



a.HTTP 头参数说明:

字段名称	字段类型	说明
signature	string	签名
request-time	int	10 位时间戳
appid	string	appid 商户创建时分配, 可在后台获取

b.业务参数说明:

字段名称	字段类型	必填	说明
amount	number	是	订单金额,单位元
currency	string	是	货币类型, 需要用户支付的货币类型
trade_no	string	是	商户订单号, 要纯数字可以加下划线, 举例: 89_4587855或95828552
return_url	string	否	支付成功后, 页面跳转链接地址;(限长 250 字符)
items	string	否	商品列表, 需先转为json 对象字符串再转为base64 后再传输, <b>base64encode(Json::encode(items))</b>
consumer_id	string	否	商户客户唯一识别号
billing_data	string	否	客户卡账单信息, 需先转为json 对象字符串再转为 base64 后再传 输, <b>base64encode(Json::encode(billing_data))</b>

billing\_data 字段: 该值数据类为对象, 注意: 有些支付方式\*为必须的!!!

字段名称	字段类型	说明
email*	string	邮箱地址
first_name*	string	姓
last_name*	string	名
phone*	string	手机号
country*	string	国家
city*	string	城市
address_1*	string	地址 1
address_2(非必要)	string	地址 2
postcode*	number	邮政编码
state*	string	州/省

items 字段:

字段名称	字段类型	说明
name	string	商品名称
description	string	商品描述
imageurl	url	商品图, 需远程可显示

c. 返回数据说明:

类型: json, 把返回数据做json 解析后得到数据字段:

字段名称	字段类型	说明
code	int	为 0 时代表成功, 否则为失败, 请参考 msg 字段
msg	string	当 code 为负数时的说明
data	array	成功时返回的数据, 失败时不存在该字段



6. 收款通知回调时签名内参数的排列顺序（一定要按此顺序）：

**签名举例：**amount=269.99 currency=USD&trade\_no=157&order\_id=87\_6931d886caa2f&status=successful

## 六.支付通知

仅在支付后台设置了通知地址并且支付成功后，该功能才会向该地址发送通知。通知失败重发的时间间隔为(s 为秒): 5s, 30s, 60s, 120s, 300s,1500s。

a.HTTP 头参数说明：

字段名称	字段类型	说明
signature	string	签名
request-time	int	10 位时间戳
appid	string	appid 商户创建时分配，可在后台获取

b.请求：

通知地址：客户提供通知方式：POST

请求类型：Content-Type=application/json c.业务参数说明：

字段名称	字段类型	说明
amount	number	订单金额,单位元
currency	string	货币类型，需要用户支付的货币类型
trade_no	string	商户订单号
order_id	string	平台支付订单号
status	string	successful 表示成功，其他表示失败，失败说明参考 msg 字段
msg	string	当 status 不为successful 时的说明，成功状态时内容为空

C. 商户返回:

请在接收到系统通知后返回 success 字符串 (小写) , 并且 http 状态码为 200, 接收通知的功能里不能有其他的入口校验, 如有请关闭, 否则不能正常接收通知。

D. 以下为收到通知校验的伪代码流程

```
data = [  
    'amount' => getPost(amount), 'currency' =>  
    getPost(currency), 'trade_no' =>  
    getPost(trade_no), 'order_id' =>  
    getPost(order_id), 'status' => getPost(status),  
    'msg' => getPost(msg),  
];  
  
appid = APPID; secret = SECRET;  
  
timestamp = getHeader(request-time); signature =  
getHeader(signature)  
  
args = toUrlParams(data). '&appsecret='.secret; sign =  
makeSign(appid, timestamp, args);  
  
Return signature == sign
```

## 七.demo

demo 有一个简单的下单、支付、通知的流程, 其他语言请参考实现!