

目录

概述	1.1
接口规则	1.2
参数规定	1.3

接口

刷脸支付	2.1
刷卡支付	2.2
预下单支付	2.3
运维管理	2.4

- 超赢聚合支付API在线文档
 - 版本说明

超赢聚合支付API在线文档



当前版本: v2.5.2

最后维护: 尹自强, yinziqiang@chaoying.com.cn

更新日期: 2019-08-05

版本说明

【v2.5.2】2019-08-05

- 添加获取超赢商户基础信息接口

【v2.5.1】2019-06-10

- 修改退款查询接口移除商户单号out_trade_no和平台单号transaction_id请求参数
- 修复退款查询接口BUG

【v2.5.0】2019-06-04

- 修改所有接口pid和version请求参数为必填项
- 修改支付类接口响应参数添加订单应付金额base_fee字段, total_fee字段修改为订单实付金额
- 修改刷脸支付-提交支付接口添加商品详情goods_detail请求参数, goods_tag字段修改为订单优惠标记; 添加营销详情promotion_detail响应参数
- 修改刷卡支付-提交支付接口添加商品详情goods_detail请求参数, goods_tag字段修改为订单优惠标记; 添加营销详情promotion_detail响应参数
- 修改预下单支付-微信预下单接口添加商品详情goods_detail请求参数, goods_tag字段修改为订单优惠标记;
- 修改预下单支付-微信预下单结果通知添加营销详情promotion_detail响应参数
- 修改刷卡支付-订单查询接口添加营销详情promotion_detail响应参数
- 修改交易退款接口添加应付金额base_fee, 应退金额base_refund_fee; refund_fee字段修改为实退金额
- 修改退款查询接口添加应退金额汇总base_refund_fee_summary, 应退金额base_refund_fee响应字段
- 修改支付类接口响应参数添加微信服务商商户号wx_mch_id字段, 微信子商户号wx_sub_mch_id字段 (具体参考各接口响应参数)

【v2.4.1】2019-04-10

- 获取超赢商户微信子商户号接口说明修改

【v2.4.0】2019-04-03

- 添加获取超赢商户微信子商户号接口

【v2.3.7】2019-03-20

- 微信预下单接口添加pid和version请求参数
- 支付宝预下单接口添加pid和version请求参数

【v2.3.6】 2019-01-17

- 修改刷卡支付-交易退款接口移除scene请求参数
- 修改刷卡支付-交易撤销接口移除scene请求参数

【v2.3.5】 2019-01-09

- 修改刷脸支付-提交支付接口移除sub_openid请求参数

【v2.3.4】 2018-12-28

- 修改获取SDK调用凭证接口移除store_id响应参数

【v2.3.3】 2018-12-28

- 修改获取SDK调用凭证接口添加mch_id和paytype请求参数
- 修改刷脸支付接口添加paytype请求参数

【v2.3.2】 2018-12-27

- 修改刷脸支付接口face_code参数描述

【v2.3.1】 2018-12-27

- 添加获取SDK调用凭证接口
- 修改刷脸支付接口添加openid和sub_openid请求参数

【v2.3.0】 2018-12-20

- 添加刷脸支付接口

【v2.2.2】 2018-10-23

- 获取超赢商户号接口响应参数添加商户全称和商户简称

【v2.2.1】 2018-10-22

- 微信预下单接口添加门店编号参数
- 支付宝预下单接口添加门店编号参数

【v2.2.0】 2018-10-19

- 刷卡支付-提交支付接口移除paytype参数

【v2.1.2】 2018-10-18

- 更新获取超赢商户号接口

【v2.1.1】 2018-09-13

- 添加下发获取超赢商户号验证短信接口
- 更新获取超赢商户号接口

【v2.0.1】 2018-08-23

- 刷卡支付接口示例报文修改

【v2.0.0】 2018-08-16

- 在线文档改版

【v1.0.18】 2018-07-02

- 微信公众号、小程序、APPx2支付时sub_appid改为必传项

【v1.0.17】 2018-06-27

- 商户进件接口地址修改
- 商户进件接口添加门头照和内景照参数

© 超赢科技 all right reserved, powered by Gitbook文件修订时间 2019-08-05 14:23:11

- 协议规则
- 签名验证
- 请求格式
- 响应格式

协议规则

传输方式	为保证交易安全性，采用HTTPS传输（开发时可使用HTTP）
数据格式	若无指定，默认返回数据为JSON格式（application/json）
字符编码	统一采用UTF-8字符编码
内容类型	统一采用x-www-form-urlencoded编码格式
签名算法	MD5，后续会兼容SHA1、SHA256、HMAC等
签名要求	部份接口需要校验签名

签名验证

签名生成的通用步骤如下：

第一步，设所有传输的数据为集合M，将集合M内非空参数值的参数按照参数名ASCII码从小到大排序（字典序），使用URL键值对的格式（即key1=value1&key2=value2...）拼接成字符串stringA。

特别注意以下重要规则：

- 参数名ASCII码从小到大排序（字典序）；
- 如果参数的值为空不参与签名；
- 参数名区分大小写；
- 验证调用返回或主动通知签名时，传送的sign参数不参与签名，将生成的签名与该sign值作校验；
- 接口可能增加字段，验证签名时必须支持增加的扩展字段。

第二步，在stringA最后拼接上key得到stringSignTemp字符串（即stringA&key={KEY}），并对stringSignTemp进行MD5运算，再将得到的字符串所有字符转换为大写，得到sign值signValue。

请求格式

请求示例

```
// Javascript
var settings = {
  "async": true,
  "crossDomain": true,
  "url": "http://{BaseURL}/UnifiedPay/Gateway",
  "method": "POST",
  "headers": {
    "content-type": "application/x-www-form-urlencoded",
    "cache-control": "no-cache",
    "X-Requested-With": "XMLHttpRequest",
  },
  "data": {
    "method": "pay",
  },
}
```

```

    "mch_id": "00000001",
    "sign", "00000000000000000000000000000000"
  }
}

$.ajax(settings).done(function (response) {
  console.log(response);
});

```

```

# Python
import requests

url = "http://{BaseURL}/UnifiedPay/Gateway"

payload = "mch_id=00000001&method=pay&sign=00000000000000000000000000000000"
headers = {
  'content-type': "application/x-www-form-urlencoded",
  'cache-control': "no-cache",
  'X-Requested-With': "XMLHttpRequest",
}

response = requests.request("POST", url, data=payload, headers=headers)
print(response.text)

```

响应格式

字段名	必填	类型	说明
state	是	String	通讯状态，详见参数规定
code	是	String	状态码，详见参数规定
msg	否	String	返回信息；若调用失败则为错误原因
trade_state	否	String	交易状态，详见参数规定
sign	是	String	响应结果的签名串

成功示例

```

{
  "state": "SUCCESS",
  "code": "10000",
  "msg": "SUCCESS",
  "trade_state": "SUCCESS",
  "sign": "00000000000000000000000000000000"
}

```

失败示例

```

{
  "state": "FAIL",
  "code": "10002",
  "msg": "签名错误",
  "sign": "00000000000000000000000000000000"
}

```


- [state](#) - 通讯状态
- [code](#) - 状态码
- [scene](#) - 支付场景
- [trade_state](#) - 交易状态
- [paytype](#) - 支付方式
- [picType](#) - 进件图片类型
- [typeId](#) - 图像识别类型
- [DealType](#) - 经营类型
- [LegalPersonType](#) - 营业执照持有人类型
- [LegalPersonCertType](#) - 法人证件类型
- ([QrCodeMchApply](#) - 商户进件) [flag_type](#) - 商户类型
- [AccountType](#) - 结算账户类型
- [IdCardType](#) - 结算账户银行卡持卡人证件类型
- [PaymentChannelId](#) - 支付通道编码

编码统一使用大写

state - 通讯状态

编码	描述
SUCCESS	成功
FAIL	失败
VER	版本过低

code - 状态码

编码	描述
10000	成功
10001	支付中
10002	转入退款
10003	未支付
10004	已关闭
10005	已撤销
10006	支付错误
20002	签名错误
20003	响应结果签名错误
20004	参数错误
20005	商户错误
20006	系统错误
20007	订单不存在

20008	退款错误
20009	交易查询错误
20010	退款查询错误
20011	交易撤销错误
20012	代理商错误
20013	商户交易配置错误
30001	网络异常

scene - 支付场景

编码	描述
1	刷卡支付
2	公众号/服务窗支付
3	APP支付
4	原生扫码支付

trade_state - 交易状态

编码	描述
SUCCESS	成功
FAIL	失败
PROCESSING	处理中
NEED_QUERY	需要查询
REFUND	转入退款

paytype - 支付方式

编码	描述
WECHAT	微信支付
ALIPAY	支付宝支付
JDPAY	京东支付
QQ	QQ钱包支付
UNIONPAY	银联支付
BESTPAY	翼支付

picType - 进件图片类型

编码	描述
1	二代身份证
2	营业执照
3	组织机构证件照
4	商户协议照

typeld - 图像识别类型

编码	描述
2	二代身份证正面
3	二代身份证背面
17	银行卡
2008	营业执照

DealType - 经营类型

编码	描述
0	实体
1	虚拟

LegalPersonType - 营业执照持有人类型

编码	描述
0	个人
1	企业

LegalPersonCertType - 法人证件类型

编码	描述
0	身份证
1	护照

（QrCodeMchApply - 商户进件）flag_type - 商户类型

编码	描述
0	个体户
1	企业
2	政府及事业单位

AccountType - 结算账户类型

编码	描述
1	企业
2	个人

IdCardType - 结算账户银行卡持卡人证件类型

编码	描述
1	身份证
2	护照

PaymentChannelId - 支付通道编码

编码	描述
SPDB	浦发
CEB	光大
BESTPAY	翼支付
TTG	淘淘谷
JDPAY	京东钱包
UNIONPAY	银联钱包
WECHAT	微信
ALIPAY	支付宝
MUNITEDPASS	汇商通盈
SAND	杉德

© 超赢科技 all right reserved, powered by Gitbook文件修订时间 2018-12-20 15:18:10

- [获取SDK调用凭证](#)
- [提交支付](#)

刷脸支付

获取SDK调用凭证

应用场景

获取SDK调用凭证。

接口详情

接口地址	https://{BaseURL}/UnifiedPay/Gateway
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
method	是	getfaceauth	接口名称，getfaceauth
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号
pid	是	yunpos	调用方产品名称
sign	是	00000000000000000000000000000000	请求参数的签名串

请求参数

参数	必填	示例值	说明
paytype	是	WECHAT	支付方式，详见参数规定
mch_id	是	00000001	超赢商户号
store_id	是	test_store_01	门店编号
store_name	是	测试门店01	门店名称
device_id	是	00000001	终端设备编号
rawdata	是	00000001	初始化数据

请求参数示例

```
method=getfaceauth&agent_id=130000000000000000&paytype=WECHAT&mch_id=00000001&version=1.0&pid=yunpos&store_id=test_store_01&store_name=%E6%B5%8B%E8%AF%95%E9%97%A8%E5%BA%9701&device_id=CYW00000000000000000000&rawdata=eNPJNgR26U88XXXXXXXXXX&sign=00000000000000000000000000000000
```

响应结果

字段名	必填	说明
state	是	通讯状态，详见参数规定
code	是	状态码，详见参数规定
msg	否	返回信息
trade_state	否	交易状态，详见参数规定
sign	是	响应结果的签名串

以下字段在state和trade_state都为SUCCESS的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号
appid	是	服务商appid
sub_appid	否	子商户sub_appid(服务商模式)
channel_mch_id	是	服务商商户号
sub_mch_id	否	子商户号(服务商模式)
authinfo	是	SDK调用凭证
expires_in	是	authinfo的有效时间，单位秒。例如： 3600 。在有效时间内, 对于同一台终端设备，相同的参数的前提下(如：相同的公众号、商户号、门店编号等)，可以用同一个authinfo，多次调用SDK的getWxpayfaceCode接口。
nonce_str	是	随机字符串
wx_mch_id	否	微信服务商商户号
wx_sub_mch_id	否	微信子商户号

响应结果示例

```

{
  "state": "SUCCESS",
  "code": "10000",
  "trade_state": "SUCCESS",
  "msg": "SUCCESS",
  "mch_id": "00000001",
  "appid": "wx2b029c08a6232582",
  "channel_mch_id": "1900007071",
  "authinfo": "IGJSU0kyjStHQ0nWkI49U6LGhiJbTpr2KkM0iD2kb16DRmyySXXXXXX",
  "expires_in": "3600",
  "nonce_str": "k1MErGrCODf06uiv",
  "wx_mch_id": "1264300000",
  "wx_sub_mch_id": "1266500000",
  "sign": "00000000000000000000000000000000"
}
```

提交支付

应用场景

刷脸支付可用于线下各个场景，尤其对于不方便拿出手机支付等场景尤为方便。在支持刷脸支付的机具上选择使用微信刷脸支付付款，只需刷一下脸然后输入本人微信绑定的手机号即可完成付款，方便快捷又安全。

接口详情

接口地址	https://{BaseURL}/UnifiedPay/Gateway
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
method	是	facepay	接口名称，facepay
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号
pid	是	yunpos	调用方产品名称
sign	是	00	请求参数的签名串

请求参数

参数	必填	示例值	说明
paytype	是	WECHAT	支付方式，详见参数规定
mch_id	是	00000001	超赢商户号
openid	是	00000001	用户标识
out_trade_no	是	1497769914931	商户系统内部的订单号,5到32个字符、只能包含字母数字或者下划线，区分大小写，确保在商户系统唯一
device_info	否	013467007045764	终端设备号，商户自定义。特别说明：对于QQ钱包支付，此参数必传，否则会报错。
body	是	image形象店-深圳腾大- QQ公仔	商品描述
attach	否	说明	商户附加信息，可做扩展参数
total_fee	是	1	总金额，以分为单位，只能为整数
mch_create_ip	否	114.114.114.114	调用支付API的机器IP
face_code	是	120061098828009406	人脸凭证
op_user_id	否	00000001	操作员帐号，默认为商户号
op_shop_id	否	md_001	门店编号
op_device_id	否	device_01	设备编号
goods_tag	否	CY_PROMOTION_001	订单优惠标记，用于优惠券或者满减使用
		[{"goods_id":"CY000","goods_name":"促	

		销单品","quantity":1,"price":1}}]	
--	--	--------------------------------	--

goods_detail为JSON数组类型结构如下

参数	必填	示例值	说明
goods_id	是	CY000000000001	商品编码，由半角的大小写字母、数字、中划线、下划线中的一种或几种组成
pay_goods_id	否	20010001	微信/支付宝的商品编码（没有可不传）
goods_name	否	纸巾	商品名称
quantity	是	1	商品数量（整数）
price	是	100	商品单价（整数），单位为：分

请求参数示例

```
method=facepay&agent_id=13000000000000000000&paytype=WECHAT&mch_id=00000001&version=1.0&pid=yunpos&out_trade_no=1497769914931&face_code=123123123&body=%E8%B6%85%E8%B5%A2%E6%94%AF%E4%BB%98&total_fee=4&goods_tag=CY_PROMOTION_001&goods_detail=[{"goods_id":"CY000000","goods_name":"促销单品-CY000000000000","quantity":1,"price":2}, {"goods_id":"CY000001","goods_name":"促销单品-CY000000000001","quantity":1,"price":2}]]&sign=00000000000000000000000000000000
```

响应结果

字段名	必填	说明
state	是	通讯状态，详见参数规定
code	是	状态码，详见参数规定
msg	否	返回信息
trade_state	否	交易状态，详见参数规定
sign	是	响应结果的签名串

以下字段在state和trade_state都为SUCCESS的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号
appid	是	调用接口提交的公众账号ID
is_subscribe	是	用户是否关注公众账号，仅在公众账号类型支付有效，取值范围：Y或N；Y-关注；N-未关注
openid	是	用户在商户 appid 下的唯一标识
sub_appid	是	调用接口提交的子商户公众账号ID
sub_is_subscribe	是	用户是否关注子公众账号，仅在公众账号类型支付有效，取值范围：Y或N；Y-关注；N-未关注
sub_openid	是	子商户appid下用户唯一标识，如需返回则请求时需要传sub_appid
transaction_id	是	平台交易号

transaction_id	是	平台交易号
out_transaction_id	是	第三方订单号
out_trade_no	是	商户系统内部的定单号，32个字符内、可包含字母
base_fee	是	应付金额，以分为单位，只能为整数
total_fee	是	实付金额，以分为单位，只能为整数
coupon_fee	否	代金券金额，代金券金额<=订单金额，订单金额 - 代金券金额 = 现金支付金额
fee_type	否	货币类型，符合 ISO 4217 标准的三位字母代码，默认人民币：CNY
attach	否	商家数据包，原样返回
bank_type	否	付款银行
bank_billno	否	银行订单号，若为第三方支付则为空
time_end	是	支付完成时间，格式为yyyyMMddHHmmss，如2009年12月25日9点10分10秒表示为20091225091010。时区为GMT+8 Beijing
nonce_str	是	随机字符串
promotion_detail	否	营销详情，返回值为Json格式
wx_mch_id	否	微信服务商商户号
wx_sub_mch_id	否	微信子商户号

响应结果示例

```
{
  "state": "SUCCESS",
  "code": "10000",
  "trade_state": "SUCCESS",
  "msg": "SUCCESS",
  "mch_id": "00000001",
  "appid": "wx2b029c08a6232582",
  "is_subscribe": "N",
  "openid": "oBmIts3wbluEiYfp6zDgBWsC9Evs",
  "transaction_id": "4200000233201812272744871942",
  "out_transaction_id": "4200000233201812272744871942",
  "out_trade_no": "1545890307",
  "base_fee": "4",
  "total_fee": "2",
  "fee_type": "CNY",
  "bank_type": "CFT",
  "time_end": "20181227140034",
  "nonce_str": "k1MErGrCODf06uiv",
  "promotion_detail": "{\n\"promotion_detail\": [{\n\"promotion_id\": \"6348962444\", \"name\": \"\n维他减2分\", \"scope\": \"S\nINGLE\", \"type\": \"DISCOUNT\", \"amount\": 2, \"activity_id\": \"9447213\", \"wxpay_contribute\": 0, \"merchant_contribute\": 2, \"other_contribute\": 0, \"goods_detail\": [{\n\"goods_id\": \"CY0000000000\", \"quantity\": 1, \"price\": 2, \"discount_amo\nunt\": 1, \"goods_remark\": \"\n单品券活动No.002\", {\n\"goods_id\": \"CY0000000001\", \"quantity\": 1, \"price\": 2, \"discount_\namount\": 1, \"goods_remark\": \"\n单品券活动No.002\"}]}]}",
  "wx_mch_id": "1264300000",
  "wx_sub_mch_id": "1266500000",
  "sign": "00000000000000000000000000000000"
}
```


- [提交支付](#)
- [订单查询](#)
- [交易退款](#)
- [退款查询](#)
- [交易撤销](#)

刷卡支付

提交支付

应用场景

收银员使用扫码设备读取微信（或支付宝等电子钱包）用户刷卡授权码以后，调用该接口发起支付对用户进行收款。

接口详情

接口地址	https://{BaseURL}/UnifiedPay/Gateway
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
method	是	pay	接口名称，pay
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号
pid	是	yunpos	调用方产品名称
sign	是	00000000000000000000000000000000	请求参数的签名串

请求参数

参数	必填	示例值	说明
mch_id	是	00000001	超赢商户号
out_trade_no	是	1497769914931	商户系统内部的订单号,5到32个字符、只能包含字母数字或者下划线,区分大小写,确保在商户系统唯一
device_info	否	013467007045764	终端设备号,商户自定义。特别说明:对于QQ钱包支付,此参数必传,否则会报错。
body	是	image形象店-深圳腾大- QQ公仔	商品描述
attach	否	说明	商户附加信息,可做扩展参数
total_fee	是	1	总金额,以分为单位,只能为整数

mch_create_ip	否	114.114.114.114	调用支付API的机器IP
auth_code	是	120061098828009406	扫码支付授权码，设备读取用户展示的条码信息
op_user_id	否	00000001	操作员帐号，默认为商户号
op_shop_id	否	md_001	门店编号
op_device_id	否	device_01	设备编号
goods_tag	否	hot	订单优惠标记，用于优惠券或者满减使用
goods_detail	否	[[{"goods_id":"CY000","goods_name":"促销单品","quantity":1,"price":1}]]	商品详情，JSON Array格式

goods_detail为JSON数组类型结构如下

参数	必填	示例值	说明
goods_id	是	CY000000000001	商品编码，由半角的大小写字母、数字、中划线、下划线中的一种或几种组成
pay_goods_id	否	20010001	微信/支付宝的商品编码（没有可不传）
goods_name	否	纸巾	商品名称
quantity	是	1	商品数量（整数）
price	是	100	商品单价（整数），单位为：分

请求参数示例

```
method=pay&agent_id=13000000000000000000&mch_id=00000001&version=1.0&pid=yunpos&out_trade_no=1497769914931&auth_code=123123123&body=%E8%B6%85%E8%B5%A2%E6%94%AF%E4%BB%98&total_fee=4&goods_tag=CY_PROMOTION_001&goods_detail=[[{"goods_id":"CY000000","goods_name":"促销单品-CY000000000000","quantity":1,"price":2}, {"goods_id":"CY000001","goods_name":"促销单品-CY000000000001","quantity":1,"price":2}]]&sign=00000000000000000000000000000000
```

响应结果

字段名	必填	说明
state	是	通讯状态，详见参数规定
code	是	状态码，详见参数规定
msg	否	返回信息
trade_state	否	交易状态，详见参数规定
sign	是	响应结果的签名串

以下字段在state和trade_state都为SUCCESS的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号

appid	是	调用接口提交的公众账号ID
is_subscribe	是	用户是否关注公众账号，仅在公众账号类型支付有效，取值范围：Y或N；Y-关注；N-未关注
openid	是	用户在商户 appid 下的唯一标识
sub_appid	是	调用接口提交的子商户公众账号ID
sub_is_subscribe	是	用户是否关注子公众账号，仅在公众账号类型支付有效，取值范围：Y或N；Y-关注；N-未关注
sub_openid	是	子商户appid下用户唯一标识，如需返回则请求时需要传sub_appid
transaction_id	是	平台交易号
out_transaction_id	是	第三方订单号
out_trade_no	是	商户系统内部的定单号，32个字符内、可包含字母
base_fee	是	订单应付金额，单位为分
total_fee	是	订单实付金额，单位为分
coupon_fee	否	代金券金额，代金券金额<=订单金额，订单金额 - 代金券金额 = 现金支付金额
fee_type	否	货币类型，符合 ISO 4217 标准的三位字母代码，默认人民币：CNY
attach	否	商家数据包，原样返回
bank_type	否	付款银行
bank_billno	否	银行订单号，若为第三方支付则为空
time_end	是	支付完成时间，格式为yyyyMMddHHmmss，如2009年12月25日9点10分10秒表示为20091225091010。时区为GMT+8 Beijing
nonce_str	是	随机字符串
promotion_detail	否	营销详情，返回值为Json格式
wx_mch_id	否	微信服务商商户号
wx_sub_mch_id	否	微信子商户号

响应结果示例

```
{
  "state": "SUCCESS",
  "code": "10000",
  "trade_state": "SUCCESS",
  "msg": "SUCCESS",
  "mch_id": "00000001",
  "appid": "wx1f87d44db95cba7a",
  "is_subscribe": "N",
  "openid": "oywgtuCJFeGzT-QtF-8U7FHb1z3Q",
  "sub_appid": "wxce38685bc050ef82",
  "sub_is_subscribe": "N",
  "sub_openid": "oHmbktxFlpoEPo2015G0JniV2q-A",
  "transaction_id": "7551000001201706196281085687",
  "out_transaction_id": "4005572001201706196460269701",
  "out_trade_no": "1497862554883",
  "base_fee": "1",
  "total_fee": "1",
  "fee_type": "CNY",
  "bank_type": "CFT",
  "time_end": "20170619165616",
```

```

"nonce_str": "a849df6660cb4354b6fe5b23120a73ce",
"promotion_detail": "{\n\"promotion_detail\": [{\n\"promotion_id\": \"6348962444\", \"name\": \"维他减2分\", \"scope\": \"SINGLE\", \"type\": \"DISCOUNT\", \"amount\": 2, \"activity_id\": \"9447213\", \"wxpay_contribute\": 0, \"merchant_contribute\": 2, \"other_contribute\": 0, \"goods_detail\": [{\n\"goods_id\": \"CY0000000000\", \"quantity\": 1, \"price\": 2, \"discount_amount\": 1, \"goods_remark\": \"单品券活动No.002\"}, {\n\"goods_id\": \"CY0000000001\", \"quantity\": 1, \"price\": 2, \"discount_amount\": 1, \"goods_remark\": \"单品券活动No.002\"}]}]}",
"wx_mch_id": "1264300000",
"wx_sub_mch_id": "1266500000",
"sign": "00000000000000000000000000000000"
}

```

订单查询

应用场景

根据商户订单号或者平台订单号查询平台的具体订单信息。

接口详情

接口地址	https://{BaseURL}/UnifiedPay/Gateway
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
method	是	orderquery	接口名称，orderquery
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号
pid	是	yunpos	调用方产品名称
sign	是	00000000000000000000000000000000	请求参数的签名串

请求参数

参数	必填	示例值	说明
paytype	是	WECHAT	支付方式，详见参数规定
mch_id	是	00000001	超赢商户号
out_trade_no	否	1497769914931	商户系统内部的订单号，out_trade_no和transaction_id至少一个必填，同时存在时out_trade_no优先
transaction_id	否	7551000001201706166172780576	平台交易号，out_trade_no和transaction_id至少一个必填，同时存在时out_trade_no优先。

请求参数示例

```

method=orderquery&agent_id=130000000000000000&version=1.0&pid=yunpos&paytype=WECHAT&mch_id=00000001&out_trade_no=1497769914931&sign=00000000000000000000000000000000

```

响应结果

字段名	必填	说明
state	是	通讯状态，详见参数规定
code	是	状态码，详见参数规定
msg	否	返回信息
trade_state	否	交易状态，详见参数规定
sign	是	响应结果的签名串

以下字段在state和trade_state都为SUCCESS的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号
appid	是	调用接口提交的公众账号ID
is_subscribe	是	用户是否关注公众账号，仅在公众账号类型支付有效，取值范围：Y或N;Y-关注;N-未关注
openid	是	用户在商户 appid 下的唯一标识
sub_appid	是	调用接口提交的子商户公众账号ID
sub_is_subscribe	是	用户是否关注子公众账号，仅在公众账号类型支付有效，取值范围：Y或N;Y-关注;N-未关注
sub_openid	是	子商户appid下用户唯一标识，如需返回则请求时需要传sub_appid
transaction_id	是	平台交易号
out_transaction_id	是	第三方订单号
out_trade_no	是	商户系统内部的定单号，32个字符内、可包含字母
base_fee	是	应付金额、订单金额，以分为单位，只能为整数
total_fee	是	实付金额，以分为单位，只能为整数
coupon_fee	否	代金券金额，代金券金额<=订单金额，订单金额 - 代金券金额 = 现金支付金额
fee_type	否	货币类型，符合 ISO 4217 标准的三位字母代码，默认人民币：CNY
attach	否	商家数据包，原样返回
bank_type	否	付款银行
bank_billno	否	银行订单号，若为第三方支付则为空
time_end	是	支付完成时间，格式为yyyyMMddHHmmss，如2009年12月25日9点10分10秒表示为20091225091010。时区为GMT+8 Beijing
nonce_str	是	随机字符串
trade_type	是	交易类型
promotion_detail	否	营销详情，返回值为Json格式

响应结果示例

```
{
  "state": "SUCCESS",
  "code": "10000",
  "trade_state": "SUCCESS",
  "msg": "SUCCESS",
  "mch_id": "00000001",
  "appid": "wx4da448cd29920000",
  "is_subscribe": "Y",
  "openid": "o4he1jo7fA1rIWT00A3hDbGwc29w",
  "sub_appid": "wx4da448cd29920000",
  "sub_is_subscribe": "Y",
  "sub_openid": "o4he1jo7fA1rIWT00A3hDbGwc29w",
  "transaction_id": "4200000334201905246610520000",
  "out_transaction_id": "4200000334201905246610520000",
  "out_trade_no": "T0020190524102840000",
  "base_fee": "4",
  "total_fee": "2",
  "coupon_fee": "2",
  "fee_type": "CNY",
  "bank_type": "CFT",
  "time_end": "20190524103044",
  "nonce_str": "1CQmxHpi2H62HsGg",
  "trade_type": "MICROPAY",
  "promotion_detail": "{\\"promotion_detail\\": [{\\"promotion_id\\":\\"6348962444\\",\\"name\\":\\"维他减2分\\",\\"scope\\":\\"SINGLE\\",\\"type\\":\\"DISCOUNT\\",\\"amount\\":2,\\"activity_id\\":\\"9447213\\",\\"wxpay_contribute\\":0,\\"merchant_contribute\\":2,\\"other_contribute\\":0,\\"goods_detail\\": [{\\"goods_id\\":\\"CY0000000000\\",\\"quantity\\":1,\\"price\\":2,\\"discount_amount\\":1,\\"goods_remark\\":\\"单品券活动No.002\\"}, {\\"goods_id\\":\\"CY0000000001\\",\\"quantity\\":1,\\"price\\":2,\\"discount_amount\\":1,\\"goods_remark\\":\\"单品券活动No.002\\"}]}]}",
  "wx_mch_id": "1264300000",
  "wx_sub_mch_id": "1266500000",
  "sign": "00000000000000000000000000000000"
}
```

交易退款

应用场景

商户针对某一个已经成功支付的订单发起退款，操作结果在同一会话中同步返回。

退款方式

目前只支持原路返回退款

说明：退到银行卡则是非实时的，每个银行的处理速度不同，一般发起退款后1-3个工作日内到账。

同一笔单的部分退款需要设置相同的订单号和不同的 **out_refund_no**。一笔退款失败后重新提交，要采用原来的 **out_refund_no**。总退款金额不能超过用户实际支付金额(代金券金额不能退款)

退款限制

商户在退款操作时应该注意退款限制，避免发起不会成功的退款请求，下面是主要的退款限制：

1. 在平台系统中，只要退款累计金额不超过交易单支付总额，一笔交易单可以多次退款，退款申请单号（退款接口中有此参数）唯一确定一次退款，而不是交易单号确定一次退款。退款申请单号由商户生成，所以商户一定要保证退款申请单的唯一性。商家在退款过程中要特别注意，只有在能确定退款失败的情况下，才能重新发起另一笔退款。
2. 目前大多数银行都支持全额退款和部分退款，但是也有少数银行不支持全额退款或部分退款，或者不支持退款。在这种情况下，商户可以与买家协调，线下直接退款给买家。

接口详情

接口地址	https://{BaseURL}/UnifiedPay/Gateway
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
method	是	refund	接口名称, refund
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号
pid	是	yunpos	调用方产品名称
sign	是	00000000000000000000000000000000	请求参数的签名串

请求参数

参数	必填	示例值	说明
paytype	是	WECHAT	支付方式, 详见参数规定
mch_id	是	00000001	超赢商户号
out_trade_no	否	1497769914931	商户系统内部的订单号, out_trade_no和transaction_id至少一个必填, 同时存在时transaction_id优先
transaction_id	否	7551000001201706166172780576	平台单号, out_trade_no和transaction_id至少一个必填, 同时存在时transaction_id优先
out_refund_no	是	TK-1497769914931-01	商户退款单号, 32个字符内、可包含字母, 确保在商户系统唯一。同个退款单号多次请求, 平台当一个单处理, 只会退一次款。如果出现退款不成功, 请采用原退款单号重新发起, 避免出现重复退款。
total_fee	是	1	订单应付金额, 单位为分
refund_fee	是	1	申请退款金额, 单位为分
op_user_id	是	00000001	操作员帐号, 默认为商户号

请求参数示例

```
method=refund&agent_id=130000000000000000&version=1.0&pid=yunpos&paytype=WECHAT&mch_id=00000001&out_trade_no=1497769914931&out_refund_no=TK-1497769914931-01&sign=00000000000000000000000000000000
```

响应结果

字段名	必填	说明
state	是	通讯状态, 详见参数规定
code	是	状态码, 详见参数规定

msg	否	返回信息
trade_state	否	交易状态，详见参数规定
sign	是	响应结果的签名串

以下字段在state和trade_state都为SUCCESS的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号
out_trade_no	是	商户系统内部的定单号，32个字符内、可包含字母
transaction_id	是	平台交易号
out_transaction_id	是	第三方退款单号
out_refund_no	是	商户退款单号
refund_id	是	平台退款单号
refund_channel	否	退款渠道，ORIGINAL—原路退款，默认
base_fee	是	订单应付金额，单位为分
total_fee	是	订单实付金额，单位为分
base_refund_fee	是	申请退款金额，单位为分
refund_fee	是	实际退款金额，单位为分
coupon_refund_fee	是	代金券退款金额 <= 退款金额，退款金额-代金券退款金额为现金
nonce_str	是	随机字符串

响应结果示例

```
{
  "state": "SUCCESS",
  "code": "10000",
  "trade_state": "SUCCESS",
  "msg": "SUCCESS",
  "mch_id": "00000001",
  "out_trade_no": "T0020190517145341153",
  "transaction_id": "4200000330201905173095298659",
  "out_transaction_id": "4200000330201905173095298659",
  "out_refund_no": "TKT0020190517145341153-B",
  "refund_id": "50000300412019051709584177333",
  "base_fee": "8",
  "total_fee": "6",
  "base_refund_fee": "2",
  "refund_fee": "1",
  "coupon_refund_fee": "1",
  "nonce_str": "78GTQmdy1SxwFXxE",
  "sign": "00000000000000000000000000000000"
}
```

退款查询

应用场景

注意

- ## 接口详情

接口地址	https://{BaseURL}/UnifiedPay/Gateway
提交方式	POST
校验签名	是

[illegible]

参数	必填	示例值	说明
paytype	是	WECHAT	支付方式，详见参数规定
mch_id	是	00000001	超赢商户号
out_refund_no	否	TK-1497769914931-01	商户退款单号，32个字符内、可包含字母,确保在商户系统唯一。
refund_id	否	7551000001201706215157548269	平台退款单号关于refund_id、out_refund_no必填一个，如果同时存在优先级为：refund_id > out_refund_no；特殊说明：如果是支付宝，refund_id、out_refund_no必填其中一个。

method=refundquery&agent_id=13000000000000000&version=1.0&pid=yunpos&paytype=WCHAT&mch
_id=00000001&out_refund_no=TKT0020190524102840000&sign=00000000000000000000000000000000
0

字段名	必填	说明
state	是	通讯状态，详见参数规定
code	是	状态码， 详见参数规定
msg	否	返回信息

trade_state	否	交易状态，详见参数规定
sign	是	响应结果的签名串

以下字段在state和trade_state都为SUCCESS的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号
appid	否	调用接口提交的公众账号ID
is_subscribe	否	用户是否关注公众账号，仅在公众账号类型支付有效，取值范围：Y或N;Y-关注;N-未关注
openid	否	用户在商户 appid 下的唯一标识
sub_appid	否	调用接口提交的子商户公众账号ID
sub_is_subscribe	否	用户是否关注子公众账号，仅在公众账号类型支付有效，取值范围：Y或N;Y-关注;N-未关注
sub_openid	否	子商户appid下用户唯一标识，如需返回则请求时需要传sub_appid
transaction_id	是	平台交易号
out_transaction_id	是	第三方订单号
out_trade_no	是	商户系统内部的定单号，32个字符内、可包含字母
refund_count	是	退款笔数
base_refund_fee_summary	是	申请退款汇总金额，以分为单位，只能为整数
refund_fee_summary	是	实际退款汇总金额，以分为单位，只能为整数
refund_list	是	退款单集合
nonce_str	是	随机字符串
wx_mch_id	否	微信服务商商户号
wx_sub_mch_id	否	微信子商户号

以下字段在refund_list中返回

字段名	必填	说明
serial_no	是	退款序号，按退款申请时间升序排列
refund_id	是	平台退款单号
out_refund_id	是	第三方退款单号
out_refund_no	是	商户退款单号
refund_channel	否	退款渠道，ORIGINAL—原路退款，默认
base_refund_fee	是	当次退款申请金额，以分为单位，只能为整数
refund_fee	是	当次实际退款金额，以分为单位，只能为整数
coupon_refund_fee	否	代金券退款金额 <= 退款金额，退款金额 - 代金券退款金额为现金

refund_status	是	退款状态：SUCCESS—退款成功；FAIL—退款失败；PROCESSING—退款处理中；NOTSURE—未确定，需要商户原退款单号重新发起；CHANGE—转入代发，退款到银行发现用户的卡作废或者冻结了，导致原路退款银行卡失败，资金回流到商户的现金帐号，需要商户人工干预，通过线下或者平台转账的方式进行退款。
refund_time	否	退款时间，格式为yyyyMMddHHmmss，如2009年12月25日9点10分10秒表示为20091225091010。时区为GMT+8 Beijing

响应结果示例

```
{
  "state": "SUCCESS",
  "code": "10000",
  "trade_state": "SUCCESS",
  "msg": "SUCCESS",
  "mch_id": "00000001",
  "appid": "wx4da448cd29920000",
  "transaction_id": "4200000334201905246610520000",
  "out_transaction_id": "4200000334201905246610520000",
  "out_trade_no": "T0020190524102840000",
  "base_refund_fee_summary": "4",
  "refund_fee_summary": "2",
  "refund_count": "1",
  "refund_list": "[{\"serial_no\": \"0\", \"refund_id\": \"50000000482019052409653860000\", \"out_refund_id\": \"50000000482019052409653860000\", \"out_refund_no\": \"TKT0020190524102840000\", \"base_refund_fee\": \"4\", \"refund_fee\": \"2\", \"coupon_refund_fee\": \"2\", \"refund_status\": \"SUCCESS\", \"refund_channel\": \"ORIGINAL\", \"refund_time\": \"2019-05-24 11:11:33\"}]",
  "nonce_str": "DjRzpkdtpM0S16HW",
  "wx_mch_id": "1264300000",
  "wx_sub_mch_id": "1266500000",
  "sign": "00000000000000000000000000000000"
}
```

交易撤销

应用场景

当支付返回失败，或收银系统超时需要取消交易，可以调用该接口。接口逻辑：支付失败的关单，支付成功的撤销支付。注意：5分钟的订单才可以撤销，其他正常支付的单如需实现相同功能请调用退款接口。

调用支付接口后请勿立即调用撤销订单接口，建议支付后至少**15s**后再调用撤销订单接口。

接口详情

接口地址	https://{BaseURL}/UnifiedPay/Gateway
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
method	是	cancel	接口名称
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号

pid	是	yunpos	调用方产品名称
sign	是	00000000000000000000000000000000	请求参数的签名串

请求参数

参数	必填	示例值	说明
paytype	是	WECHAT	支付方式，详见参数规定
mch_id	是	00000001	超赢商户号
out_trade_no	是	1497769914931	商户系统内部的订单号,5到32个字符、只能包含字母数字或者下划线，区分大小写，确保在商户系统唯一
op_user_id	是	00000001	操作员帐号

请求参数示例

```
method=cancel&agent_id=13000000000000000&version=1.0&pid=yunpos&paytype=WECHAT&mch_id=0000001&out_trade_no=1497769914931&op_user_id=yzq&sign=00000000000000000000000000000000
```

响应结果

字段名	必填	说明
state	是	通讯状态，详见参数规定
code	是	状态码，详见参数规定
msg	否	返回信息
trade_state	否	交易状态，详见参数规定
sign	是	响应结果的签名串

以下字段在state和trade_state都为SUCCESS的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号
nonce_str	是	随机字符串
wx_mch_id	否	微信服务商商户号
wx_sub_mch_id	否	微信子商户号

响应结果示例

```
{
  "state": "SUCCESS",
  "code": "10000",
  "trade_state": "SUCCESS",
  "msg": "SUCCESS",
  "mch_id": "00000001",
  "nonce_str": "f1625e94362d4d82aafcc5fc9d1f9325",
  "wx_mch_id": "1264300000",
  "wx_sub_mch_id": "1266500000",
  "sign": "00000000000000000000000000000000"
}
```


- [微信预下单](#)
- [支付宝预下单](#)
- [微信预下单结果通知](#)
- [支付宝预下单结果通知](#)

预下单支付

微信预下单

应用场景

用户通过微信公众号或扫描二维码，可以支付完成下单购买流程。

接口详情

接口地址	https://{BaseURL}/Open/WECHAT_PreOrder
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号
pid	是	yunpos	调用方产品名称
nonce_str	否	00000000000000000000000000000000	随机字符串
sign	是	00000000000000000000000000000000	请求参数的签名串

请求参数

参数	必填	示例值	说明
mch_id	是	00000001	超赢商户号
sub_appid	是	wxaee08d155d9de0c5	公众号AppId。 trade_type=JSPAY、APP、APP_BANK时，此参数必传
is_minipg	否	1	是否小程序支付。1：小程序支付；0：表示公众号支付
buyer_id	否	oRA5quGvJc1wwwjMW-tnpLLqD-FM	在sub_appid下通过微信授权取得的买家sub_openid。 trade_type=JSPAY，此参数必传
device_info	否	013467007045764	终端设备号，商户自定义。
body	是	image形象店-深圳腾大- QQ公仔	商品描述
out_trade_no	是	1497769914931	商户系统内部的订单号,5到32个字符、只能包含字母数字，区分大小

			写，确保在商户系统唯一
total_fee	是	1	总金额，以分为单位，只能为整数
mch_create_ip	否	114.114.114.114	调用支付API的机器IP
time_start	否	20091225091010	交易起始时间，格式为yyyyMMddHHmmss，如2009年12月25日9点10分10秒表示为20091225091010。时区为GMT+8 Beijing
time_expire	否	20091225091510	交易结束时间，格式为yyyyMMddHHmmss，如2009年12月25日9点15分10秒表示为20091225091510。时区为GMT+8 Beijing
notify_url	否	http://baidu.com/notify	接收支付异步通知回调地址，通知url必须为直接可访问的url，不能携带参数
op_user_id	否	00000001	操作员帐号，默认为商户号
op_shop_id	否	00000001	门店编号
limit_pay	否	NO_CREDIT	no_credit指定不能使用信用卡支付
trade_type	是	JSPAY	交易类型：JSPAY-服务窗支付、NATIVE-原生扫码支付、APP-APP支付、APP_BANK-APP支付银行模式
product_id	否	1223541321407035645	trade_type=NATIVE，此参数必传。此id为二维码中包含的商品ID，商户自行定义
goods_tag	否	CY_PROMOTION_001	订单优惠标记，用于优惠券或者满减使用
goods_detail	否	[[{"goods_id":"CY000","goods_name":"促销单品","quantity":1,"price":1}]]	商品详情，JSON Array格式

goods_detail为JSON数组类型结构如下

参数	必填	示例值	说明
goods_id	是	CY000000000001	商品编码，由半角的大小写字母、数字、中划线、下划线中的一种或几种组成
pay_goods_id	否	20010001	微信/支付宝的商品编码（没有可不传）
goods_name	否	纸巾	商品名称
quantity	是	1	商品数量（整数）
price	是	100	商品单价（整数），单位为：分

请求参数示例

```
agent_id=13000000000000000000&version=1.0&pid=yunpos&mch_id=00000001™_type=NATIVE&body=超
赢支付聚合支付
&out_trade_no=1497769914931&total_fee=4&goods_tag=CY_PROMOTION_001&goods_detail=
```



```
[{"goods_id":"CY000000","goods_name":"促销单品-CY000000000000","quantity":1,"price":2},
{"goods_id":"CY000001","goods_name":"促销单品-
CY000000000001","quantity":1,"price":2}]&sign=00000000000000000000000000000000
```

响应结果

字段名	必填	说明
state	是	通讯状态，详见参数规定
code	是	状态码，详见参数规定
msg	否	返回信息
trade_state	否	交易状态，详见参数规定

以下字段在state和trade_state都为SUCCESS的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号
appid	是	调用接口提交的公众账号ID
sub_openid	是	子商户appid下用户唯一标识，如需返回则请求时需要传sub_appid
out_trade_no	是	商户系统内部的定单号，32个字符内、可包含字母
token_id	否	动态口令
pay_info	否	支付信息，用以唤起微信支付；trade_type=JSPAY时，此参数有返回
code_img_url	否	此参数的值即是根据code_url生成的可以扫码支付的二维码图片地址；trade_type=NATIVE时，此参数有返回
code_url	否	商户可用此参数自定义去生成二维码后展示出来进行扫码支付；trade_type=NATIVE时，此参数有返回
services	否	支持的交易类型，多个以竖线分割
nonce_str	是	随机字符串
wx_mch_id	否	微信服务商商户号
wx_sub_mch_id	否	微信子商户号

响应结果示例

```
{
  "state": "SUCCESS",
  "code": "10000",
  "trade_state": "SUCCESS",
  "msg": "SUCCESS",
  "mch_id": "3338102513618944",
  "out_trade_no": "1497769914931",
  "token_id": "16c5f40274e084b123a08f1313d53da57",
  "pay_info": "{\"appId\":\"wxc3d2e734ae326831\",\"timeStamp\":\"1511947589427\",\"status\":\"0\",\"signType\":\"MD5\",\"package\":\"prepay_id=wx2017112917262941a3a33e980001282756\",\"callback_url\":\"\",\"nonceStr\":\"1511947589427\",\"paySign\":\"CD871BCD09B3D0C6339B2D0DE72DE7EE\"}",
  "wx_mch_id": "1264300000",
  "wx_sub_mch_id": "1266500000",
  "nonce_str": "00000000000000000000000000000000"
}
```

支付宝预下单

应用场景

用户通过支付宝服务窗或扫描二维码，可以支付完成下单购买流程。

接口详情

接口地址	https://{BaseURL}/Open/ALIPAY_PreOrder
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号
pid	是	yunpos	调用方产品名称
nonce_str	否	00000000000000000000000000000000	随机字符串
sign	是	00000000000000000000000000000000	请求参数的签名串

请求参数

参数	必填	示例值	说明
mch_id	是	00000001	超赢商户号
device_info	否	013467007045764	终端设备号，商户自定义。
body	是	image形象店-深圳腾大-QQ公仔	商品描述
out_trade_no	是	1497769914931	商户系统内部的订单号,5到32个字符、只能包含字母数字，区分大小写，确保在商户系统唯一
total_fee	是	1	总金额，以分为单位，只能为整数
mch_create_ip	否	114.114.114.114	调用支付API的机器IP
time_start	否	20091225091010	交易起始时间，格式为yyyyMMddHHmmss，如2009年12月25日9点10分10秒表示为20091225091010。时区为GMT+8 Beijing。注：交易起始时间与交易结束时间需要同时传入才会生效。
time_expire	否	20091225091510	交易结束时间，格式为yyyyMMddHHmmss，如2009年12月25日9点15分10秒表示为20091225091510。时区为GMT+8 Beijing。注：交易起始时间与交易结束时间需要同时传入才会生效。
goods_tag	否	CY_PROMOTION_001	订单优惠标记，用于优惠券或者满减使用

notify_url	否	http://baidu.com/notify	接收支付异步通知回调地址，通知url必须为直接可访问的url，不能携带参数
op_user_id	否	00000001	操作员帐号，默认为商户号
op_shop_id	否	00000001	门店编号
limit_pay	否	NO_CREDIT	no_credit指定不能使用信用卡支付
trade_type	是	NATIVE	交易类型；JSPAY-服务窗支付、NATIVE-原生扫码支付、APP-APP支付，APP_BANK-APP支付银行模式
product_id	否	1223541321407035645	trade_type=JSPAY时，此参数必传。此id为二维码中包含的商品ID，商户自行定义
buyer_logon_id	否	13800000000	买家支付宝账号。trade_type=JSPAY时，和buger_id不能同时为空
buyer_id	否	2088822494428872	买家支付宝用户ID。trade_type=JSPAY时，和buyer_logon_id不能同时为空

请求参数示例

```
agent_id=13000000000000000000&version=1.0&pid=yunpos&mch_id=00000001&body=超赢支付聚合支付
&out_trade_no=1497769914931&total_fee=1&trade_type=NATIVE&sign=0000000000000000000000000000000000000000
```

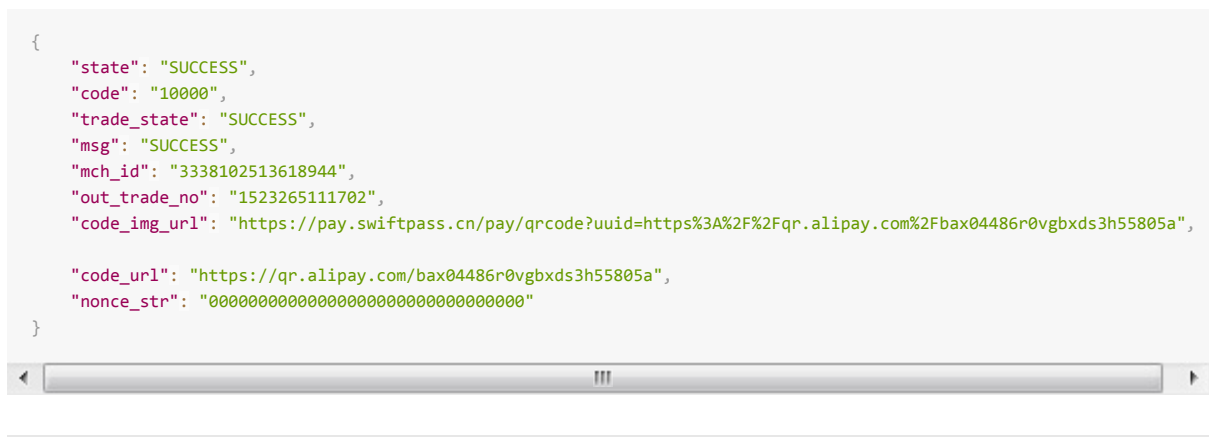
响应结果

字段名	必填	说明
state	是	通讯状态，详见参数规定
code	是	状态码，详见参数规定
msg	否	返回信息
trade_state	否	交易状态，详见参数规定

以下字段在state和trade_state都为SUCCESS的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号
out_trade_no	是	商户系统内部的定单号，32个字符内、可包含字母
pay_info	否	JSON字符串，自行唤起支付宝钱包支付；trade_type=JSPAY时，此参数有返回
pay_url	否	仅作为参考使用，商户需自己实现该支付页面；trade_type=JSPAY时，此参数有返回
code_img_url	否	此参数的值即是根据code_url生成的可以扫码支付的二维码图片地址；trade_type=NATIVE时，此参数有返回
code_url	否	商户可用此参数自定义去生成二维码后展示出来进行扫码支付；trade_type=NATIVE时，此参数有返回
nonce_str	是	随机字符串

响应结果示例



微信预下单结果通知

应用场景

通知 URL 是初始化请求接口中提交的参数 `notify_url`，支付完成后，平台会把相关支付和用户信息发送到该 URL，商户需要接收处理信息。

对后台通知交互时，如果平台收到商户的应答不是纯字符串 **success** 或超过 3 秒后返回时，平台认为通知失败，平台会通过一定的策略（通知频率为 **0/15/15/30/180/1800/1800/1800/1800/3600**，单位：秒）间接性重新发起通知，尽可能提高通知的成功率，但不保证通知最终能成功。

由于存在重新发送后台通知的情况，因此同样的通知可能会多次发送给商户系统。商户系统必须能够正确处理重复的通知。

推荐的做法是，当收到通知进行处理时，首先检查对应业务数据的状态，判断该通知是否已经处理过，如果没有处理过再进行处理，如果处理过直接返回结果成功。在对业务数据进行状态检查和处理之前，要采用数据锁进行并发控制，以避免函数重入造成的数据混乱。

特别注意：商户后台接收到通知参数后，一定要对接收到通知参数里的订单号 `out_trade_no` 和订单金额 `total_fee` 和自身业务系统的订单和金额做校验，校验一致后才更新数据库订单状态，防止数据泄漏导致出现“假通知”，造成资金损失。

后台通知通过请求中的 `notify_url` 进行，POST 方式给商户系统（通知参数内容为 HASH 的字符串）

接口详情

接口地址	该链接是通过 微信预下单 接口中提交的参数 <code>notify_url</code> 设置，如果链接无法访问，商户将无法接收到通知。
提交方式	POST
校验签名	是

通知参数

字段名	必填	说明
<code>state</code>	是	通讯状态，详见参数规定
<code>result_code</code>	是	状态码，SUCCESS-成功，其他-失败
<code>sign</code>	是	通知的签名串

以下字段在state和trade_state都为0的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号
device_info	否	终端设备号
pay_result	是	支付结果：0—成功；其它—失败
out_trade_no	是	商户系统内部的订单号，32个字符内、可包含字母
out_transaction_id	是	第三方订单号
transaction_id	是	平台交易号
fee_type	否	货币类型，符合 ISO 4217 标准的三位字母代码，默认人民币：CNY
total_fee	是	总金额，以分为单位，只能为整数
coupon_fee	否	代金券金额，代金券金额<=订单金额，订单金额 - 代金券金额 = 现金支付金额
time_end	是	支付完成时间，格式为yyyyMMddHHmmss，如2009年12月25日9点10分10秒表示为20091225091010。时区为GMT+8 Beijing
bank_type	否	付款银行
trade_type	是	交易类型：pay.alipay.native原生扫码支付，pay.alipay.jsipay服务窗支付
sub_appid	否	调用接口提交的子商户公众账号ID
sub_openid	否	子商户appid下用户唯一标识，用户关注时存在
nonce_str	是	随机字符串
promotion_detail	否	营销详情，返回值为Json格式
wx_mch_id	否	微信服务商商户号
wx_sub_mch_id	否	微信子商户号

请求参数示例

```
bank_type=CFT&mch_id=f20170519135901860000&nonce_str=1523929649387&out_trade_no=1523929500000&out_transaction_id=2018041721001004870537670226&pay_result=0&result_code=SUCCESS&sign=00000000000000000000000000000000&state=SUCCESS&time_end=20180417094729&total_fee=1&trade_type=pay.weixin.native&transaction_id=199520439266201804177199552446
```

后台通知结果反馈

返回结果	结果说明
success	处理成功，平台收到此结果后不再进行后续通知
fail或其它字符	处理不成功，平台收到此结果或者没有收到任何结果，系统通过补单机制再次通知

支付宝预下单结果通知

应用场景

通知 URL 是初始化请求接口中提交的参数 `notify_url`，支付完成后，平台会把相关支付和用户信息发送到该 URL，商户需要接收处理信息。

对后台通知交互时，如果平台收到商户的应答不是纯字符串 `success` 或超过 3 秒后返回时，平台认为通知失败，平台会通过一定的策略（通知频率为 `0/15/15/30/180/1800/1800/1800/1800/3600`，单位：秒）间接性重新发起通知，尽可能提高通知的成功率，但不保证通知最终能成功。

由于存在重新发送后台通知的情况，因此同样的通知可能会多次发送给商户系统。商户系统必须能够正确处理重复的通知。

推荐的做法是，当收到通知进行处理时，首先检查对应业务数据的状态，判断该通知是否已经处理过，如果没有处理过再进行处理，如果处理过直接返回结果成功。在对业务数据进行状态检查和处理之前，要采用数据锁进行并发控制，以避免函数重入造成的数据混乱。

特别注意：商户后台接收到通知参数后，一定要对接收到通知参数里的订单号 `out_trade_no` 和订单金额 `total_fee` 和自身业务系统的订单和金额做校验，校验一致后才更新数据库订单状态，防止数据泄漏导致出现“假通知”，造成资金损失。

后台通知通过请求中的 `notify_url` 进行，POST 方式给商户系统（通知参数内容为 HASH 的字符串）

接口详情

接口地址	该链接是通过 支付宝预下单 接口中提交的参数 <code>notify_url</code> 设置，如果链接无法访问，商户将无法接收到通知。
提交方式	POST
校验签名	是

通知参数

字段名	必填	说明
<code>state</code>	是	通讯状态，详见参数规定
<code>result_code</code>	是	状态码，SUCCESS-成功，其他-失败
<code>sign</code>	是	通知的签名串

以下字段在 `state` 和 `trade_state` 都为 0 的时候有返回

字段名	必填	说明
<code>mch_id</code>	是	超赢商户号
<code>device_info</code>	否	终端设备号
<code>pay_result</code>	是	支付结果：0—成功；其它—失败
<code>out_trade_no</code>	是	商户系统内部的订单号，32 个字符内、可包含字母
<code>out_transaction_id</code>	是	第三方订单号
<code>transaction_id</code>	是	平台交易号
<code>fee_type</code>	否	货币类型，符合 ISO 4217 标准的三位字母代码，默认人民币：CNY
<code>total_fee</code>	是	总金额，以分为单位，只能为整数

invoice_amount	否	用户在交易中支付的可开发票的金额
time_end	是	支付完成时间，格式为yyyyMMddHHmmss，如2009年12月25日9点10分10秒表示为20091225091010。时区为GMT+8 Beijing
bank_type	否	付款银行
trade_type	是	交易类型：pay.alipay.native原生扫码支付，pay.alipay.jspay服务窗支付
buyer_user_id	是	买家支付宝用户ID
buyer_logon_id	否	买家支付宝账号
fund_bill_list	否	交易支付使用的资金渠道
nonce_str	是	随机字符串

请求参数示例

```
bank_type=ALIPAYACCOUNT&buyer_logon_id=13800000000%40qq.com&buyer_user_id=2088822494428000&device_info=device_info_yzq_01&fee_type=CNY&fund_bill_list=%5b%7b%22amount%22%3a%220.01%22%2c%22fundChannel%22%3a%22ALIPAYACCOUNT%22%7d%5d&gmt_create=20180417094613&invoice_amount=1&mch_id=f20170519135901860000&nonce_str=1523929649387&out_trade_no=1523929500000&out_transaction_id=2018041721001004870537670226&pay_result=0&result_code=SUCCESS&sign=00000000000000000000000000000000&state=SUCCESS&time_end=20180417094729&total_fee=1&trade_type=pay.alipay.native&transaction_id=199520439266201804177199552446
```

后台通知结果反馈

返回结果	结果说明
success	处理成功，平台收到此结果后不再进行后续通知
fail或其它字符	处理不成功，平台收到此结果或者没有收到任何结果，系统通过补单机制再次通知

© 超赢科技 all right reserved, powered by Gitbook文件修订时间 2019-06-04 15:08:27

- [注册超赢虚拟商户](#)
- [下发获取超赢商户号验证短信](#)
- [获取超赢商户号](#)
- [获取超赢商户微信子商户号](#)
- [获取超赢商户基础信息](#)

运维管理

注册超赢虚拟商户

应用场景

该接口提供第三方接入使用我方聚合支付业务时注册成超赢虚拟商户功能。

接口详情

接口地址	https://{BaseURL}/UnifiedPay/Mch_Reg
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号
pid	是	yunpos	调用方产品名称
sign	是	00000000000000000000000000000000	请求参数的签名串

请求参数

参数	必填	示例值	说明
-	是	-	具体参数，联调时请联系技术支持获取

请求参数示例

```
agent_id=130000000000000000&version=1.0&pid=yunpos&key1=aaa&key2=bbb&sign=00000000000000000000000000000000
```

响应结果

字段名	必填	说明
state	是	通讯状态，详见参数规定
code	是	状态码，详见参数规定
msg	否	返回信息
sign	是	响应结果的签名串

以下字段在state为SUCCESS，code为10000的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号

响应结果示例

```
{
  "state": "SUCCESS",
  "code": "10000",
  "msg": "SUCCESS",
  "mch_id": "00000001",
  "nonce_str": "f1625e94362d4d82aafcc5fc9d1f9325",
  "sign": "00000000000000000000000000000000"
}
```

下发获取超赢商户号验证短信

应用场景

该接口提供使用手机号获取短信验证码功能。

接口详情

接口地址	https://{BaseURL}/Mch/GetMchSMSValid
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号
pid	是	yunpos	调用方产品名称
sign	是	00000000000000000000000000000000	请求参数的签名串

请求参数

参数	必填	示例值	说明
mobile	是	13800000000	手机号

请求参数示例

```
agent_id=130000000000000000&version=1.0&pid=yunpos&mobile=13800000000&sign=00000000000000000000000000000000
```

响应结果

字段名	必填	说明
-----	----	----

state	是	通讯状态，详见参数规定
code	是	验证码有效时长 单位：分钟
msg	否	返回信息
sign	是	响应结果的签名串

响应结果示例

```

{
  "state": "SUCCESS",
  "code": "20",
  "msg": "短信验证码已发送",
  "sign": "00000000000000000000000000000000"
}

```

获取超赢商户号

应用场景

该接口提供获取超赢商户号功能。

接口详情

接口地址	https://{BaseURL}/Mch/GetMchId
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号
pid	是	yunpos	调用方产品名称
sign	是	00000000000000000000000000000000	请求参数的签名串

请求参数

参数	必填	示例值	说明
mobile	否	13800000000	手机号。（手机号+验证码）或（用户名+用户密码）不能同时为空
valid	否	1234	验证码。（手机号+验证码）或（用户名+用户密码）不能同时为空
username	否	13800000000	用户名。（手机号+验证码）或（用户名+用户密码）不能同时为空
password	否	12345678	用户密码。（手机号+验证码）或（用户名+用户密码）不能同时为空

请求参数示例

```
agent_id=13000000000000000&version=1.0&pid=yunpos&mobile=13800000000&valid=1234&sign=00000000000000000000000000000000
```

响应结果

字段名	必填	说明
state	是	通讯状态，详见参数规定
code	是	状态码，详见参数规定
msg	否	返回信息
sign	是	响应结果的签名串

以下字段在state为SUCCESS，code为10000的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号
mch_trade_code	是	商户交易码，大小写敏感
mch_full_name	是	商户全称
mch_short_name	是	商户简称
mch_state	是	商户状态。0-待审核、20-启用、30-停用

响应结果示例

```
{
  "state": "SUCCESS",
  "code": "10000",
  "msg": "SUCCESS",
  "mch_id": "f0000000000000001",
  "mch_trade_code": "f6742168cc",
  "mch_full_name": "广州市超赢信息科技有限公司",
  "mch_short_name": "超赢科技",
  "mch_state": "20",
  "sign": "00000000000000000000000000000000"
}
```

获取超赢商户微信子商户号

应用场景

该接口提供获取超赢商户微信子商户号功能。

接口详情

接口地址	https://{BaseURL}/Open/GetPartnerId
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号
pid	是	yunpos	调用方产品名称
sign	是	00000000000000000000000000000000	请求参数的签名串

请求参数

参数	必填	示例值	说明
mch_id	是	00000001	超赢商户号

请求参数示例

```
agent_id=130000000000000000&version=1.0&pid=yunpos&mch_id=00000001&sign=00000000000000000000000000000000
```

响应结果

字段名	必填	说明
state	是	通讯状态，详见参数规定
code	是	状态码，详见参数规定
msg	否	返回信息
sign	是	响应结果的签名串

以下字段在state为SUCCESS，code为10000的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号
partner_list	是	超赢商户微信子商户信息列表

以下字段在partner_list中返回

字段名	必填	说明
pay_channel	是	所属支付通道编码
appid	否	公众号AppId
partner_id	否	微信子商户号，在微信方未报备通过则为空

响应结果示例

```
{
  "state": "SUCCESS",
  "code": "10000",
  "msg": "SUCCESS",
  "mch_id": "00000001",
  "partner_list": "[{"pay_channel": "CEB", "partner_id": "257012345", "appid": "wx2a8d87e234100000"}, {"pay_channel": "CCB", "partner_id": "257012346", "appid": "wx2a8d87e234100000"}]",
  "sign": "00000000000000000000000000000000"
```

}

获取超赢商户基础信息

应用场景

该接口提供获取超赢商户基础信息功能。

接口详情

接口地址	https://{BaseURL}/Open/GetMchInfo
提交方式	POST
校验签名	是

公共请求参数

参数	必填	示例值	说明
agent_id	是	130000000000000000	代理商编号
version	是	1.0	调用方版本号
pid	是	yunpos	调用方产品名称
sign	是	00000000000000000000000000000000	请求参数的签名串

请求参数

参数	必填	示例值	说明
mch_id	是	00000001	超赢商户号

请求参数示例

```
agent_id=130000000000000000&version=1.0&pid=yunpos&mch_id=00000001&sign=00000000000000000000000000000000
```

响应结果

字段名	必填	说明
state	是	通讯状态，详见参数规定
code	是	状态码，详见参数规定
msg	否	返回信息
sign	是	响应结果的签名串

以下字段在state为SUCCESS，code为10000的时候有返回

字段名	必填	说明
mch_id	是	超赢商户号
FullName	是	商户全称

ShortName	是	商户简称
-----------	---	------

响应结果示例

```
{
  "state": "SUCCESS",
  "code": "10000",
  "msg": "SUCCESS",
  "mch_id": "00000001",
  "FullName": "测试商户全称",
  "ShortName": "测试商户简称",
  "sign": "00000000000000000000000000000000"
}
```

© 超赢科技 all right reserved, powered by Gitbook文件修订时间 2019-08-05 14:20:29