



Merchant API Documentation

Alipay Web Pay Interface

Document version: 1.5.2

API version: 1.5.2

TABLE OF CONTENTS

| | | |
|-------|---------------------------------------|-------------------|
| 1 | INTRODUCTION | 错误! 未定义书签。 |
| 1.1 | Document overview | 错误! 未定义书签。 |
| 1.2 | Reading objects | 错误! 未定义书签。 |
| 2 | PROGRAM OVERVIEW | 错误! 未定义书签。 |
| 2.1 | Industry background | 错误! 未定义书签。 |
| 2.2 | Business implementation process | 错误! 未定义书签。 |
| 2.2.1 | WAP payment service | 错误! 未定义书签。 |
| 3 | DATA FORMAT | 错误! 未定义书签。 |
| 3.1 | Request data | 错误! 未定义书签。 |
| 3.2 | XML data format | 错误! 未定义书签。 |
| 4 | DIGITAL SIGNATURE | 错误! 未定义书签。 |
| 4.1 | Original string of signature | 错误! 未定义书签。 |
| 4.2 | Method of signature | 错误! 未定义书签。 |
| 5 | REPLENISHMENT MECHANISM | 错误! 未定义书签。 |
| 6 | WAP PRE-ORDER INTERFACE | 错误! 未定义书签。 |
| 6.1 | WAP pre-order interface | 错误! 未定义书签。 |
| 6.1.1 | Business function | 错误! 未定义书签。 |
| 6.1.2 | Interactive mode | 错误! 未定义书签。 |
| 6.1.3 | Request parameter list | 错误! 未定义书签。 |

| | |
|--|-------------------|
| 6.1.4Return result | 错误! 未定义书签。 |
| 6.2WAP payment notification interface | 错误! 未定义书签。 |
| 6.2.1Notification result parameter list | 错误! 未定义书签。 |
| 6.2.2Back-end notification result feedback | 错误! 未定义书签。 |
| 6.3Order query interface | 错误! 未定义书签。 |
| 6.3.1Business function | 错误! 未定义书签。 |
| 6.3.2Interactive mode | 错误! 未定义书签。 |
| 6.3.3Request parameter list | 错误! 未定义书签。 |
| 6.3.4Return result | 错误! 未定义书签。 |
| 6.4Refund interface | 错误! 未定义书签。 |
| 6.4.1Business function | 错误! 未定义书签。 |
| 6.4.2Interactive mode | 错误! 未定义书签。 |
| 6.4.3Request parameter list | 错误! 未定义书签。 |
| 6.4.4Return result | 错误! 未定义书签。 |

7NOTES..... **错误! 未定义书签。**

8 ERROR CODE..... **错误! 未定义书签。**

| Document Changes | | |
|------------------|---------|--|
| Time | Version | Description |
| 2018-10-25 | 1.0 | First draft |
| 2018-11-01 | 1.1 | Request URL is changed to https://gateway.wepayez.com/pay/gateway |
| 2019-06-12 | 1.2 | Update SHA256 & RSA signature algorithm. |
| 2021-02-01 | 1.4.2 | Update description |
| 2022-06-02 | 1.4.3 | Add channel_id |
| 2022-07-06 | 1.5 | Update description of fields mch_id, device_info, time_start, time_expire, trade_state, etc. |
| 2022-07-06 | 1.5 | Add the precautions for using the query interface |
| 2022-08-16 | 1.5.1 | Added swiftpass gateway request timeout description |

| | | |
|----------------|-------|--|
| 2022-08-1 6 | 1.5.1 | Add parameter "sign_agentno" |
| 2022-08-1 6 | 1.5.1 | Modify the description of the parameter "Body" |
| 2023-02-1 8 | 1.5.2 | Added new cover sheet and update sign type MD5 |

1 Introduction

1.1 Abstract

QR Pay is a payment method which means merchants integrate with the Third Party's API and generate a QR Code to show customers for each transaction. Then customers use Alipay to scan the QR Code generated by merchants to complete the payment.



QR Pay can be applied in both on-line and off-line payment scenes, like off-line vending machine with screen, off-line cashier, on-line mall on PC and etc.

1.2 Audience

This document is provided to technical and business staff of merchants for reference.

2 Program Overview

2.1 Industry Background

Alipay payment is a payment service function which is provided based on Alipay application, and meanwhile it provides the commercial tenants with such support functions as sales, marketing analysis and management of account. Customers could complete a payment by scanning QR code, being scanned payment QR code or other payment methods.

2.2 Business realization process

2.2.2 Business of Web pay

Usage scenario:

Step (1): Buyer will see an Alipay button when in merchant's website. They can press the button to select Alipay to pay. Merchant generate an QR Code following Alipay rule then display it in its website/screen. As shown in Figure 1&2.

The screenshot shows a payment page with the following elements:

- Payment Method Selection:** "Internet banking (Direct debit)" is selected, with a note "Processing fee applies". Other options include "Credit card / Debit card" and "E-Gift voucher".
- Total Amount:** "Total amount due ≈4,364.58 CNY".
- 4-Step Process:**
 - Select preferred bank/ payment service and click "Purchase".
 - Login and make payment. Please ensure you have disabled browser pop-up blocker.
 - Complete payment within 10 minutes to avoid time-out error.
 - Receive itinerary after payment is confirmed.
- Alipay Logo:** The Alipay logo is prominently displayed.
- Flight Summary (Right Panel):**
 - RETURN:**
 - FD 358: SIN to DMK, 1055, 03 Dec 2016 to 1215, 03 Dec 2016
 - FD 504: DMK to HKG, 1530, 03 Dec 2016 to 1910, 03 Dec 2016
 - Fare, taxes and fees:** (Expanded)
 - Add-ons:** Inclusive of VAT
 - User:** jean liu
 - Checked baggage:** ≈140.01 CNY (20kg)
 - Total:** ≈2,046.95 CNY
 - TOTAL CNY:** ≈4,364.58
 - Purchase Button:** A green button labeled "Purchase »".
- Footer:** Includes "AskAirAsia", "Terms and conditions", "Sitemap", and a disclaimer: "Usage of the AirAsia website states your compliance of our Term of use and Privacy policy ©2015 AirAsia Berhad. This website is owned and operated by AirAsia Berhad ('AirAsia'). 19".

Figure 1

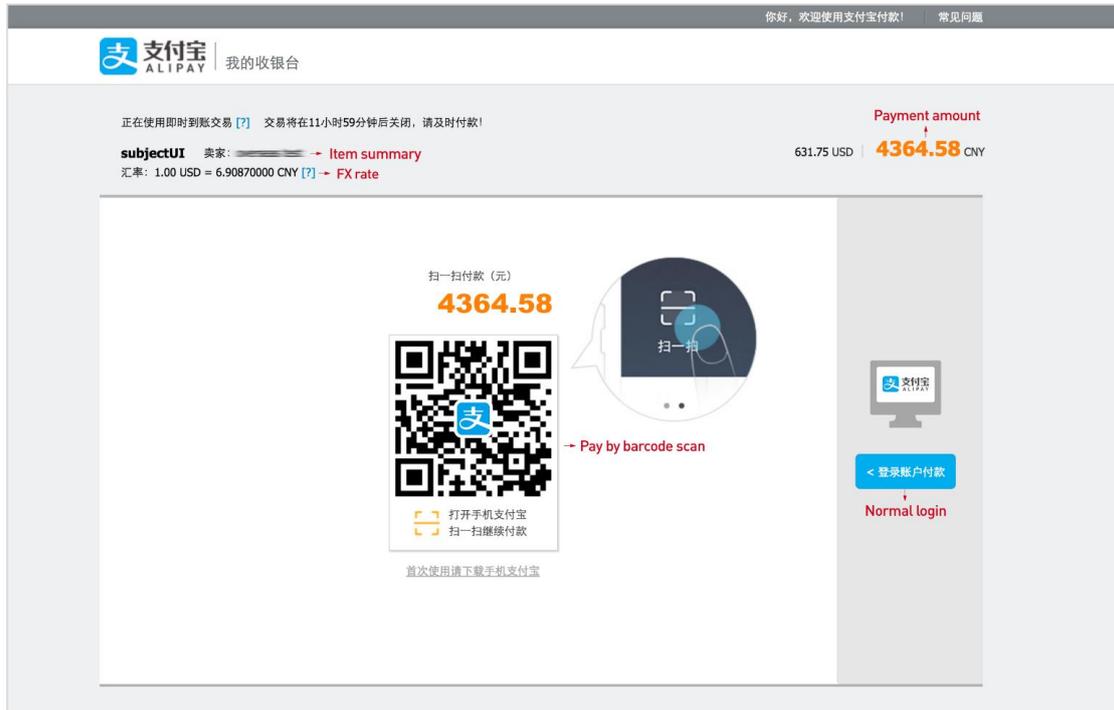


Figure 2

Step (2):The payer scans the QR Code with their Alipay to access the Merchants' product data and proceeds with the transaction. As shown in Figure 3

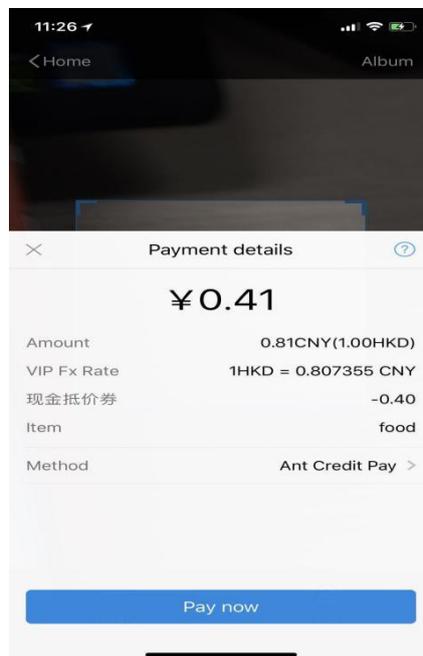


Figure 3

Step (3): The payer confirms their transaction and enters their payment password. As shown in Figure 4

Step (4): The payer is prompted of a successful payment after completing their payment. As

shown in Figure 5

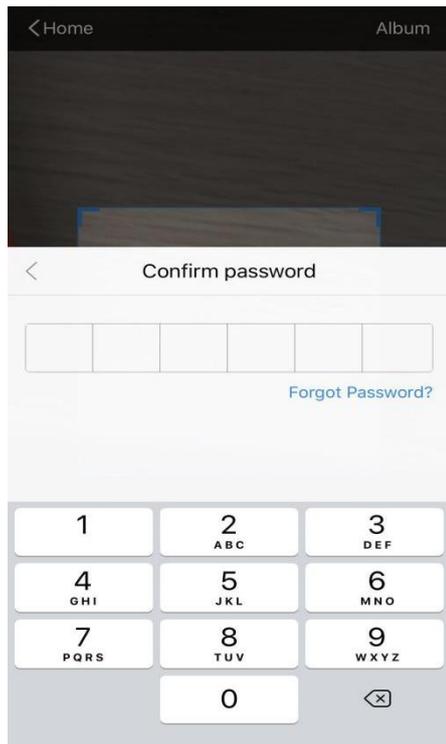


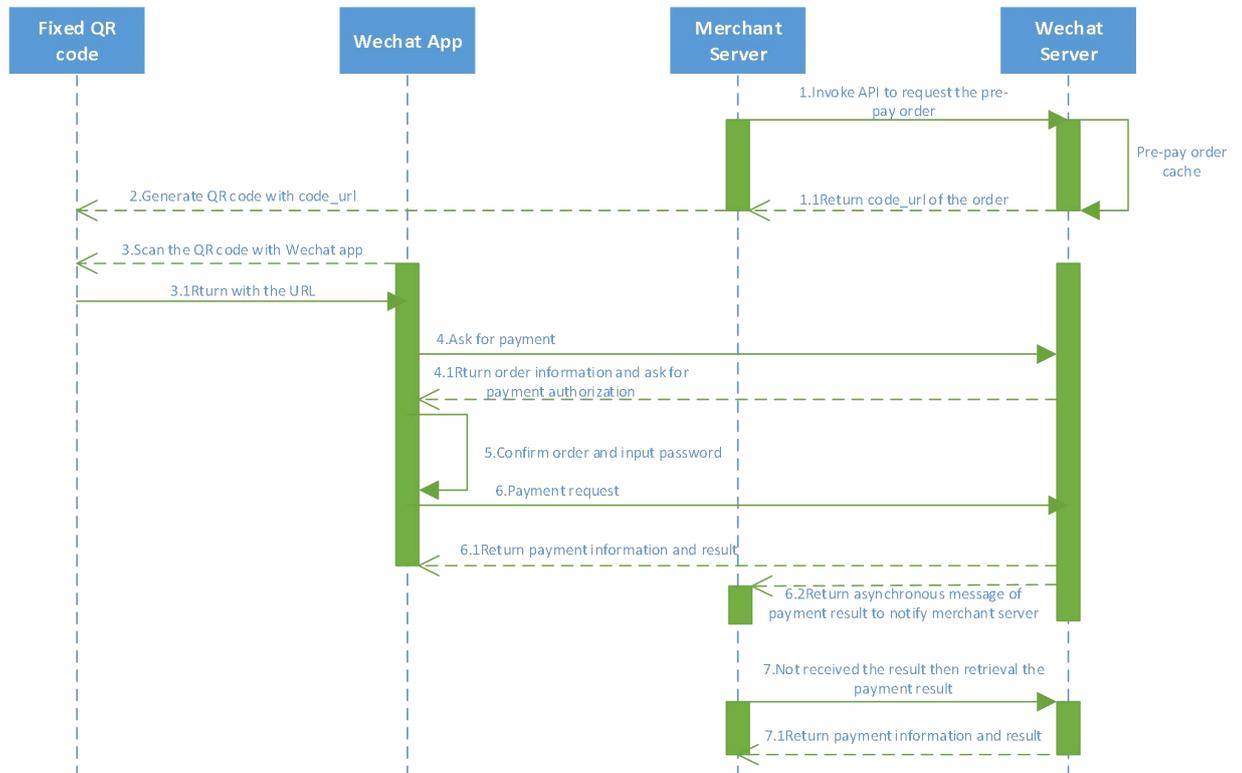
Figure4



Figure5

Time sequence of interface calls:

After the merchant generates an order, the QR code pay interface will be invoked to obtain code_url, and this URL will generate QR code, and after the user scans the QR code, the payment could be invoked.



3 Data Format

3.1 Request data

Used HTTPS POST protocol. To ensure security, transfer data must be signed.

```

<xml>
<service><![CDATA[pay.alipay.webpay.intl]]></service>
<version>2.0</version>
<charset><![CDATA[UTF-8]]></charset>
<sign_type><![CDATA[SHA256]]></sign_type>
<mch_id>7551000001</mch_id>
<out_trade_no><![CDATA[755100000169306196F]]></out_trade_no>
<device_info><![CDATA[69306196F]]></device_info>
<body><![CDATA[Payment]]></body>
    
```

```
<total_fee>50000</total_fee>
<mch_create_ip><![CDATA[124.79.127.180]]></mch_create_ip>
<payment_inst><![CDATA[ALIPAYCN]]></payment_inst>
<notify_url><![CDATA[https://xxxxxxx/notification]]></notify_url>
>
<callback_url><![CDATA[https://xxxxxxxn/xxxx]]></callback_url>
<time_start>20230210180432</time_start>
<time_expire>20230210181932</time_expire>
<nonce_str>1676023472</nonce_str>
<sign><![CDATA[19CFC6852A8AF4C2292E02287D4FA240DC73D7EC8AE403B423
B982918C7623E4]]></sign>
</xml>
```

3.2XML data format

Used Standard XML protocol. First-level node only. No nested nodes.

Protocol error return:

```
<xml>
<status>500</status>
<message><![CDATA[SYSERR]]></message>
</xml>
```

Successful response:

```
<xml>
<charset><![CDATA[UTF-8]]></charset>
<mch_id>7551000001</mch_id>
<nonce_str><![CDATA[20230210180433]]></nonce_str>
<pay_url><![CDATA[https://pay.wepayez.com/pay/webpay?token_id=755
1000001b1189f704ba&service=pay.alipay.webpay.intl]]></pay_url>
<result_code><![CDATA[0]]></result_code>
<sign><![CDATA[20C01C839D370F558D5F898B658F13EC]]></sign>
<sign_type><![CDATA[MD5]]></sign_type>
<status><![CDATA[0]]></status>
<version><![CDATA[2.0]]></version>
</xml>
```

Normal error return:

```
<xml>
<status>0</status>
<message><![CDATA[OK]]></message>
<mch_id><![CDATA[10000100]]></mch_id>
<device_info><![CDATA[1000]]></device_info>
<nonce_str><![CDATA[sthBJ9QyUG6vkrjJ]]></nonce_str>
<sign><![CDATA[6277A96D7875D4FF23AA7B6A4C3046AB]]></sign>
```

```
<result_code>1</result_code>
<err_code><![CDATA[AUTHCODE_EXPIRE]]></err_code>
<err_code_des><![CDATA[二維碼已過期，請刷新再試]]></err_code_des>
</xml>
```

The field named status return '0': successful. Other value means fail.

4 Digital Signature

To ensure the authenticity and integrity of transmissible data, we need to verify the signed data after it being received.

There are two steps in digital signature.

1. Follow the rules to connect the original string that needs to be signed;
2. Calculate the result of signature according to specific algorithm and key.

Generally, the failed result will not be signed.

4.1 Original string of signature

The original string of signature will be assembled into character string according to the following modes no matter whether it is request or response:

1. Besides the sign field, all parameter fields will be **ranked in ascending order according to the ASCII** of the field name and then connected in the format of QueryString (i.e. key1=value1&key2=value2...), and the null value will not transfer and will not participate in formation of string of signature.

2. In the original string of signature, both the field name and field value will adopt original values and will not conduct URL Encode.

3. **The response or notification information returned by platform might increase parameters due to upgrading, and this case should be allowed when the response signature is verified.**

Example:

Calling an interface with following fields:

```
<xml>
<body><![CDATA[測試支付]]></body>
<mch_create_ip><![CDATA[127.0.0.1]]></mch_create_ip>
<mch_id><![CDATA[101520000465]]></mch_id>
<nonce_str><![CDATA[1409196838]]></nonce_str>
<notify_url><![CDATA[http://227.0.0.1:9001/javak/]]></notify_url>
<out_trade_no><![CDATA[141903606228]]></out_trade_no>
<service><![CDATA[pay.alipay.webpay.intl]]></service>
<sign><![CDATA[94B0F11B228BA9DDA2E20E3A9C8B3A2F]]></sign>
<total_fee><![CDATA[1]]></total_fee>
```

```
</xml>
```

The signature field sequence:

```
body=測試支付
&mch_create_ip=127.0.0.1&mch_id=101520000465&nonce_str=1409196838
&notify_url=http://227.0.0.1:9001/javak/&out_trade_no=14190360622
8&service=pay.alipay.webpay.intl&total_fee=1
```

4.2 Method of signature

SHA256 & RSA signature are supported

Signature is a kind of abstract generation algorithm, and if the content of communication keys of the merchant is added to the back of the original string of signature and then signature operation is made, then the abstract character string formed is the signature result. In order to facilitate the comparison, the signature result is uniformly converted to the uppercase letter.

Notes: the coded character set appointed when converting the character string into the byte stream at signing should be in accordance with parameter charset.

SHA256 signature calculation formula:

sign = **SHA256**("The signature field sequence strings"&key="**signature key**").

toUpperCase

Example:

There are XML afferent parameters:

```
<xml>
<auth_code>135187250012923035</auth_code>
<body>test</body>
<charset>UTF-8</charset>
<mch_create_ip>127.0.0.1</mch_create_ip>
<mch_id>127530000052</mch_id>
<nonce_str>1542940680925</nonce_str>
<out_trade_no>1542940643087</out_trade_no>
<service>unified.trade.micropay</service>
<sign>10F2F6DC0D5E008B967CC3C86FC58179686B4EE42F4F68B4A7668501B60
30C29</sign>
<sign_type>SHA256</sign_type>
```

```
<total_fee>2</total_fee>
<version>2.0</version>
</xml>
```

Merchant signature key: 18e0a2ad5d5571af14b855fcf33091f4

i:the first step of which is to connect the original string(string1) that needs signature according to certain rules:

```
auth_code=135187250012923035&body=test&charset=UTF-8&mch_create_ip=127.0.0.1&mch_id=127530000052&nonce_str=1542940680925&out_trade_no=1542940643087&service= unified.trade.micropay&total_fee=1
```

ii:the second step of which is to choose SHA256 and keys to calculate the result of signature(sign):

```
sign
=SHA256(string1&key=18e0a2ad5d5571af14b855fcf33091f4).toUpperCase
=SHA256(auth_code=135187250012923035&body=test&charset=UTF-8&mch_create_ip=127.0.0.1&mch_id=127530000052&nonce_str=1542940680925&out_trade_no=1542940643087&service=unified.trade.micropay&total_fee=1&key=18e0a2ad5d5571af14b855fcf33091f4).toUpperCase()
="10F2F6DC0D5E008B967CC3C86FC58179686B4EE42F4F68B4A7668501B6030C29"
```

RSA signature calculation formula:

The RSA algorithm has always been the most widely used "asymmetric encryption algorithm". By adding the content of the RSA private key of the merchant communication after the original string is signed, the result string is the result of the RSA operation.

Note: The set of coded characters specified when converting a string to a byte stream at signature should match the parameter charset. The RSA key pair is generated by the merchant. You need to upload the public key to the merchant portal and download the SwiftPass public key for the response.

Sign = RSA (request parameter string, merchant RSA private key)

Description: The name of the platform signature algorithm: RSA_1_256, corresponding to the standard signature algorithm name: SHA256WithRSA, the length of the RSA key is required to be 2048.

sign = **RSA**("The signature field sequence strings"&key="**signature private key**").

toUpperCase

Example:

There are XML afferent parameters:

```
<xml>
<out_trade_no><![CDATA[6057113230875088]]></out_trade_no>
<nonce_str><![CDATA[wNzpaD0sN17KI80yBQwINNhfM0IeNqap]]></nonce_str>
<time_expire><![CDATA[]]></time_expire>
<mch_create_ip><![CDATA[127.0.0.1]]></mch_create_ip>
<sign_type><![CDATA[RSA_1_256]]></sign_type>
<total_fee><![CDATA[1]]></total_fee>
<notify_url><![CDATA[http://www.baidu.cn/notify.aspx]]></notify_url>
<body><![CDATA[Test pay]]></body>
<version><![CDATA[1.0]]></version>
<mch_id><![CDATA[102532336411]]></mch_id>
<time_start><![CDATA[]]></time_start>
<attach><![CDATA[Additional information]]></attach>
<service><![CDATA[pay.alipay.native]]></service>
<sign><![CDATA[SE008JDir0uwOuBy8d48SmdfG37PyGvhtqu8pDTo0DpdNkgkLuegNsb6SaL/dEfzu035bwAVwiKSc9m9xBKohGJEMtzRm3tLNnpQ0BDpLlYNWQkr0JN3JAjy6wklicSmfbjXgEvWCcx17MUe59NfI10JRR3MgFg/ySYq2cT4U/o6WJisxtIbre0ZPM66WL815gAzosJW3Gwr+B8fkRyOckF4w64i0TM20xoSHGEa8w1utOFik5wvxyPHb/JcjhRyrBnb2LFZy5Rq3XzZLW6FaK1gunWnjK5+4NiNuHDgm7CKZPp8BphA/qnJtAMGfexEZ8J4z9ktyyWdk8b8VWJ5ow==]]>
</sign>
</xml>
```

Merchant RSA Private key:

```
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBCwggSjAgEAAoIBAQCfU8v4BUr81SKm/H0ahbdQZjEp08nMyk+XuYSatHwnU4//m47R+4G2YB4Z6PHsJi4+ScfJpQutFhKrFwTXZ6TDqLvaqZDDkjq5G271g+PmrzFp7f40/E9m0qjeL64Rjra0rZq123dvPW4vVomMRgRcoPOn0YVp+M6T5PaFgE4M8dh41MzZ57gVwOdd08F99Z92f3QgZtEjI+/EXvMenXxb/aRoFNkt+Wdk2ELJ6MIP0d9U5v3WgLuuv5QnQYzj/RMr8GD+wrDYiNQJxsaTmE/OEJggsumhD4eYY5Y1Ry2EIN504cujYVKU1wOSZgq9oJCynGR0aPuQWx58IHxEtAgMBAAECggEAHfEFd8qm2PTE21Tavec7F+TcgD84IUaz0dZnURtx6YIOoZ5+LH/zVG6juYLJU/Oo5RPac+iMVS68u2JMCp7zm8Ft7B3JkrbuHLNHGuR6Q7PQuXN8PkDcOxqDmZ2kPJz14PZvBZRE0abdug+tMatGzpGAuJzrWcB/N0oVIvrXp9PnOqfo/Y5nxmpOFCImJppIS3AL1pftNtQZo9G15CPHDYtpUbxPtd2MjjW4OLxKuPrOHSwUgo6LW9XSwNXfcuK+lbzLL0Bh1WD9IV/+yCEUEb1N87yxxfhpQFaAhXj5W+B3YsMOZuK93+XMOpyMw8EpUDMOBOnvwb0NSHUrV2RUAQKBgQDT0jlnNS1e7+tjPzFtOhGPj1uCBPAEIEHAcnPg80bEiujxMLCnGaAvmnTrMu4Xo0e5fAP4F7R6UD+IUsfr3CAAU7CadQ49TW+SovAvCiy9AZuSVVIwYnu6QdYgFyPKe1LZYAEq5k+mB1Vh5q0R
```

```
oxMNAAspGYKq8+4MmmsJi7X7QKBgQDAunCOqIiH128bs/1VRThDpzuRW5Qr/SRbO2saVg
5RSHnO/nGT2OuxSTTkc8yrx7qd9SmAxXl5kR238DhMOQOnRBomldmVtAJuJgrdQyt0wXf
eQVQqshqCUaE/xhEbpSCdbPSZbKZZdp1V0y605vXIhxw+1qAvXLcxw46s3R92QQKBgQC1
Q+ejywkVPDILHMwSSehvwThufkCYWYUbbbcVDowpOe5AMoZidtNju7MNjg2rLHTsCx/kBz
Or+7THNw14R7kTiEmg09cO+fu5rHXepGgtig+GJukaZPZ6/bMZJvGOLgOhHmomwG/jdwp
gVtIGBCh6BW5JZcSImT+ykIOoYfvDRuQKBgCgwoHxnBGFFoRoLxE3dhpSk8LT05cbueIB
VuZw6UC3+8PeK82AjIbLMUy04QHupoG6Dyu3BP/1r10jd3L94PBzLBDL7Gm4vJTqW0Dkn
Yo5sMXS1JrnofcKjBv7nbHXZTx3EtJSxpVaOdpca/HpsCuCP3AH2e1yk9sZ3wu6lBYSBA
oGACYM60j1CVRNSzXUNRgiwfwzS69qIleezPc7xQEganpVBI9SZcTNp1kpDKmQikXJ4Yb
5XWn12HCY/sFeBW6Su3ruNqXvg1XiUPbH6A6nxd5B3QX0ms9+wDm6ONysPLRdKbfFO0md
P4CeyuGPdvDIMXP4dJdLhMUL4pcJLI0B7gBE=
```

i: the first step of which is to connect the original string(string1) that needs signature according to certain rules:

```
attach=Additional information&body=Test pay
&mch_create_ip=127.0.0.1&mch_id=102532336411&nonce_str=wNzpaD0sN1
7KI80yBQwINNhfM0IeNqapnotify_url=http://www.baidu.cn/notify.aspx&
out_trade_no=6057113230875088&service=pay.alipay.native&sign_type
=RSA_1_256&total_fee=1&version=1.0
```

ii: the second step of which is to choose RSA and keys to calculate the result of signature(sign):

```
sign=RSA(string1, Merchant RSA Private key)==RSA(attach=Additional
information&body=Test
pay&mch_create_ip=127.0.0.1&mch_id=102532336411&nonce_str=wNzpaD0
sN17KI80yBQwINNhfM0IeNqapnotify_url=http://www.baidu.cn/notify.as
px&out_trade_no=6057113230875088&service=pay.alipay.native&sign_t
ype=RSA_1_256&total_fee=1&version=1.0,MIIEvQIBADANBgkqhkiG9w0BAQE
FAASCBAQwggSjAgEAAoIBAQCfU8v4BUR81SKm/H0ahbdQZjEpO8nMyk+xuYSatHwn
U4//m47R+4G2YB4Z6PHsJi4+ScfJpQutFhKrFwTXZ6TDqLvaqZDDkjq5G271g+Pmr
zFp7f40/E9m0qjeL64Rjra0rZql23dvPW4vVomMRgRcoPOn0YWVp+M6T5PaFgE4M8
dh4lMzZ57gVwOdd08F99Z92f3QgZtEjI+/EXvMenXxb/aRofNkt+Wdk2ELJ6MIP0d
9UU5v3WgLuuv5QnQYzj/RMr8GD+wrDYiNQJxsaTmE/OEJggsumhD4eYY5YlRy2EI
N504cuJYVKU1wOSZgq9oJCynGR0aPuQWx58IHxetAgMBAAECggEAHfEFd8qm2PTE2
lTAvec7F+TcgD84IUaz0dZnURtx6YIOoZ5+LH/zVG6juYlJU/Oo5RPac+iMVS68u2
JMCp7zm8Ft7B3JkrbuHLNHGuR6Q7PQuXN8PkDcOxqDmZ2kPJz14PZvBZRE0abdug+
tMatGzpGAuJzrWcB/N0oVIvrXp9PnOqfo/Y5nxmpOFCImJppIS3AL1pftNtQZo9G1
5CPHDYtpUbXpTd2MjjW4OLxKuPRoHSwUgo6LW9XSwnXfcuK+lbzLL0BhlWD9IV/+y
CEUEblN87yxxfhpQFaAhXj5W+B3YsMOZuK93+XMOpYmw8EpUDMOBOnvwb0NSHUrV2
RUAQKBgQDT0jlnNS1e7+tjPzFtOhGPj1uCBPAEIEHAcnPgd80bEiujxMLCnGaAvmn
TrMu4Xo0e5fAP4F7R6UD+IUsfr3CAAU7CadQ49TW+SovAvciy9AZuSVVIwynu6QdY
gFyPKelLZYAEq5k+mB1Vh5q0RoxMNAAspGYKq8+4MmmsJi7X7QKBgQDAunCOqIiH1
28bs/1VRThDpzuRW5Qr/SRbO2saVg5RSHnO/nGT2OuxSTTkc8yrx7qd9SmAxXl5kR
238DhMOQOnRBomldmVtAJuJgrdQyt0wXfeQVQqshqCUaE/xhEbpSCdbPSZbKZZdp1
V0y605vXIhxw+1qAvXLcxw46s3R92QQKBgQC1Q+ejywkVPDILHMwSSehvwThufkCY
WYUbbbcVDowpOe5AMoZidtNju7MNjg2rLHTsCx/kBzOr+7THNw14R7kTiEmg09cO+f
```

```
u5rHXepGgtig+GJukaZPZ6/bMZJvGOLgOhHmomwG/jdwpGvtIGBCh6BW5JZcSImT+
ykIOoYfvDRuQKBgCgwOHxnBGFfORoLxE3dhpSk8LT05cbueIBVuZW6UC3+8PeK82A
jIbLMUy04QHupoG6Dyu3BP/1r10jd3L94PBzLbLD7Gm4vJTqW0DknYo5sMXS1Jrno
fcKjBv7nbHXZTx3EtJSxpVaOdpcA/HpsCuCP3AH2e1yk9sZ3wu6lBYSBAoGACYM60
j1CVRNSZxUNRgiwfwzS69qIleezPc7xQEganpVBI9SZcTNp1kpDKmQikXJ4Yb5XWn
12HCY/sFeBW6Su3ruNqxxvg1XiUPbh6A6nxd5B3QX0mS9+wDm6ONysPLRdKbfFO
0mdP4CeyuGPdvDIMXP4dJdLhMUL4pcJLI0B7gBE=" ) =
SE008JDir0uwOuBy8d48SmdfG37PyGvhtqu8pDTo0DpdNkgkLuegNsb6SaL/dEzfzU
O35bwAVwiKSc9m9xBKohGJEMtzRm3tLNNpQ0BDpLlYNWQkr0JN3JAjy6wk1icSmfb
jXgEvWCcx17MUe59NfI10JRR3MgFg/ySYq2cT4U/o6WJisxtIbre0ZPM66WL8
```

5 Mechanism to Supplement Order

Notes: in respect of the back-end notification interactive mode, if the response of the merchant received by platform is not **pure character string “success”** or if the response is given **after 5 seconds**, then the notification will be deemed as unsuccessful, and platform will adopt certain strategies (**the interval of notification: 0/15/15/30/180/1800/1800/1800/1800/3600 Unit: seconds**) to re-initiate notification **intermittently**, to improve the success rate of notification, but platform will not guarantee the final success of notification. Because there is the case where the back-end notification will be resent, the same notification might be sent to the merchant system many times. The merchant system must be able to process the repeated notification in a right way. The method recommended by platform is to first examine the status of the corresponding business data upon receiving the notification to judge whether the notification has been disposed, and in case it hasn't been dispose, it will be disposed otherwise, and if it has been disposed, **the pure character string “success”** will be returned directly. Before status examination and disposal of business data are conducted, data lock should be adopted for concurrency control to avoid data chaos caused by **repeated data interposition** in the function.

6 Web Pay Interface

6.1 Web pay

6.1.1 Business function

The QR code payment request need to be initialized. The request is for generating QR code to make transactions of QR Code Pay.

6.1.2 Interactive mode

Request: Background request interaction mode

Return& Notification: Background request interaction mode + **Background notifies interaction mode**

6.1.3 Request Parameters

Request URL: <https://gateway.wepayez.com/pay/gateway>

POST request with content of XML

| Field Name | Required | Type | Description |
|--------------------------|----------|-------------|--|
| Normal Parameters | | | |
| service | Yes | String(32) | Interface type. Value : pay.alipay.webpay.intl |
| version | No | String(8) | Version number. default value : 2. 0 |
| charset | No | String(8) | Value : UTF-8 |
| sign_type | No | String(16) | SHA256: 'SHA256' RSA: 'RSA_1_256' |
| mch_id | Yes | String(32) | Merchant ID, Specifies an unique id assigned by platform. Only Store ID or Ordinary Merchant ID is valid. |
| groupno | No | String (32) | Master merchant ID |
| out_trade_no | Yes | String(32) | The unique trade reference(merchant order id of 5-32 bits) in merchant's system. Letter, number and underline are allowed. Case-sensitive |
| device_info | Yes | String(32) | Specifies a Terminal device id. You can query based on this field on the swiftpass merchant platform |
| body | Yes | String(128) | Description of merchants' goods.Can upload 128 English, 42 Chinese |
| attach | No | String(127) | Merchant additional information. |
| total_fee | Yes | Int | Integral number is allowed only. The unit of the fee is the minimal unit of the local currency. |
| mch_create_ip | Yes | String(16) | Specifies the machine IP that calls the API. |
| supplier | No | String(16) | This value will be shown in the merchant name of Alipay cashier desk |
| notify_url | Yes | String(255) | Specifies the callback address for receiving platform payment notifications. Should be absolute path and ensure platform |

| | | | |
|---------------------|-----------|-------------------|---|
| | | | accessible. i.e. http://wap.tenpay.com/tenpay.asp |
| callback_url | Yes | String(255) | Access the url when the transaction completed. |
| time_start | No | String(14) | Order created date. Format: yyyyMMddHHmmss. i.e.20091225091010. GMT+8 Beijing Time. Alipay can set the time range from 1 minute to 15 days, if not uploading, the default is 2 hours |
| time_expire | No | String(14) | Order timeout date. Format: yyyyMMddHHmmss. i.e.20091225091010. GMT+8 Beijing Time. Alipay can set the time range from 1 minute to 15 days, if not uploading, the default is 2 hours |
| op_user_id | No | String(32) | Specifies the Operator ID. This field shows mch_id by default. |
| goods_tag | No | String(32) | Specifies the label of goods, which is a parameter in the coupon feature for businesses. |
| nonce_str | Yes | String(32) | Included in platform payment API protocols to ensure unpredictability for signatures. 32 characters or fewer. |
| sign | Yes | String(344) | Please refer to section 4 'Digital Signature'. |
| payment_inst | No | String(10) | Used to determine the type of wallet used for payment: If the price currency is not HKD, this field is not passed; If the quoted currency is HKD, please send ALIPAYHK/ALIPAYCN, ALIPAYHK: local wallet; ALIPAYCN: cross-border wallet |
| sign_agentno | No | String(32) | When the merchant is represented by the channel, the agency channel number is sent, and the corresponding sign needs to be signed with the channel's sign_key |

Demo

```

<xml>
<body><![CDATA[changyoyo]]></body>
<charset><![CDATA[UTF-8]]></charset>
<device_info><![CDATA[changyoyo]]></device_info>
<mch_create_ip><![CDATA[58.33.106.38]]></mch_create_ip>
<mch_id><![CDATA[181520234234]]></mch_id>
<nonce_str><![CDATA[HFfP43tL2i]]></nonce_str>
<notify_url><![CDATA[http://58.33.106.38:8080/api/ali/resultNotify]]></notify_url>
<out_trade_no><![CDATA[202209261130000000]]></out_trade_no>
<sign_agentno><![CDATA[1231231]]></sign_agentno>
<payment_inst><![CDATA[AlipayHK]]></payment_inst>
<service><![CDATA[pay.alipay.webpay.intl]]></service>
<sign><![CDATA[B0ECE637F82C135BD39C12E8F51443CEE08FF4A8C8FC2764D90D8770805216
D1]]></sign>
<sign_type><![CDATA[SHA256]]></sign_type>
<time_expire><![CDATA[20220926114000]]></time_expire>
<time_start><![CDATA[20220926113000]]></time_start>
<total_fee><![CDATA[1]]></total_fee>

```

```
<version><![CDATA[2.0]]></version>
</xml>
```

6.1.4 Response result

Data return in real time with XML format

| Field Name | Required | Type | Description |
|---|----------|--------------|--|
| version | Yes | String(8) | Version number. default value : 2. 0 |
| charset | Yes | String(8) | Value : UTF-8 |
| sign_type | Yes | String(16) | SHA256: 'SHA256' RSA: 'RSA_1_256' |
| status | Yes | String(16) | "0" : success. Others value : fail. Specifies communicating label (not transaction label). The status of a transaction is determined by the value of result_code. |
| message | No | String(128) | Return message. Only return when the signature verification invalid. |
| The following fields will returned when status is "0" | | | |
| result_code | Yes | String(16) | "0" : success. Others value : fail. |
| groupno | No | String (32) | Master merchant ID |
| mch_id | Yes | String(32) | Merchant ID, Specifies an unique id assigned by platform. Only Store ID or Ordinary Merchant ID is valid. |
| device_info | No | String(32) | Specifies a Terminal device id. You can query based on this field on the swiftpass merchant platform |
| nonce_str | Yes | String(32) | Included in platform payment API protocols to ensure unpredictability for signatures. 32 characters or fewer. |
| err_code | No | String(32) | Reference error code |
| err_msg | No | String (128) | Error information description |
| sign | Yes | String(344) | Please refer to section 4 'Digital Signature'. |
| sign_agentno | No | String(32) | When the merchant is represented by the channel, the agency channel number is sent, and the corresponding sign needs to be signed with the channel's sign_key |
| The following fields will returned when status and result_code both are "0" | | | |
| pay_url | Yes | String(64) | Its used for Web Pay, merchant use the link to display the QR Code in its website. |

6.2 Notification interface

6.2.1 Notification request parameters

The notification URL is the parameter `notify_url` submitted in section 6.1. And after the payment is completed, platform will send related payment and user information to the URL. And the merchant needs to receive and dispose such information.

When interaction of the back-end notification is being made, if the response received by platform from the merchant is not pure character string "success" or if the response is given after 5 seconds, then it will be deemed as unsuccessful by platform, and platform will adopt certain strategies (**the interval of notification: 0/15/15/30/180/1800/1800/1800/1800/3600 Unit: seconds**) to intermittently re-initiate notification to do its best to improve the rate of success of notification, but the final success of notification will not be guaranteed.

Because there is the case where the back-end notification will be resent, the same notification might be sent to the merchant system many times. **The merchant system must be able to process the repeated notification in a right way.**

The method recommended by platform is to first examine the status of the corresponding business data upon receiving the notification to judge whether the notification has been disposed, and in case it hasn't been disposed, it will be disposed otherwise, and if it has been disposed, the pure character string success will be returned directly. Before status examination and disposal of business data are conducted, data lock should be adopted for concurrency control to avoid data chaos caused by repeated data interposition in the function.

Notices: After the back-end of the merchant has received the parameters of notification. It's required to verify `out_trade_no` and `total_fee` in notification received according to the order number and amount of order of their own business system. And the status of order in the database will only be allowed to update after the verification is consistent.

Suggestion: After creating an order and initiating payment, if the payment success notification request is not received within 5 minutes, it is recommended to initiate an order query interface, which is queried every 5 seconds, for a total of 12 queries.

The back-end notification will return by POST mode through `notify_url` which sent in payment

request.

(Notify in XML format)

| Field Name | Required | Type | Description |
|---|----------|--------------|--|
| version | Yes | String(8) | Version number. default value : 2.0 |
| charset | Yes | String(8) | Value : UTF-8 |
| sign_type | Yes | String(16) | SHA256: 'SHA256' RSA: 'RSA_1_256' |
| status | Yes | String(16) | "0" : success. Others value : fail. Specifies communicating label (not transaction label). The status of a transaction is determined by the value of result_code. |
| message | No | String(128) | Return message. Only return when the signature verification invalid. |
| The following fields will returned when status is "0" | | | |
| result_code | Yes | String(16) | "0" : success. Others value : fail. |
| channel_id | No | String(32) | Channel ID, which is the Agent ID. If the Agent mode is activated on portal, this field is required. Once this field is filled, the signature for the request had to be made with the unified key of the agent, instead of the original key of the store id. |
| groupno | No | String (32) | Master merchant ID |
| mch_id | Yes | String(32) | Merchant ID, Specifies an unique id assigned by platform. Only Store ID or Ordinary Merchant ID is valid. |
| device_info | No | String(32) | Specifies a Terminal device id. You can query based on this field on the swiftpass merchant platform |
| nonce_str | Yes | String(32) | Included in platform payment API protocols to ensure unpredictability for signatures. 32 characters or fewer. |
| err_code | No | String(32) | Reference error code |
| err_msg | No | String (128) | Error information description |
| sign | Yes | String(344) | Please refer to section 4 'Digital Signature'. |
| sign_agentno | No | String(32) | When the merchant is represented by the channel, the agency channel number is sent, and the corresponding sign needs to be signed with the channel's sign_key |
| The following fields will returned when status and result_code both are "0" | | | |
| openid | Yes | String(128) | It is the only user identification under the current appid. |
| trade_type | Yes | String(32) | Value : pay.alipay.webpay.intl |

| | | | |
|--------------------|-----|-------------|--|
| pay_result | Yes | Int | Payment result. "0": success. Others fail. |
| pay_info | No | String(64) | Payment result information. Payment successful return null. |
| transaction_id | Yes | String(32) | platform order number |
| out_transaction_id | Yes | String(32) | order number provided by the third-party (Alipay) |
| out_trade_no | Yes | String(32) | Specifies an order number created by a merchant's system, which is consistent with request. |
| total_fee | Yes | Int | Specifies the total amount. Integral number is allowed only. The unit of the fee is the minimal unit of the local currency. |
| fee_type | No | String(8) | Complies with ISO 4217 standards. |
| attach | No | String(127) | Specifies merchant's data package, which is returned as it is. |
| bank_type | Yes | String(16) | String states bank_type |
| time_end | Yes | String(14) | Specifies the transaction payment time in the format of yyyyMMddHHmmss, such as 20091225091010 for Dec 25, 2009 09:10:10.GMT+8 Beijing Time. |

6.2.2 Response of notification

platform server will send notification, then post send XML data flow, the notify_url of the merchant will receive the result of notification, with the method of reception being as written in demo (for example, callback method in php, notify.aspx file in c#, TestPayResultServlet method in java), and after the merchant conducts business disposal, the feedback of result of disposal needs to be given in the form of pure character string, with its content being as follows:

| Returned results | Description |
|-------------------------|---|
| success | The disposal is successful, and after swiftpass system receives this result, no ongoing notification will be made. |
| fail or other character | The disposal fails, and if swiftpass receives this result or doesn't receive any result, then the system will resend the notification through the mechanism of supplementing order (for detail please see section 5). |

6.3 Retrieve transaction result interface

6.3.1 Business function

Retrieve transaction result information with platform order number or merchant order number.

6.3.2 Interactive mode

Background interaction mode.

6.3.3 Request parameters

Request URL: <https://gateway.wepayez.com/pay/gateway>

POST request with content of XML

| Field Name | Required | Type | Description |
|----------------|----------|-------------|--|
| service | Yes | String(32) | Value : unified.trade.query |
| version | No | String(8) | Version number. default value : 2. 0 |
| charset | No | String(8) | Value : UTF-8 |
| sign_type | No | String(16) | SHA256: 'SHA256' RSA: 'RSA_1_256' |
| mch_id | Yes | String(32) | Merchant ID, Specifies an unique id assigned by platform. Only Store ID or Ordinary Merchant ID is valid. |
| groupno | No | String (32) | Master merchant ID |
| out_trade_no | No | String(32) | The unique trade reference of merchant system. out_trade_no and transaction_id at least one required. transaction_id priority when both be filled. |
| transaction_id | No | String(32) | The unique trade reference of platform system. out_trade_no and transaction_id at least one required. transaction_id priority when both be filled. |
| nonce_str | Yes | String(32) | Included in platform payment API protocols to ensure |

| | | | |
|--------------|-----|-------------|---|
| | | | unpredictability for signatures. 32 characters or fewer. |
| sign | Yes | String(344) | Please refer to section 4 'Digital Signature'. |
| sign_agentno | No | String(32) | When the merchant is represented by the channel, the agency channel number is sent, and the corresponding sign needs to be signed with the channel's sign_key |

6.3.4 Response parameters

Data return in real time with XML format

| Field Name | Required | Type | Description |
|---|----------|--------------|--|
| version | Yes | String(8) | Version number. default value : 2. 0 |
| charset | Yes | String(8) | Value : UTF-8 |
| sign_type | Yes | String(16) | SHA256: 'SHA256' RSA: 'RSA_1_256' |
| status | Yes | String(16) | "0" : success. Others value : fail. Specifies communicating label (not transaction label). The status of a transaction is determined by the value of trade_state . |
| message | No | String(128) | Return message. Only return when the signature verification invalid. |
| The following fields will returned when status is "0" | | | |
| result_code | Yes | String(16) | "0" : success. Others value : fail. |
| mch_id | Yes | String(32) | Merchant ID, Specifies an unique id assigned by platform. Only Store ID or Ordinary Merchant ID is valid. |
| groupno | No | String (32) | Master merchant ID |
| device_info | No | String(32) | Specifies a Terminal device id. You can query based on this field on the swiftpass merchant platform |
| nonce_str | Yes | String(32) | Included in platform payment API protocols to ensure unpredictability for signatures. 32 characters or fewer. |
| err_code | No | String(32) | Reference error code |
| err_msg | No | String (128) | Error information description |
| sign | Yes | String(344) | Please refer to section 4 'Digital Signature'. |
| sign_agentno | No | String(32) | When the merchant is represented by the channel, the agency channel number is sent, and the corresponding sign needs to be signed with the |

| channel's sign_key | | | |
|---|-----|-------------|---|
| The following fields will returned when status and result_code both are "0" | | | |
| trade_state | Yes | String(32) | SUCCESS:Payment successful REFUND:Order to be refunded NOTPAY:Order not paid PAYERROR:Payment failed (payment status failed to be returned by bank or other reasons) |
| The following fields will returned when trade_state is "SUCCESS" | | | |
| trade_type | Yes | String(32) | Value : pay.alipay.webpay.intl |
| transaction_id | Yes | String(32) | The unique trade reference of platform system. |
| out_transaction_id | Yes | String(32) | order number provided by the third-party (Alipay) |
| out_trade_no | Yes | String(32) | The unique trade reference of merchant system. |
| total_fee | Yes | Int | The total amount of the transaction.The unit of the fee is the minimal unit of the local currency. |
| coupon_fee | No | Int | Coupon amount. coupon_fee <= total_fee. total_fee - oupon_fee = cash pay amount |
| fee_type | No | String(8) | Complies with ISO 4217 standards. |
| attach | No | String(127) | Specifies merchant's data package, which is returned as it is. |
| bank_type | No | String(16) | String states bank_type |
| bank_billno | No | String(32) | Bank order number. Null for Alipay payment. |
| time_end | Yes | String(14) | Specifies the transaction payment time in the format of yyyyMMddHHmmss, such as 20091225091010 for Dec 25, 2009 09:10:10. GMT+8 Beijing. |

7 Refund&Query Interface

7.1Refund interface

7.1.1Business function

If the merchant initiate refund in respect of an order that has been successfully paid, then the result of the operation will be synchronously returned in the same dialogue.

I. Refund mode

Currently, the fund of the refund will follow the same route back to customers' account.

Notes: Generally the amount will be refunded to the bank account within 1 to 3 working days after the refund is initiated.

The different parts of refund of a same order need to be set with the same order number and the different out_refund_no. After a refund that fails is resubmitted, the original out_refund_no will be adopted. The total sum of refund couldn't surpass the amount actually paid by users (the amount of the cash coupon couldn't be refunded).

II. Restrictions of refund

When operating the refund, the merchant should pay attention to the restrictions of refund to avoid the request of refund that will not be successful, and the main restrictions of refund are as follows:

In platform system, only if the accumulative amount of refund doesn't surpass the total sum of payment for the transacted order, then the same order could be refunded for several times, and the refund application form number (there are such parameters in the refund interface) rather than the transaction order number shall be solely used to confirm one time of refund. The refund application form number is to be generated by the merchant, and so the merchant has to guarantee the uniqueness of the refund application form. In the process of refund, the merchant needs to pay special attention that only when the refund is confirmed to fail, then could another refund be re-initiated.

Currently most banks support full refund and partial refund, but there are a few banks that don't support full refund or partial refund or don't support refund. In such case the merchant could negotiate with the buyer to refund to the Alipay balance account.

Currently only refund interface without key is provided, and in case the merchant needs refund interface with key, please contact the business person.

7.1.2Interactive mode

Background interaction mode

7.1.3 Request parameters

Request URL: <https://gateway.wepayez.com/pay/gateway>

POST request with content of XML

| Field Name | Required | Type | Description |
|----------------|----------|-------------|---|
| service | Yes | String(32) | Value : unified.trade.refund |
| version | No | String(8) | Version number. default value : 2. 0 |
| charset | No | String(8) | Value : UTF-8 |
| sign_type | No | String(16) | SHA256: 'SHA256' RSA: 'RSA_1_256' |
| mch_id | Yes | String(32) | Merchant ID, Specifies an unique id assigned by platform. Only Store ID or Ordinary Merchant ID is valid. |
| groupno | No | String (32) | Master merchant ID |
| out_trade_no | No | String(32) | The unique trade reference of merchant system. out_trade_no and transaction_id at least one required. transaction_id priority when both be filled. |
| transaction_id | No | String(32) | The unique trade reference of platform system. out_trade_no and transaction_id at least one required. transaction_id priority when both be filled. |
| out_refund_no | Yes | String(32) | Specifies the internal refund number, which is unique in the system. A single transaction can be processed as multiple partial refunds, with the total sum of the partial refunds being equal to the original one. If the refund is not successful. The recall function should be used with same refund number to avoid duplication of refunds. |
| total_fee | Yes | Int | The total amount of the transaction. The unit of the fee is the minimal unit of the local currency. |
| refund_fee | Yes | Int | Refund amount. The unit of the fee is the minimal unit of the local currency. Partial refund can be supported.. Partial refund can be supported. |
| op_user_id | Yes | String(32) | Specifies the Operator ID. This field shows mch_id by default. |
| refund_channel | No | String(16) | Value : ORIGINAL. The money will refund back to where it came from. |
| nonce_str | Yes | String(32) | Included in platform payment API protocols to ensure unpredictability for signatures. 32 characters or fewer. |
| sign | Yes | String(344) | Please refer to section 4 'Digital Signature'. |
| sign_agentn | No | String(32) | When the merchant is represented by the channel, the |

| | | | |
|---|--|--|--|
| o | | | agency channel number is sent, and the corresponding sign needs to be signed with the channel's sign_key |
|---|--|--|--|

7.1.4 Response parameters

Data return in real time with XML format

| Field Name | Required | Type | Description |
|---|----------|--------------|--|
| version | Yes | String(8) | Version number. default value : 2. 0 |
| charset | Yes | String(8) | Value : UTF-8 |
| sign_type | Yes | String(16) | SHA256: 'SHA256' RSA: 'RSA_1_256' |
| status | Yes | String(16) | "0" : success. Others value : fail. Specifies communicating label (not transaction label). The status of a transaction is determined by the value of result_code. |
| message | No | String(128) | Return message. Only return when the signature verification invalid. |
| The following fields will returned when status is "0" | | | |
| result_code | Yes | String(16) | "0" : success. Others value : fail. |
| groupno | No | String (32) | Master merchant ID |
| mch_id | Yes | String(32) | Merchant ID, Specifies an unique id assigned by platform. Only Store ID or Ordinary Merchant ID is valid. |
| device_info | No | String(32) | Specifies a Terminal device id. You can query based on this field on the swiftpass merchant platform |
| nonce_str | Yes | String(32) | Included in platform payment API protocols to ensure unpredictability for signatures. 32 characters or fewer. |
| err_code | No | String(32) | Reference error code |
| err_msg | No | String (128) | Error information description |
| sign | Yes | String(344) | Please refer to section 4 'Digital Signature'. |
| sign_agentno | No | String(32) | When the merchant is represented by the channel, the agency channel number is sent, and the corresponding sign needs to be signed with the channel's sign_key |
| The following fields will returned when status and result_code both are "0" | | | |
| transaction_id | Yes | String(32) | The unique trade reference of platform system. |
| out_trade_no | Yes | String(32) | The unique trade reference of merchant system. |
| out_refund_no | Yes | String(32) | Specifies the internal refund number, which is unique in the |

| | | | |
|-------------------|-----|------------|---|
| | | | merchant system. |
| refund_id | Yes | String(32) | Specifies the internal refund number, which is unique in the platform system. |
| refund_channel | Yes | String(16) | Value : ORIGINAL. The money will refund back to where it came from. |
| refund_fee | Yes | Int | Refund amount. The unit of the fee is the minimal unit of the local Currency. Partial refund can be supported. |
| coupon_refund_fee | No | Int | Coupon refund amount. $coupon_refund_fee \leq refund_fee$. $refund_fee - coupon_refund_fee = \text{cash refund amount}$ |

8 Notes

1. The unit of any related amount is the minimal unit of the currency, and decimals are not allow.
2. notify_url means that platform server directly initiates request from back end to merchant's server, and when disposing, the merchant could not examine the user's cookie or session; the merchant's updating of DB and other goods delivery procedure needs to be made after notify_url is completed to ensure that to supplement order after order fails will be successful.
3. notify_url means that it might have repeated notification and the merchant needs to do away with the repeated ones to avoid repeated goods delivery.
4. notify_url means the receive notification, and if the merchant disposes successfully or if the examined order has been disposed, then the successful disposal mark, the pure character string success needs to be returned, and the character string success is not case sensitive; if we don't receive the returned success, then our server will keep sending notification to you until three hours later; if it's assumed that all orders don't return success, then the load of notification of our server will be increase, and the worst case is that the notification normally sent to the merchants might delay; besides, we will urge you to perfect, and if you don't improve for a long time period, then the R&D or operation and maintenance technology staff will adopt control measures over the payment interface opened by your company. For the parameters sent in requesting interface in the document, if the one required to fill is given a

yes, then it has to be sent (in case of lack, an error will be alarmed), and if the one required to fill is given a no, then it will be optional to be sent.

5. For the returned parameters, if the one required to fill is given a yes, then it has to be returned, and if the one required to fill is given a no, then it will be optional to be returned.

Because of upgrading or configuration or other cases, the actually returned parameters might not be totally consistent with those in the document, and the actually received parameters shall prevail.

6. Other notes

- (1) The problem of capital and small letter of parameters

Please pay attention to the problem of capital and small letter required in the document, such as **“after signature, the character of the character string needs to be converted to uppercase”**.

- (2) The problem of format of parameters

All introduced parameters are of the type of character string, and please pay attention to the specific requirements in different places of the document.

- (3)The problem of time stamp

Please use Linux time stamp, and note that its format is character string.

- (4) The problem of order number payment made by the same merchant

If the payment of order of the merchant fails, then it's required to generate a new order number to re-initiate payment, and the original order number needs to be invoked for closing to avoid repeated payment; After an order is placed in the system, if the payment of users is time out, the system will exit and no longer accept the payment to avoid ongoing payment of users, and please invoke the close order interface.

Notes: after an order is generated, to immediately invoke the close order interface will not be allowed, because the shortest invocation interval is 5 minutes.

7.Request swiftpass gateway If there is no clear result of synchronization within 10 seconds, it can be considered that the transaction request has timed out

9 Error Code

Alipay:

Please check https://global.alipay.com/docs/ac/gr/error_code for details.

| Error code | Description |
|--------------------------|---|
| ACCESS_FORBIDDEN | You have no permission to use the product. Check your agreement with Alipay. See ACCESS_FORBIDDEN for details. |
| AGREEMENT_NOT_EXIST | The related agreement does not exist. See AGREEMENT_NOT_EXIST for details. |
| ALIN10023 | An order is being paid by a second payer. See ALIN10023 for details. |
| ALIN10070 | The signature is incorrect. See ALIN10070 for details. |
| ALIN10129 | In APP payment, the PID does not match the environment. For example, the PID is from production environment but is tested in sandbox environment. See ALIN10129 for details. |
| ALIN38173 | The orderInfo string in the request is not parsed correctly. See ALIN38173 for details. |
| ALIN42282 | The same external transaction number, out_trade_no, is used to repeatedly initiate requests, and the request parameters are inconsistent. See ALIN42282 for details. |
| ALIN43877 | The merchant called Alipay domestic payment solution interface, but the buyer is trying to make payment in a foreign country (outside of China). See ALIN43877 for details. |
| ALIN43911 | Transaction risks are detected at the user side. See ALIN43911 for details. |
| BEYOND_PAY_RESTRICTION | The balance payment amount exceeds the limit. Change to other payment method such as cash or credit card. See ALIN43911 for details. |
| BUYER_BALANCE_NOT_ENOUGH | Insufficient balance for current operation. Top-up the balance or link a bank card. |

| | |
|--|---|
| | See BUYER_BALANCE_NOT_ENOUGH for details. |
| BUYER_BANKCARD_BALANCE_NOT_ENOUGH | Insufficient bank balance for current operation. Top-up the balance or link a bank card. See BUYER_BANKCARD_BALANCE_NOT_ENOUGH for details. |
| BUYER_ENABLE_STATUS_FORBID | The buyer account is disabled because of identity or security reasons. See BUYER_ENABLE_STATUS_FORBID for details. |
| BUYER_NOT_EXIST | The buyer_identity_code value is incorrect. See BUYER_NOT_EXIST for details. |
| BUYER_NOT_MAINLAND_CERT | The user is not an eligible customers of the Alipay payment products that you use. See BUYER_NOT_MAINLAND_CERT for details. |
| BUYER_PAYMENT_AMOUNT_DAY_LIMIT_ERROR | The total payment amount for the buyer exceeded user daily max-amount limit. Change to other payment method such as cash or credit card. See BUYER_PAYMENT_AMOUNT_DAY_LIMIT_ERROR for details. |
| BUYER_PAYMENT_AMOUNT_MONTH_LIMIT_ERROR | The total payment amount for the buyer exceeded user monthly max-amount limit. Change to other payment method such as cash or credit card. See BUYER_PAYMENT_AMOUNT_MONTH_LIMIT_ERROR for details. |
| CONTACT_NO_MANDATORY | contact_no is not passed in the request parameters. See CONTACT_NO_MANDATORY for details. |
| CONTEXT_INCONSISTENT | The transaction information is inconsistent by the same out_trade_no. Check the request data. See CONTEXT_INCONSISTENT for details. |
| CURRENCY_NOT_SAME | The refund currency is inconsistent. See CURRENCY_NOT_SAME for details. |
| CURRENCY_NOT_SUPPORT | The currency is not supported. Check the agreement with Alipay. See CURRENCY_NOT_SUPPORT for details. |
| DATA_NOT_EXIST | The secondary merchant is not successfully registered. See DATA_NOT_EXIST for details. |
| DISCORDANT_REPEAT_REQUEST | The same out_return_no is used to initiate multiple refund requests, and the request parameters are inconsistent. See DISCORDANT_REPEAT_REQUEST for details. |

| | |
|-----------------------------------|---|
| DUPLICATE_PAY_CURRENCY_NOT_EQUAL | <p>More than one payment requests share a same out_trade_no, and the parameter currency is not the same.</p> <p>See DUPLICATE PAY CURRENCY NOT EQUAL for details.</p> |
| DUPLICATE_REQUEST | <p>The registration request for the same PM / PMS is submitted already and is currently under review.</p> <p>See DUPLICATE_REQUEST for details.</p> |
| EXCHANGE_AMOUNT_OR_CURRENCY_ERROR | <p>The exchange amount or currency is incorrect. Check the amount and the currency parameter.</p> <p>See EXCHANGE_AMOUNT_OR_CURRENCY_ERROR for details.</p> |
| EXIST_FORBIDDEN_WORD | <p>Prohibited words are included in the transaction request.</p> <p>See EXIST_FORBIDDEN_WORD for details.</p> |
| HAS_NO_PRIVILEGE | <p>The contract of the PID expires or is terminated.</p> <p>See HAS_NO_PRIVILEGE for details.</p> |
| ILLEGAL_ACCESS_SWITCH_SYSTEM | <p>You are not allowed to access the system of this type. Contact Alipay Technical Support if needed.</p> |
| ILLEGAL_ARGUMENT | <p>The parameter is incorrect. Check each request parameter according to the API specification.</p> <p>See ILLEGAL_ARGUMENT for details.</p> |
| ILLEGAL_CLIENT_IP | <p>The client IP address is illegal. Check the Client IP address.</p> <p>See ILLEGAL_CLIENT_IP for details.</p> |
| ILLEGAL_EXTERFACE | <p>The Alipay API you call is incorrect.</p> <p>See ILLEGAL_EXTERFACE for details.</p> |
| ILLEGAL_PARTNER | <p>The partner ID is incorrect. Ensure the value of partner parameter matches the partner value provided by Alipay.</p> <p>See ILLEGAL_PARTNER for details.</p> |
| ILLEGAL_PARTNER_EXTERFACE | <p>The PID used in the request did not sign the corresponding contract.</p> <p>See ILLEGAL_PARTNER_EXTERFACE for details.</p> |
| ILLEGAL_SECURITY_PROFILE | <p>The matching private key configuration is not found.</p> <p>For the solution, see ILLEGAL_SECURITY_PROFILE.</p> |
| ILLEGAL_SIGN | <p>Errors exist in the signature.</p> <p>See Digital signature for details.</p> |
| ILLEGAL_SIGN_TYPE | <p>Alipay's request only supports three signing methods: MD5, RSA, and RSA2. If other values are passed to sign_type, this error will be returned.</p> |

| | |
|------------------------------------|--|
| | Please check if sign_type is one of MD5, RSA, RSA2, and check if the value is passed incorrectly. |
| INVALID_PARAMETER | The parameter name is wrong, the parameter value does not meet the requirements, and required parameters are not filled, and so on. See INVALID_PARAMETER for details. |
| INVALID_RECEIVE_ACCOUNT | Error exists in the receipt account information. See INVALID_RECEIVE_ACCOUNT for details. |
| INVALID_ROUNDED_AMOUNT | The refund with this amount may violate the rule that the calculated amount of both CNY and foreign currency should be fully or not fully refunded at the same time. Take a transaction with 0.07 CNY (0.01 USD) as an example. A refund with 0.06 CNY would not be accepted because this comes to a situation where there is 0.01 CNY (0 USD) left for this transaction. See INVALID_ROUNDED_AMOUNT for details. |
| LBS_GEOGRAPHIC_INFORMATION_INVALID | The actual country of the address defined by store_address does not match the country defined by store_country, or the address defined by store_address cannot be found on Google Map. See LBS_GEOGRAPHIC_INFORMATION_INVALID for details. |
| MCC_TYPE_ILLEGAL | MCC (parameter secondary_merchant_industry) is invalid. See MCC_TYPE_ILLEGAL for details. |
| MOBILE_PAYMENT_SWITCH_OFF | The Payment Code feature is disabled in the customer's Alipay app. See MOBILE_PAYMENT_SWITCH_OFF for details. |
| MORE_THAN_ALLOW_REFUND_FOREX_FEE | The principle of Alipay's split-refund is to return the original way. If a transaction involves refunding the split, and the refund amount exceeds the actual receipt amount of the account, this error code will be returned. See MORE_THAN_ALLOW_REFUND_FOREX_FEE for details. |
| NOT_CERTIFIED_USER | Buyer qualifications do not match. See NOT_CERTIFIED_USER for details. |
| NOT_PRIVATE_ACCOUNT_USER | The transaction can only be paid with the buyer's personal account, not the business account. See NOT_PRIVATE_ACCOUNT_USER for details. |
| NOT_SUPPORT_PAYMENT_INST | The Alipay Wallet version is not supported. |

| | |
|------------------------------|--|
| | <p>See NOT_SUPPORT_PAYMENT_INST for details.</p> |
| PARAM_ILLEGAL | <p>The parameter is illegal. Modify the parameter according to the API document.</p> <p>For more information about this error code in certain interfaces, see PARAM_ILLEGAL.</p> |
| PAYER_ENABLE_STATUS_FORBID | <p>The buyer's Alipay account is deactivated and the real-name verification information cannot be sent to the customs.</p> <p>See PAYER_ENABLE_STATUS_FORBID for details.</p> |
| PAYMENT_FAIL | <p>The transaction failed.</p> <p>See PAYMENT_FAIL for details.</p> |
| PAYMENT_REQUEST_HAS_RISK | <p>The merchant status is abnormal, or the buyer's payment has risks, such as offsite payment or large payment.</p> <p>See PAYMENT_REQUEST_HAS_RISK for details.</p> |
| PRODUCT_AMOUNT_LIMIT_ERROR | <p>The transaction amount exceeded the limit.</p> <p>See PRODUCT_AMOUNT_LIMIT_ERROR for details.</p> |
| PULL_MOBILE_CASHIER_FAIL | <p>Failed to call the mobile cashier.</p> <p>See PULL_MOBILE_CASHIER_FAIL for details.</p> |
| PURCHASE_TRADE_NOT_EXIST | <p>The order for the refund request does not exist (status is TRADE_NOT_EXIST) or has not been paid (status is WAIT_BUYER_PAY).</p> <p>See PURCHASE_TRADE_NOT_EXIST for details.</p> |
| QRCODE_HAS_BEEN_EXIST | <p>The QR code already exists. Try again or Contact Alipay Technical Support if needed.</p> <p>See QRCODE_HAS_BEEN_EXIST for details.</p> |
| REASON_TRADE_REFUND_FEE_ERR | <p>Invalid refund amount.</p> <p>See REASON_TRADE_REFUND_FEE_ERR for details.</p> |
| REFUND_CHARGE_ERROR | <p>The refund is initiated before Alipay receive the payment from the user.</p> <p>See REFUND_CHARGE_ERROR for details.</p> |
| REFUND_REQUEST_HAS_RISK | <p>The merchant status is abnormal and transactions are not refundable.</p> <p>See REFUND_REQUEST_HAS_RISK for details.</p> |
| REFUNDMENT_VALID_DATE_EXCEED | <p>The transaction exceeded the valid refund period.</p> <p>See REFUNDMENT_VALID_DATE_EXCEED for details.</p> |
| REGISTRATION_NO_MANDATORY | <p>registration_no is not passed in the request parameters.</p> <p>See REGISTRATION_NO_MANDATORY for details.</p> |

| | |
|---------------------------------|---|
| REPEATED_REFUNDMENT_REQUEST | Duplicated refund request. Please do not refund repeatedly. See REPEATED_REFUNDMENT_REQUEST for details. |
| REQUEST_AMOUNT_EXCEED | The value of refund_amount or the total refund amount is more than the transaction amount. See REQUEST_AMOUNT_EXCEED for details. |
| RESTRICTED_MERCHANT_INDUSTRY | The amount exceeded the merchant industry single order amount limit. See RESTRICTED_MERCHANT_INDUSTRY for details. |
| RETURN_AMOUNT_ERROR | The refund amount in CNY, which is the value of return_rmb_amount, is less than 0.01 when converted to the settlement amount in foreign currency. See RETURN_AMOUNT_ERROR for details. |
| RETURN_AMOUNT_EXCEED | The total refund amount exceeds the amount actually paid for the transaction. See RETURN_AMOUNT_EXCEED for details. |
| SECONDARY_MERCHANT_ID_BLANK | No secondary merchant information was passed in the payment request. See SECONDARY_MERCHANT_ID_BLANK for details. |
| SECONDARY_MERCHANT_ID_INVALID | The parameter secondary_merchant_id passed in the request is incorrect. See SECONDARY_MERCHANT_ID_INVALID for details. |
| SECONDARY_MERCHANT_STATUS_ERROR | The secondary merchant status is abnormal. See SECONDARY_MERCHANT_STATUS_ERROR for details. |
| SELLER_BALANCE_NOT_ENOUGH | When refunding, the seller's account balance was insufficient and the refund failed. See SELLER_BALANCE_NOT_ENOUGH for details. |
| SELLER_NOT_EXIST | The parameter seller_id in the request is not consistent with partner. Or seller_email does not match partner. See SELLER_NOT_EXIST for details. |
| SOUNDWAVE_PARSER_FAIL | The payment code (buyer_identity_code) is invalid. See SOUNDWAVE_PARSER_FAIL for details. |
| SYSTEM_ERROR | Alipay system failed to process the request because of temporary internal glitch. |

| | |
|-----------------------|---|
| | See SYSTEM_ERROR for details. |
| TARGET_SYSTEM_ERROR | For interface alipay.acquire.overseas.spot.pay, parameter extend_info is not in JSON format. Make sure the format of extend_info is JSON. |
| TOTAL_FEE_EXCEED | The transaction amount exceeds the limit. See TOTAL_FEE_EXCEED for details. |
| TRADE_BUYER_NOT_MATCH | The buyer does not match. See TRADE_BUYER_NOT_MATCH for details. |
| TRADE_CANCEL_TIME_OUT | The cancellation request is beyond the opening hours. See TRADE_CANCEL_TIME_OUT for details. |
| TRADE_HAS_CLOSE | Payment or refund is not allowed for orders with a status of TRADE_CLOSED. See TRADE_HAS_CLOSE for details. |
| TRADE_HAS_FINISHED | The transaction is successfully paid and the refund period has expired. No payment or cancel request is allowed. See TRADE_HAS_FINISHED for details. |
| TRADE_HAS_SUCCESS | Already successful transactions are not allowed to initiate a payment again. See TRADE_HAS_SUCCESS for details. |
| TRADE_NOT_EXIST | The transaction does not exist. See TRADE_NOT_EXIST for details. |
| TRADE_SETTLE_ERROR | The error might be caused by the following reasons: Errors exist in split_fund_info. For example, the value of transOut is incorrect. Multiple partial refunds are requested within a same second. Network jitter occurs during the settlement. See TRADE_SETTLE_ERROR for details. |
| TRADE_STATUS_ERROR | The corresponding transaction status is not allowed for current operation. See TRADE_STATUS_ERROR for details. |
| UNKNOW | In the in-store payment, if the amount is small and confidential, payment is returned to SUCCESS. If the amount is large, the confirmation payment page will be invoked, and the return code is UNKNOW. See UNKNOW for details. |

| | |
|--------------------------------|--|
| UNKNOWN_ERROR | The service is temporarily unavailable. Try again later. |
| 购汇金额或币种错误 | The transaction amount in transaction currency is rounded to 0 in the settlement currency. See 购汇金额或币种错误 for details. |
| 系统有点儿忙, 一会儿再试试, 或者可以在 电脑上付款 | The customer cannot use Alipay Global Payment Products. See 系统有点儿忙…… for details. |