyBggggAACCCCAAAIIIAAEgMaeAaaz/tLeo705K0yxQIEEEE
AAAQQQQAABBBBBAAJkEkv7yGG481SGqE9F0jE5I8QAABBBBBAAAAAEEEEEAAAAEkBsjw/B7FxQLUn6zooTfQ
oIIIAAggggAACCCAABIJ5PIBkw/B7FxLUn6zooTfQ
oIIIAAgggggAACCCAABIJ5PIajK2kDLiuMQIEEGMECCDGCBBBBAAAAEEEEDqUqytShg9CZKnxoy8YEGBBBBA
AAEEEEEAAaQOSOOZgt98qZNNNnnv7RSf+IAQIIIIIIAAggggAACCCAAFIEZGKaReOUmfRj9bOKAQE
EEAAAQQQQAABBBBBAAHkrkMYzBSfSss63pIlIsQAABBBBBAAAAAaQSQACATDdiYuDQmOEnnTQICCCCAAA
IIIIAAgggggAAynEpc3lBhKxncSveSrGn7gUQQQAABBBBBAAAAAEEECAAASQGiFJvL0CUAkQpPQCRClAlAlAJE
KUCUqq7/Vhpz81YibhcAAAASUVORK5CYII=",
        "qr_code_url": "weixin://wxpay/bizpayurl?pr=ZObbdsW",
        "currency_rate": "5.13710",
        "merchant_reference": "",
        "signature": "cf56a6bbe1aa42e1c273c4d7fe6233f70a6b10ba39e88938b13469f3315e
4c41"
    }
}
```

SHA-256 HMAC Signature

Rearrange parameters in the `data` alphabetically (except `signature` and other parameters with value of null or empty string) and join rearranged parameters with `&`

```
message: amount=1.00&amount_cny=5.00&currency=NZD&currency_rate=5.29930&merchant_r
eference=dsi39ej430sks03&nonce=7d5a88119354301ad3fc250404493bd27abf4467283a061d1ed
11860a46e1bf3&order_id=20170829-alipay-3990527237343&organisation_id=18&org_name=L
atipay&payment_method=alipay&product_name=test&qr_code=https://qr.alipay.com/bax03
286h4vlfpxldgwq4035&type=Online&user_id=U000000051&wallet_id=W000000037&wallet_nam
e=aud01
secret: api_key
```

Payment Result Asynchronous Notification

This is a payment result notification sent by Latipay to merchants after the payment is done successfully. There is a re-try mechanism with the notification to ensure the notification could be delivered to the merchant.

```
POST merchant's callback_url
```

Parameters:

Name	Type	Description
transaction_id	String	A unique transaction identifier generated by Latipay.
merchant_reference	String	A field for identifying your transaction.
currency	String	The currency code of the transaction.
amount	String	A decimal amount.
payment_method	String	The payment method used. Possible values are alipay and wechat.
status	String	The status of the transaction. Possible values are: pending, paid, or failed.
pay_time	String	Show the payment time of the transaction order.
signature	String	The `SHA-256 HMAC` API signature.

Example input

```
{
  "signature": "14d5b06a2a5a2ec509a148277ed4cbeb3c43301b239f080a3467ff0aba4070e3",
  "transaction_id": "20170707-wechat-3473511594933",
  "merchant_reference": "dsi39ej430sks03",
  "currency": "AUD",
  "amount": "120.00",
  "payment_method": "wechat",
  "status": "paid",
  "pay_time": "2017-07-07 10:53:50"
}
```

SHA-256 HMAC Signature

Merchant backend needs to validate the signature for protecting against malicious requests.

```
message: merchant_reference + payment_method + status + currency + amount
secret: api_key
```

Expected the text "sent" in Response's body

```
sent
```

Payment Result Interface

All customers can send requests to query payment status with merchant order id(that should be unique id for the merchant) as merchant_reference by HTTP GET request.

```
GET https://api.latipay.net/v2/transaction/{merchant_reference}
```

Parameters

Name	Type	Description
user_id	String	The user account you want to use to process the transaction.
signature	String	The `SHA-256 HMAC` API signature.

SHA-256 HMAC Signature [Try your signature online](#)

```
message: merchant_reference + user_id
secret: api_key
```

Example

```
GET https://api.latipay.net/v2/transaction/1289323A122DB?user_id=U000334333&signat
ure=14d5b06a2a5a2ec509a148277ed4cbeb3c43301b239f080a3467ff0aba4070e3
```

Response

Name	Type	Description
transaction_id	String	A unique transaction identifier generated by Latipay.
merchant_reference	String	A field for identifying your transaction.
currency	String	The currency code of the transaction.
amount	String	A decimal amount.
payment_method	String	The payment method used. Possible values are alipay and wechat.
status	String	The status of the transaction. Possible values are: pending, paid, or failed.
pay_time	String	Show the payment time of the transaction order. `UTC/GMT+08:00`
signature	String	The `SHA-256 HMAC` API signature.

Example Response

```
{
  "transaction_id": "20170707-wechat-3473511594933",
  "merchant_reference": "dsi39ej430sks03",
  "currency": "AUD",
  "amount": "120.00",
  "payment_method": "wechat",
  "status": "paid",
  "pay_time": "2017-07-07 10:53:50"
  "signature": "14d5b06a2a5a2ec509a148277ed4cbeb3c43301b239f080a3467ff0aba4070e3",
}
```

Signature in Response

For security reasons, we highly recommend you verify the signature in the response.

```
message: merchant_reference + payment_method + status + currency + amount
secret: api_key
```

Example Signature Try your signature online

```
message: dsi39ej430sks03alipaypaidNZD120.00
secret: 111222333
signature: 840151e0dc39496e22b410b83058b4ddd633b786936c505ae978fae029a1e0f1
```