# 1.0 接口规则

## 协议规则

传输方式：采用HTTP传输(生产环境建议HTTPS)  
提交方式：采用POST/GET方式提交  
字符编码：UTF-8  
签名算法：MD5

## 参数规范

交易金额：默认为人民币交易，单位为分，参数值不能带小数。

## 安全规范

### 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。  
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；  
◆ 验证调用返回或支付中心主动通知签名时，传送的sign参数不参与签名，将生成的签名与该sign值作校验。  
◆ 支付中心接口可能增加字段，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，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。  
支付中心会分配业务系统key，业务系统通过该key进行数据的加签和验签操作。

**第三步：**json格式转成字符串，放入params，采用post提交

# 1.1统一下单

## 接口描述

业务通过统一下单接口可以发起任意三方支付渠道的支付订单。业务系统不必关心该如何调用三方支付，统一下单接口会根据业务系统选择的支付渠道ID，选择对应支付渠道的支付产品，发起下单请求，然后响应给业务系统支付请求所需参数。

## 接口链接

URL地址：[http://服务器地址/api/pay/create\_order](https://pay.b.xxpay.org/api/pay/create_order)

## 请求参数

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 商户ID | mchId | 是 | long | 20001222 | 分配的商户号 |
| 应用ID | appId | 是 | String(32) | 0ae8be35ff634e2abe94f5f32f6d5c4f | 该商户创建的应用对应的ID |
| 支付产品ID | productId | 否 | int | 8000 | 支付产品ID |
| 商户订单号 | mchOrderNo | 是 | String(30) | 20160427210604000490 | 商户生成的订单号 |
| 币种 | currency | 是 | String(3) | cny | 三位货币代码,人民币:cny |
| 支付金额 | amount | 是 | int | 100 | 支付金额,单位分 |
| 客户端IP | clientIp | 否 | String(32) | 210.73.10.148 | 客户端IP地址 |
| 设备 | device | 否 | String(64) | ios10.3.1 | 客户端设备 |
| 支付结果前端跳转URL | returnUrl | 否 | String(128) | [http://地址/return.htm](http://shop.xxpay.org/return.htm) | 支付结果回调URL |
| 支付结果后台回调URL | notifyUrl | 是 | String(128) | [http://地址/notify.htm](http://shop.xxpay.org/notify.htm) | 支付结果回调URL |
| 商品主题 | subject | 是 | String(64) | 测试商品1 | 商品主题 |
| 商品描述信息 | body | 是 | String(256) | 测试商品描述 | 商品描述信息 |
| 扩展参数1 | param1 | 否 | String(64) |  | 支付中心回调时会原样返回 |
| 扩展参数2 | param2 | 否 | String(64) |  | 支付中心回调时会原样返回 |
| 附加参数 | extra | 是 | String(512) | {“openId”:”o2RvowBf7sOVJf8kJksUEMceaDqo”} | 特定渠道发起时额外参数,见下面说明 |
| 签名 | sign | 是 | String(32) | C380BEC2BFD727A4B6845133519F3AD6 | 签名值，详见签名算法 |

## 返回结果

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 返回状态码 | retCode | 是 | String(16) | SUCCESS | SUCCESS/FAIL此字段标识是否成功 |
| 返回信息 | retMsg | 否 | String(128) | 签名失败 | 返回信息，如非空，为错误原因 签名失败 参数格式校验错误 |

以下字段在retCode为SUCCESS的时候有返回

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 签名 | sign | 是 | String(32) | C380BEC2BFD727A4B6845133519F3AD6 | 签名值，详见签名算法 |
| 支付订单号 | payOrderId | 是 | String(32) | 20160427210604000490 | 支付中心生成的订单号 |
| 支付参数 | payParams | 是 | JSONObject |  | 该字段返回JSON格式数据，多指调起第三方支付所需传递参数 |

（1）当请求参数productId=8004（微信公众号支付）时，返回的JSON格式数据如下：

{

"sign": "24F5916D6B1AB64918B3BF7250693813",

"payOrderId": "P2016081612553098759124940",

"retCode": "SUCCESS",

"retMsg": "",

"payParams": {

"prepayId": "wx201608161255321fc59e085b0578219742",

"appId": "wx0ab67caf7f591834",

"timeStamp": 1471323332,

"signType": "MD5",

"package": "prepay\_id=wx201608161255321fc59e085b0578219742",

"nonceStr": "08s06nujam43aot3tbl555lavvydl178",

"paySign": "21E6BF4984030DBD2DB191F676A47DA2"

}

}

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 预下单号 | prepayId | 是 | String | wx201608161255321fc59e085b0578219742 | 微信预下单号 |
| 微信签名 | paySign | 是 | String | C380BEC2BFD727A4B6845133519F3AD6 | 签名,调取微信支付时用到的签名值 |
| 应用ID | appId | 是 | String | wx0ab67caf7f591834 | 微信开放平台审核通过的应用APPID |
| 签名算法 | signType | 是 | String | MD5 | 签名算法，暂支持MD5 |
| 扩展字段 | package | 是 | String | prepay\_id=wx201608161255321fc59e085b0578219742 | 扩展字段 |
| 随机字符串 | nonceStr | 是 | String | 5K8264ILTKCH16CQ2502SI8ZNMTM67VS | 随机字符串，不长于32位 |
| 时间戳 | timeStamp | 是 | String | 1412000000 | 时间戳,10位 |

（2）当请求参数productId=8002（微信原生扫码支付）时，返回的JSON格式数据如下：

{

"sign": "3424ED83034B20CF17961720FE272A94",

"payOrderId": "pay2016081613050939023127932",

"retCode": "SUCCESS",

"retMsg": "",

"payParams": {

"prepayId": "wx201608161255321fc59e085b0578219742",

"codeUrl": "weixin://wxpay/bizpayurl?pr=sHYHnns",

"codeImgUrl": "http://wxpay/bizpayurl?pr=sHYHnns"

}

}

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 预下单号 | prepayId | 是 | String | wx201608161255321fc59e085b0578219742 | 微信预下单号 |
| 二维码链接 | codeUrl | 是 | String | weixin://wxpay/s/Anp43md | 可将该参数值生产二维码展示 |
| 二维码图片链接 | codeImgUrl | 是 | String |  |  |

（3）当请求参数productId=8007（支付宝H5支付），返回的JSON格式数据如下：

{

"payOrderId": "P0020180114172136000000",

"sign": "A1F39B0D7BD15E7A7BB6B99762302C51",

"payParams": {

"payUrl": "<form name=\"punchout\_form\" method=\"post\" action=\"https://openapi.alipay.com/gateway.do?charset=UTF-8&method=alipay.trade.wap.pay&sign=DxPbQZwi2Zpnv+qC8UdfLgSGMB//aspU34ZkRVdi7kzUlAQsDiNoT0MIsyby2c/ahNXjOx69IrPqhha6oIDUBGFodK6nZ3izfDkagbKyvoSnRVaoQ/XKeQowiCmVMiln9xN1SpiWdXE7dsQkCzCAzUD5yeAAbQbn38MkJ2TF6dFXql+T0yATgkSzYSuQxlUJVnpKCpDKYeL0NHaf58EbT63pkdOyfODWz+D1eUwbspzzul1kY7AYD2ZSHQKyW4zxQS0YzrpUKPhF2olGtgVZo2EEcqQuxIgC4hz1TTqVjD2VK5kj45BJ+0xd1DsojLXjxaR5qziFYKqSGU8OiN0yhg==&return\_url=http://www.xxxxx.org&notify\_url=http://pay.t.xxxxx.org/notify/pay/aliPayNotifyRes.htm&version=1.0&app\_id=2015081500216362&sign\_type=RSA2&timestamp=2018-01-14+17:21:37&alipay\_sdk=alipay-sdk-java-dynamicVersionNo&format=json\">\n<input type=\"hidden\" name=\"biz\_content\" value=\"{&quot;body&quot;:&quot;支付测试&quot;,&quot;out\_trade\_no&quot;:&quot;P0020180114172136000000&quot;,&quot;product\_code&quot;:&quot;QUICK\_WAP\_PAY&quot;,&quot;subject&quot;:&quot;支付测试&quot;,&quot;total\_amount&quot;:&quot;0.01&quot;}\">\n<input type=\"submit\" value=\"立即支付\" style=\"display:none\" >\n</form>\n<script>document.forms[0].submit();</script>"

},

"retCode": "SUCCESS"

}

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 支付URL | payUrl | 是 | String |  | 跳转支付wap支付所需的表单信息 |

（4）当请求参数productId=8006（支付宝扫码支付），返回的JSON格式数据如下：

{

"msg": "下单成功",

"code": 0,

"data": {

"payOrderId": "P01000120180902125417568",

"sign": "FE1E40738752D8EDD387E9015E881CDA",

"payParams": {

"codeUrl": "https://qr.alipay.com/bax04800wim0uadp1ldi20e4",

"codeImgUrl": "http://127.0.0.1:3020/api/qrcode\_img\_get?url=https://qr.alipay.com/bax04800wim0uadp1ldi20e4&widht=200&height=200"

},

"retCode": "SUCCESS"

}

}

（5）当请求参数productId=8015/8016（app微信/支付宝扫码支付），返回的JSON格式数据如下：

{

"msg": "下单成功",

"code": 0,

"data": {

"payOrderId": "P01000220180902125659800",

"sign": "30D6AF629BA891725378E1AC4F61BF62",

"payParams": {

"codeUrl": "http://....",

"codeImgUrl": "http://127.0.0.1:3020/api/qrcode\_img\_get?url=http://....&widht=200&height=200"

},

"retCode": "SUCCESS"

}

}

# 1.2 查询支付订单

## 接口描述

业务系统通过查询支付订单接口获取最新的支付订单状态，并根据状态结果进一步处理业务逻辑。

## 接口链接

URL地址：[http://地址/api/pay/query\_order](https://pay.b.xxpay.org/api/pay/query_order)

## 请求参数

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 商户ID | mchId | 是 | String(30) | 1000000010 | 支付中心分配的商户号 |
| 应用ID | appId | 是 | String(32) | 0ae8be35ff634e2abe94f5f32f6d5c4f | 该商户创建的应用对应的ID |
| 支付订单号 | payOrderId | 是 | String(30) | P20160427210604000490 | 支付中心生成的订单号，与mchOrderNo二者传一即可 |
| 商户订单号 | mchOrderNo | 是 | String(30) | 20160427210604000490 | 商户生成的订单号，与payOrderId二者传一即可 |
| 是否执行回调 | executeNotify | 否 | Boolean | true | 是否执行回调，如果为true，则支付中心会再次向商户发起一次回调，如果为false则不会发起 |
| 签名 | sign | 是 | String(32) | C380BEC2BFD727A4B6845133519F3AD6 | 签名值，详见签名算法 |

## 返回结果

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 返回状态码 | retCode | 是 | String(16) | SUCCESS | SUCCESS/FAIL此字段标识是否成功 |
| 返回信息 | retMsg | 否 | String(128) | 签名失败 | 返回信息，如非空，为错误原因 签名失败 参数格式校验错误 |

以下字段在retCode为SUCCESS的时候有返回

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 商户ID | mchId | 是 | long(30) | 20001222 | 支付中心分配的商户号 |
| 应用ID | appId | 是 | String(32) | 0ae8be35ff634e2abe94f5f32f6d5c4f | 该商户创建的应用对应的ID |
| 支付产品ID | productId | 是 | int | 8001 | 支付产品ID |
| 支付订单号 | payOrderId | 是 | String(30) | P20160427210604000490 | 支付中心生成的订单号 |
| 商户订单号 | mchOrderNo | 是 | String(30) | 20160427210604000490 | 商户生成的订单号 |
| 支付金额 | amount | 是 | int | 100 | 支付金额,单位分 |
| 币种 | currency | 是 | String(3) | cny | 三位货币代码,人民币:cny |
| 状态 | status | 是 | int | 1 | 支付状态,0-订单生成,1-支付中,2-支付成功,3-业务处理完成 |
| 渠道用户ID | channelUser | 否 | String(64) | [jmdhappy@126.com](mailto:jmdhappy@126.com) | 渠道测支付时使用的用户ID |
| 渠道订单号 | channelOrderNo | 否 | String | wx20170910211043fb206e92260071822007 | 对应的第三方支付订单号 |
| 渠道数据包 | channelAttach | 否 | String | {“bank\_type”:”CMB\_DEBIT”,”trade\_type”:”pay.weixin.micropay”} | 支付渠道数据包 |
| 支付成功时间 | paySuccTime | 否 | Long | 1505049094262 | 支付成功时间 |

# 1.3 支付结果通知

## 接口描述

## 接口链接

该链接是通过统一下单接口提交的参数notifyUrl设置，如果无法访问链接，业务系统将无法接收到支付中心的通知。

## 通知参数

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 支付订单号 | payOrderId | 是 | String(30) | P20160427210604000490 | 支付中心生成的订单号 |
| 商户ID | mchId | 是 | String(30) | 20001222 | 支付中心分配的商户号 |
| 应用ID | appId | 是 | String(32) | 0ae8be35ff634e2abe94f5f32f6d5c4f | 该商户创建的应用对应的ID |
| 支付产品ID | productId | 是 | int | 8001 | 支付产品ID |
| 商户订单号 | mchOrderNo | 是 | String(30) | 20160427210604000490 | 商户生成的订单号 |
| 支付金额 | amount | 是 | int | 100 | 支付金额,单位分 |
| 状态 | status | 是 | int | 1 | 支付状态,0-订单生成,1-支付中,2-支付成功,3-业务处理完成 |
| 渠道订单号 | channelOrderNo | 否 | String(64) | wx2016081611532915ae15beab0167893571 | 三方支付渠道订单号 |
| 渠道数据包 | channelAttach | 否 | String | {“bank\_type”:”CMB\_DEBIT”,”trade\_type”:”pay.weixin.micropay”} | 支付渠道数据包 |
| 扩展参数1 | param1 | 否 | String(64) |  | 支付中心回调时会原样返回 |
| 扩展参数2 | param2 | 否 | String(64) |  | 支付中心回调时会原样返回 |
| 支付成功时间 | paySuccTime | 是 | long |  | 精确到毫秒 |
| 通知类型 | backType | 是 | int | 1 | 通知类型，1-前台通知，2-后台通知 |
| 签名 | sign | 是 | String(32) | C380BEC2BFD727A4B6845133519F3AD6 | 签名值，详见签名算法 |

## 返回结果

业务系统处理后同步返回给支付中心，返回字符串 success 则表示成功，返回非success则表示处理失败，支付中心会再次通知业务系统。（通知频率为60/120/180/240/300,单位：秒）

# 1.4 创建收银台

## 接口描述

业务通过创建收银台接口得到收银台地址URl，用户打开收银台URL进入支付流程。

## 接口链接

创建PC收银台接口地址：[http://服务器地址/api/cashier/pc\_build](https://pay.b.xxpay.org/api/cashier/pc_build)

## 请求参数

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 商户ID | mchId | 是 | long | 20001222 | 分配的商户号 |
| 应用ID | appId | 否 | String(32) | 0ae8be35ff634e2abe94f5f32f6d5c4f | 该商户创建的应用对应的ID |
| 支付产品ID | productId | 是 | int | 8000 | 支付产品ID |
| 商户订单号 | mchOrderNo | 是 | String(30) | 20160427210604000490 | 商户生成的订单号 |
| 支付金额 | amount | 是 | int | 100 | 支付金额,单位分 |
| 支付结果前端跳转URL | returnUrl | 否 | String(128) | [http://shop.xxxxx.org/return.htm](http://shop.xxpay.org/return.htm) | 支付结果回调URL |
| 支付结果后台回调URL | notifyUrl | 是 | String(128) | [http://shop.xxxxx.org/notify.htm](http://shop.xxpay.org/notify.htm) | 支付结果回调URL |
| 商品主题 | subject | 是 | String(64) | 测试商品1 | 商品主题 |
| 商品描述信息 | body | 是 | String(256) | 测试商品描述 | 商品描述信息 |
| 签名 | sign | 是 | String(32) | C380BEC2BFD727A4B6845133519F3AD6 | 签名值，详见签名算法 |

## 返回结果

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 返回状态码 | retCode | 是 | String(16) | SUCCESS | SUCCESS/FAIL此字段标识是否成功 |
| 返回信息 | retMsg | 否 | String(128) | 签名失败 | 返回信息，如非空，为错误原因 签名失败 参数格式校验错误 |

以下字段在retCode为SUCCESS的时候有返回

| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| --- | --- | --- | --- | --- | --- |
| 签名 | sign | 是 | String(32) | C380BEC2BFD727A4B6845133519F3AD6 | 签名值，详见签名算法 |
| 收银台URl | payUrl | 是 | String |  | 收银台地址 |

返回的JSON格式数据如下：

{

"sign": "5AAD2B9513ADE7E0C76A4BC99E1A92F0",

"payUrl": "http://127.0.0.1:3020/api/cashier/pc?mchId=20000000&appId=710ddfebd2154434a8cfee1807b27eea&productId=8000,8001&amount=1&mchOrderNo=1528684776204&subject=xxxxxY%E6%94%AF%E4%BB%98%E6%B5%8B%E8%AF%95&body=XXXXX%E6%94%AF%E4%BB%98%E6%B5%8B%E8%AF%95&notifyUrl=http%3A%2F%2Fwww.baidu.com&sign=9849EC1C736C72BCD59558DACA69FD6B",

"retCode": "SUCCESS"

}

# 1.6 编码表

## 1. 支付产品

| **产品ID** | **产品名称** | **支付类型** |
| --- | --- | --- |
| 8016 | app支付宝扫码 | app支付宝扫码 |

## 2. 错误码

| **错误码值** | **描述** | **原因** | **解决方案** |
| --- | --- | --- | --- |
| 0010 | 系统错误 | 系统超时或异常 | 系统异常，请用相同参数重新调用 |