嬴鱼支付接口协议

Version 1.5

* 1. 接口规范说明

**本平台接口规范传输为https协议,数据传输的内容为JSON字符串流，报文字符集编码采用UTF-8编码传输。**

**注意，连接https服务器时，要配置成“忽略服务器的证书”的校验，否则会出现类似SSL认证失败或者授权失败等问题，JAVA跟PHP版本的DEMO中已经有相关的处理。**

## 概述

前段-》请求-》平台-》数据中心

## 报文接口定义

### 通用报文格式定义

JSON字符串类型通用报文格式主要包含有以下几块：

### 数据类型定义

为了方便维护和管理，报文内容中所有的信息域均采用标准可见ASCII码报文结构。

默认情况下，数据类型描述遵守以下规定：

|  |  |
| --- | --- |
| 类型 | 说明 |
| STR | 字符串类型，包含字母、数字、特殊等可见字符。 |
| INT | 整数值类型 |
| DOUBLE | 浮点值类型 |
| ARRAY | JSON结构列表类型 |
| OBJECT | JSON结构类型 |

JSON结构说明：JSON结构是一种数据结构，该结构的特点是以HASH MAP的键值关系存储数据，JSON结构内的键值数据类型可以是普通的数据类型（如：STR、int、double），还可以是复杂的数据类型（如：array、object）。

JSON结构列表说明：JSON结构列表类型是指由一组JSON类型组成的链表结构。

JSON数据类型以字符串的表式方式如下：

{

“STR key”:”STR value”,

“int key”: 123456,

“double key”: 123.88，

“object key”: {“int key”:123,”double key”:123.88},

“array key”:

[

{ “int key”:123 },

{ “int key”:123 },

…

]

}

所有接入终端报文传输内容格式均采用JSON STR方式传输，为确保数据的通用性，JSON结构中的数据均以基本字符串str类型、JSON数值类型为主。

* 1. 接口规范（json）

## 下单支付(扫码支付) (接入方🡪服务方系统)

接入方后台发起的二维码扫码支付接口，成功后返回一个支付URL给接入方，由接入方产生对应的二维码给客户支付。支付完后，服务方系统会通过支付状态通知的方式返回到接入方系统。考虑到网络异常等因素，对于没收到通知的订单交易，接入方需要有自己的机制通过支付状态查询去核实最终的支付结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段类型 | 字段描述 | 请求参数 | 应答参数 |
| 请求参数 | | | | |
| messageid | STR | 200001 | M |  |
| out\_trade\_no | STR | 商户接入方系统内部订单号，要求32个字符内，建议是数字、大小写字母组合，且在同一个商户号下唯一。 | M |  |
| branch\_id | STR | 服务方平台分配的商户号 | M |  |
| pay\_type | STR | 支付渠道编号  10:微信  20:支付宝  40:京东钱包  50:QQ钱包  70:银联二维码（银联钱包） | M |  |
| total\_fee | INT | 订单总金额(单位：分) | M |  |
| prod\_name | STR | 订单标题，这个可能会显示在客户支付的页面，但不同的支付方式显示会有差异 | M |  |
| prod\_desc | STR | 产品描述 | M |  |
| back\_notify\_url | STR | 后台通知url，必须为直接可访问的url，不能携带参数。示例：“https://pay.weixin.qq.com/wxpay/pay.action”.  支持http跟https | M |  |
| nonce\_str | STR | 随机字符串(随机字符串，不长于32位) | M |  |
| attach\_content | STR | 非必填，该字段在由商户接入方提交，主要用于商户携带订单的自定义数据，再支付状态通知或者支付状态查询时原样返回。如不填则后续也不返回此字段 | C |  |
| sign | STR | 签名,详见签名算法 | M |  |
| 返回参数(公用参数) | | | | |
| resultCode |  | 返回系统编码(00表示成功,其余表示失败) |  | M |
| resultDesc | STR | 返回系统描述 |  | M |
| resCode | STR | 返回业务编码(00表示成功,其余表示失败) |  | M |
| resDesc | STR | 返回业务描述 |  | M |
| nonceStr | STR | 随机字符串(随机字符串，不长于32位) |  | M |
| sign | STR | 报文签名 |  | M |
| 返回业务参数(成功时返回) | | | | |
| payUrl | STR | 支付二维码连接 |  | M |

示例报文：

{"messageid":"200001","out\_trade\_no":"100000220","back\_notify\_url":"www.baidu.com","branch\_id":"000001","prod\_name":"测试支付","prod\_desc":"测试支付描述","pay\_type":"20","total\_fee":1,"sign":"sdfsdfsdf"}

## 下单支付(网关支付) (接入方🡪服务方系统)

银行网关支付的接口，支付场景是由客户在接入方系统下单后，选择对应的银行，选择是借记卡还是贷记卡，然后确认跳转到银行的网关。在支付成功后，服务方系统会通过支付状态通知的方式返回到接入方系统。考虑到网络异常等因素，对于没收到通知的订单交易，接入方需要有自己的机制通过支付状态查询去核实最终的支付结果。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段类型 | 字段描述 | 请求参数 | 应答参数 |
| 请求参数 | | | | |
| messageid | STR | 200002 | M |  |
| out\_trade\_no | STR | 商户接入方系统内部订单号，要求32个字符内，建议是数字、大小写字母组合，且在同一个商户号下唯一。 | M |  |
| branch\_id | STR | 服务方平台分配的商户号 | M |  |
| pay\_type | STR | 支付渠道编号30:银联网关 | M |  |
| total\_fee | INT | 订单总金额(单位：分) | M |  |
| prod\_name | STR | 订单标题，这个可能会显示在客户支付的页面，但不同的支付方式显示会有差异 | M |  |
| prod\_desc | STR | 产品描述 | M |  |
| back\_notify\_url | STR | 后台通知url，必须为直接可访问的url，不能携带参数。示例：“https://pay.weixin.qq.com/wxpay/pay.action”.  支持http跟https | M |  |
| front\_notify\_url | STR | 前端通知url | M |  |
| bank\_code | STR | 银行简码 如 CCBD:建行，具体见下方银行编码列表 | M |  |
| bank\_flag | STR | 0：借记卡 1：贷记卡。客户在支付之前必须选择是借记卡还是贷记卡 | M |  |
| nonce\_str | STR | 随机字符串(随机字符串，不长于32位) | M |  |
| sign | STR | 签名,详见签名算法 | M |  |
| 返回跳转的页面代码(html代码) | | | | |
|  |  |  |  |  |
|  | | | | |
|  |  |  |  |  |

示例报文：

{"messageid":"200002","out\_trade\_no":"10000014","back\_notify\_url":"","front\_notify\_url":"","branch\_id":"000002","pay\_type":"30","prod\_desc":"测试订单","prod\_name":"测试订单名称","total\_fee":1,"bank\_code":"CCBD","bank\_flag":"0","sign":"sdfsdfsdf"}

银行编码表见下图:

|  |  |  |
| --- | --- | --- |
| **序号** | **银行简码** | **银行名称** |
| 1 | ICBCD | 工商银行 |
| 2 | ABCD | 农业银行 |
| 3 | BOC | 中国银行（大额） |
| 4 | BOCSH | 中国银行 |
| 5 | CCBD | 建设银行 |
| 6 | CMB | 招商银行 |
| 7 | SPDB | 浦发银行 |
| 8 | GDB | 广发银行 |
| 9 | BOCOM | 交通银行 |
| 10 | CNCB | 中信银行 |
| 11 | CMBCD | 民生银行 |
| 12 | CIB | 兴业银行 |
| 13 | CEBD | 光大银行 |
| 14 | HXB | 华夏银行 |
| 15 | BOS | 上海银行 |
| 16 | SRCB | 上海农商 |
| 17 | PSBCD | 邮政储蓄 |
| 18 | BCCB | 北京银行 |
| 19 | BRCB | 北京农商 |
| 20 | PAB | 平安银行 |

## 下单支付(手机H5支付) (接入方🡪服务方系统)

手机H5支付（又叫WAP支付），在手机浏览器打开接入方的下单页面，点击支付时直接跳转到支付页面，而不需要通过扫码。接入方从后台请求相关数据，跳转到返回的payUrl里去

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段类型 | 字段描述 | 请求参数 | 应答参数 |
| 请求参数 | | | | |
| messageid | STR | 200004 | M |  |
| out\_trade\_no | STR | 商户接入方系统内部订单号，要求32个字符内，建议是数字、大小写字母组合，且在同一个商户号下唯一。 | M |  |
| branch\_id | STR | 服务方平台分配的商户号 | M |  |
| pay\_type | STR | 支付渠道编号61:微信 62:支付宝 63:QQ钱包 64:京东WAP 65:银联快捷(wap) | M |  |
| total\_fee | INT | 订单总金额(单位：分) | M |  |
| prod\_name | STR | 订单标题，这个可能会显示在客户支付的页面，但不同的支付方式显示会有差异 | M |  |
| prod\_desc | STR | 产品描述 | M |  |
| back\_notify\_url | STR | 后台通知url，必须为直接可访问的url，不能携带参数。示例：“https://pay.weixin.qq.com/wxpay/pay.action”.  支持http跟https | M |  |
| front\_notify\_url | STR | 前端通知url | M |  |
| nonce\_str | STR | 随机字符串(随机字符串，不长于32位) | M |  |
| attach\_content | STR | 非必填，该字段在由商户接入方提交，主要用于商户携带订单的自定义数据，再支付状态通知或者支付状态查询时原样返回。如不填则后续也不返回此字段 | C |  |
| sign | STR | 签名,详见签名算法 | M |  |
| 应答数据 | | | | |
| resultCode |  | 返回系统编码(00表示成功,其余表示失败) |  | M |
| resultDesc | STR | 返回系统描述 |  | M |
| resCode | STR | 返回业务编码(00表示成功,其余表示失败) |  | M |
| resDesc | STR | 返回业务描述 |  | M |
| nonceStr | STR | 随机字符串(随机字符串，不长于32位) |  | M |
| sign | STR | 报文签名 |  | M |
| 返回业务参数(成功时返回) | | | | |
| payUrl | STR | 下单成功后，返回的跳转地址，接入方系统只要重定向到这个URL即可继续完成支付操作。 |  | C |

## 支付状态通知(服务方🡪商户接入方)

支付完成后，服务方系统通过下单的back\_notify\_url和front\_notify\_url，服务方系统通知给接入方的订单支付状态信息，接入方需要接受并返回应答。

对后台通知交互时（即back\_notify\_url）时，如果服务方系统到商户接入方的应答不是成功或超时，服务方系统会通过一定的策略定期重新发起通知，尽可能提高通知的成功率，但不保证通知最终能成功。

注意：back\_notify\_url同样的通知可能会多次发送给接入方系统。接入方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。

特别提醒：商户系统对于支付结果通知的内容一定要做签名验证,并校验返回的订单金额是否与商户侧的订单金额一致，防止数据泄漏导致出现“假通知”，造成资金损失。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段类型 | 字段描述 | 请求参数 | 应答参数 |
| 通知包含参数信息 | | | | |
| resultCode |  | 返回系统编码(00表示成功,其余表示失败). 此字段是通信标识，非交易标识，交易是否成功需要查看resCode来判断 | M |  |
| resultDesc | STR | 返回系统描述 | M |  |
| resCode | STR | 返回业务编码(00表示成功,其余表示失败) | M |  |
| resDesc | STR | 返回业务描述 | M |  |
| nonceStr | Str | 随机字符串，不长于32位 | M |  |
| sign | STR | 签名,详见签名算法 | M |  |
|  | | | | |
| branchId | STR | 服务方平台分配的商户号 | M |  |
| createTime | STR | 订单时间，接入方发起订单的时间戳 | M |  |
| orderAmt | INT | 订单总金额(单位:分) | M |  |
| orderNo | STR | 服务方系统订单流水号。下单支付的接口成功提交后，服务方系统的订单号会返回给接入方，后续也可作为账务核对的标准。 | M |  |
| outTradeNo | STR | 商户接入方系统内部订单号，要求32个字符内，建议是数字、大小写字母组合，且在同一个商户号下唯一 | M |  |
| productDesc | STR | 订单描述 | M |  |
| payType | STR | 支付方式 | C |  |
| attachContent | STR | 该字段在下单交易中由商户接入方提交，主要用于商户携带订单的自定义数据，此为原样返回。如请求时为空则此处不返回 | C |  |
| status | STR | 支付状态00:未支付 01:支付中 02:已支付 03:支付失败 04:取消 05:退款中 06:已退款 07:退款失败  **注意：通知不一定是订单支付成功，也可能是支付失败或者其他，接入方一定要判断这个字段值是02的时候才代表订单成功了。** | M |  |
| 商户接入方在收到此通知报文后返回以下参数(json格式),否则服务方系统会持续发送通知一段时间 | | | | |
| resCode | STR | 返回码(00:成功 其他为失败) |  | M |
| resDesc | STR | 返回描述 |  | M |

实例报文：以下为服务方发给商户接入方的通知

{"createTime":"20170503180443","status":"02","nonceStr":"L7AaafgKohiQzsrmjKYFoegsfCI4YSlp","resultDesc":"成功","outTradeNo":"81205669007812","sign":" AB963F88B5D4BAD62CC991FF92061E8E","productDesc":"alipay","orderNo":"p2017050318044500004771","branchId":"170500003817","resultCode":"00","resCode":"00","payType":"20","resDesc":"成功","orderAmt":1000}

## 支付状态查询(商户接入方🡪服务方系统)

该接口提供所有支付订单的查询，商户接入方可以通过查询订单接口主动查询订单状态，完成下一步的业务逻辑。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段类型 | 字段描述 | 请求参数 | 应答参数 |
| 请求参数 | | | | |
| messageid | STR | 200003 | M |  |
| branch\_id | STR | 服务方平台分配的商户号 | M |  |
| out\_trade\_no | STR | 商户接入方系统内部订单号，要求32个字符内，建议是数字、大小写字母组合，且在同一个商户号下唯一。 | M |  |
| nonce\_str | STR | 随机字符串(随机字符串，不长于32位) | M |  |
| sign | STR | 签名,详见签名算法 | M |  |
| 返回参数(公用参数) | | | | |
| resultCode |  | 返回系统编码(00表示成功,其余表示失败) |  | M |
| resultDesc | STR | 返回系统描述 |  | M |
| resCode | STR | 返回业务编码(00表示成功,其余表示失败) |  | M |
| resDesc | STR | 返回业务描述 |  | M |
| nonceStr | Str | 随机字符串 |  | M |
| sign | STR | 签名 |  | M |
| 查询成功的情况额外返回如下参数 | | | | |
| branchId | STR | 服务方平台分配的商户号 |  | M |
| createTime | STR | 订单时间 |  | M |
| orderAmt | INT | 订单金额(单位:分) |  | M |
| orderNo | STR | 服务方平台系统订单号，特定情况下核对账务用 |  | M |
| outTradeNo | STR | 商户接入方系统内部订单号，要求32个字符内，建议是数字、大小写字母组合，且在同一个商户号下唯一。 |  | M |
| productName | STR | 产品名称 |  | M |
| productDesc | STR | 产品描述 |  | M |
| status | STR | 支付状态00:未支付 01:支付中 02:已支付 03:支付失败 04:取消 05:退款中 06:已退款 07:退款失败 |  | M |
| resDesc | STR | 返回描述(主要为失败的原因) |  | C |

订单状态查询

{"messageid":"200003" ,"branch\_id":"000001","out\_trade\_no":"p2017040517073005176827","sign":"sdfsdfsdf"}

* 1. 签名算法

签名生成的通用步骤如下：

第一步，设所有发送或者接收到的数据为集合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。

**注意，因业务需求变更可能会对接口进行字段扩展，或者去除部分无用字段，接入方应该根据实际发送或者接受的集合去做遍历计算签名，而不能写死字段名称。**

特别注意以下重要规则：

1. ◆ 参数名ASCII码从小到大排序（字典序）；
2. ◆ 如果参数的值为空不参与签名；
3. ◆ 参数名区分大小写；
4. ◆ 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。

第二步，在stringA最后拼接上key得到stringSignTemp字符串，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。

**假设**传送的JSON参数如下：

appid： wxd930ea5d5a258f4f

mch\_id： 10000100

device\_info： 1000

body： test

nonce\_str： ibuaiVcKdpRxkhJA

第一步：对参数按照key=value的格式，并按照参数名ASCII字典序排序如下：

stringA="appid=wxd930ea5d5a258f4f&body=test&device\_info=1000&mch\_id=10000100&nonce\_str=ibuaiVcKdpRxkhJA";

**注：请求的JSON数据中有value为空的，则不参与SIGN的计算，即stringA中不出现这种空值的字段。**

第二步：拼接API密钥：

stringSignTemp=stringA+"&key=192006250b4c09247ec02edce69f6a2d"

sign=MD5(stringSignTemp).toUpperCase()="9A0A8659F005D6984697E2CA0A9CF3B7"

* 1. 接口地址

所有请求的地址均为：<https://api>.yingyupay.com:31006/yypay