
真好付

商户支付接口规范

V1.6

技术 QQ

3487159546

2018 年 6 月 22 日

目 录

- 第一章 文档描述..... 3
 - 1 文档说明3
 - 2 阅读对象3
- 第二章 系统原理..... 3
 - 1 术语定义3
 - 2 消费流程3
 - 3 查询流程4
 - 4 安全说明4
- 第三章 接口定义..... 4
 - 1. 参数定义说明4
 - 2. 接口说明4
 - 2.1. 二维码支付接口（商户系统发起请求） 4
 - 2.2. 收银台接口（商户系统发起请求） 7
 - 2.3. 订单查询接口（商户系统发起请求） 8
 - 2.4. 异步通知接口（支付平台接口发起请求） 10
- 附录 1：产品编码说明..... 12
- 附录 2：银行编码说明..... 13
- 附录 3：MD5 签名算法..... 14

第一章 文档描述

1 文档说明

本说明文档用于指导支付平台商户实现与支付平台接口的顺利对接。
请相关技术人员详细阅读本文档。

2 阅读对象

商户开发人员。
支付平台接口相关技术人员。

第二章 系统原理

1 术语定义

商户：指已独立拥有在线销售系统，并成为支付平台接口系统会员，且已使用或者将使用支付平台接口实现收付费或其他相关功能需求的企业或者个人。

外部接入接口：指支付平台通过统一的方式接受外部请求的接口。

商户系统：指使用支付平台外部接入接口的计算机系统，一般指商户所拥有的在线销售系统。

用户：指在商户系统内进行消费的企业或者个人。

商户号：指支付平台为使用外部接入接口的商户统一分配的唯一标识。

商户密钥：指商户在和支付平台确定合作意向时支付平台为商户提供的加密字符串。该信息作为商户系统和支付平台交互时身份确认的依据，需商户妥善保管。

2 消费流程

通常，一次支付最少包括 2 个交互过程。第一个过程是商户系统将在线支付等信息提交支付平台网关说明文档到支付平台接口的过程，称之为上行过程。第二个过程是在支付完成后，支付平台接口通过 server to server 方式将支付结果告知商户系统的过程，称之为下行异步通知过程。

如果用户在跳转到支付平台接口完成支付操作后，需要支付平台接口将处理结果立即返回给商户系统的下一步页面，让用户继续完成整个操作流程，此时支付还应包括第三个流程，下行同步通知过程。该过程需要在上行过程中提交参数 `return_url`。

3 查询流程

在商户系统将订单成功提交给支付平台接口后的任意时间，商户系统可发起请求查询订单的处理结果。
该流程由商户系统发起，由接口返回相应结果。

4 安全说明

支付平台采用 MD5 加密方式对传输数据进行签名验证，具体请参考相关接口定义。对 MD5 加密后的签名值，验签不区分大小写。

第三章 接口定义

1. 参数定义说明

1.1. 数据元类型定义

String	字符串
int	整数
double	双精度小数

1.2. 参数是否必选

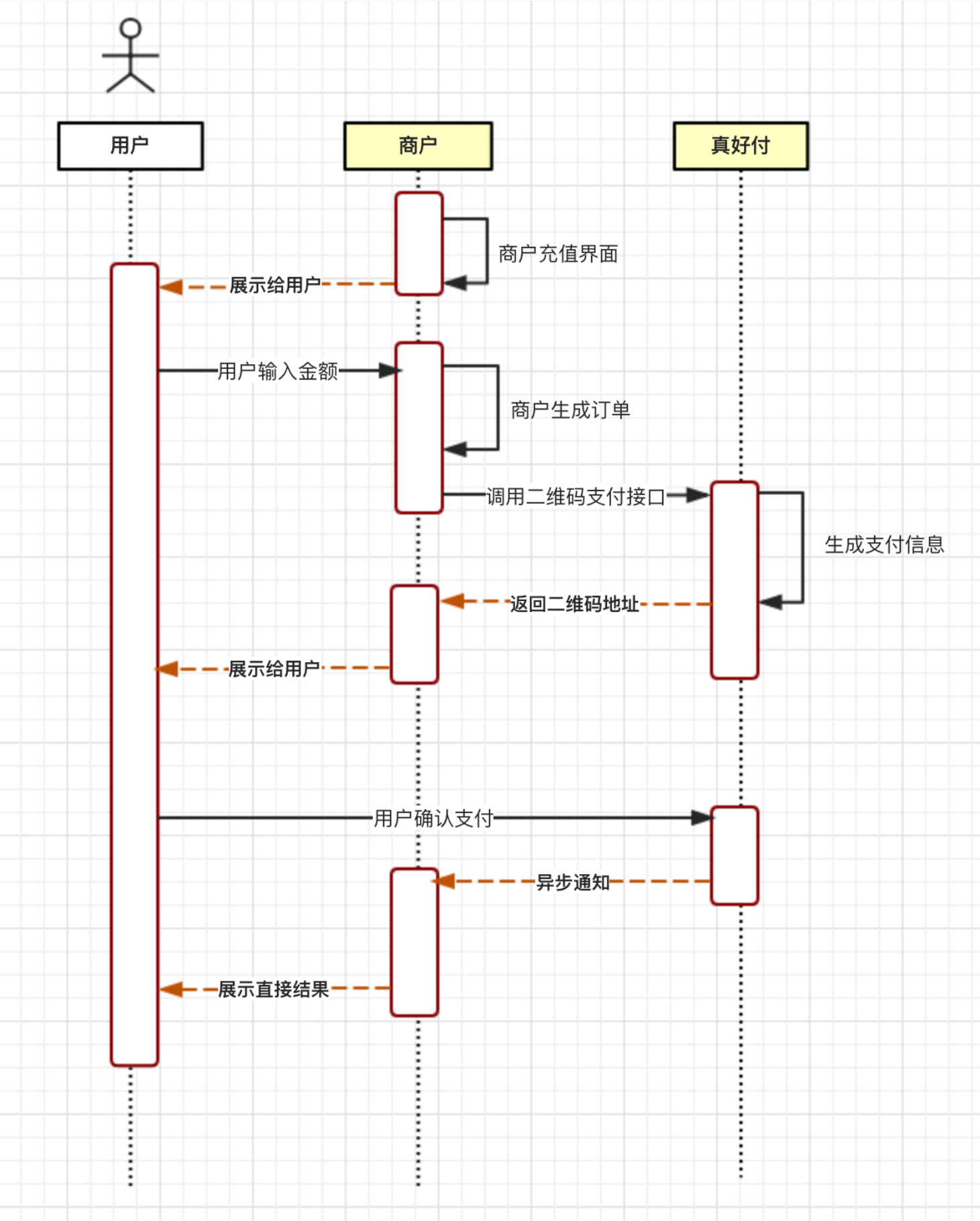
Y	必选
N	可选

2. 接口说明

2.1. 二维码支付接口（商户系统发起请求）

商户发起支付请求，调用支付系统获取微信支付二维码，商家→支付平台

2.1.1. 业务流程



2.1.2.接口调用地址

<http://pay1.527460.cn/pay>

2.1.3.接口调用方式

POST

2.1.4.参数说明

字段名	变量名	必填	类型	示例值	描述
随机字符	nonceStr	Y	String(32)	Er123sd2	随机字符串，不长于 32 位

串					
交易发起时间	startTime	Y	String(14)	20160125091010	交易时间，格式为 yyyyMMddHHmmss，如 2009 年 12 月 25 日 9 点 10 分 10 秒表示为 20091225091010
商户号	merchantNo	Y	String(32)		
商户订单号	outOrderNo	Y	String(32)	A906002474110810	每笔交易的流水号，需要保证每笔交易不重复，32 个字符内、可包含字母；异步通知、获取二维码、查询订单详情均需要使用此字段
交易金额	amount	Y	Int	10010	交易金额，单位为分
客户端 ip	client_ip	Y	String		
时间戳	timestamp	Y	Long	151231432342	接口请求时间戳
订单描述	description	N	String(128)		
异步通知地址	notifyUrl	N	String(255)		支付成功后，支付平台通知商户系统订单成功的地址。支付平台优先使用此字段，如果此字段为空，将使用商户后台绑定的异步通知地址。如果两个都为空，将不通知商户系统订单支付成功。
扩展字段	extra	N	String(255)		异步通知接口将原样返回
产品编码	productCode	Y	String(32)	1101	支付类型，具体参考《附录 1：产品编码说明》
银行编码	bankCode	N	String(32)	CCB	网银支付，必填，具体参考《附录 2：银行编码说明》
验签字段	sign	Y	String(32)		MD5 签名，具体参考《附录 3：MD5 签名算法》

2.1.5. 返回参数说明

字段名	变量名	必填	类型	示例值	描述
接口状态码	status	Y	Int	200	200：接口正常返回 其他：接口异常
状态信息	msg	Y	String(100)		交易时间，格式为 yyyyMMddHHmmss，如 2009 年 12 月 25 日 9 点 10 分 10 秒表示为 20091225091010
返回的数据	data	Y	JSON		
二维码	payUrl	Y	String(200)	http://pay2.xxx.com/qr.png	二维码地址
随机字符串	nonceStr	Y	String(32)	Er123sd2	随机字符串，不长于 32 位
时间戳	timestamp	Y	Long	151231432342	接口响应时间戳
验签字段	sign	Y	String(32)		MD5 签名，具体参考《附录 3：MD5 签名算法》

2.1.6. 返回参数示例

```
{
  "status": 200,
  "msg": "",
  "data": {
    "payUrl": "http://xxxx.png",
    "nonceStr": "12asdfas131234f",
```

```

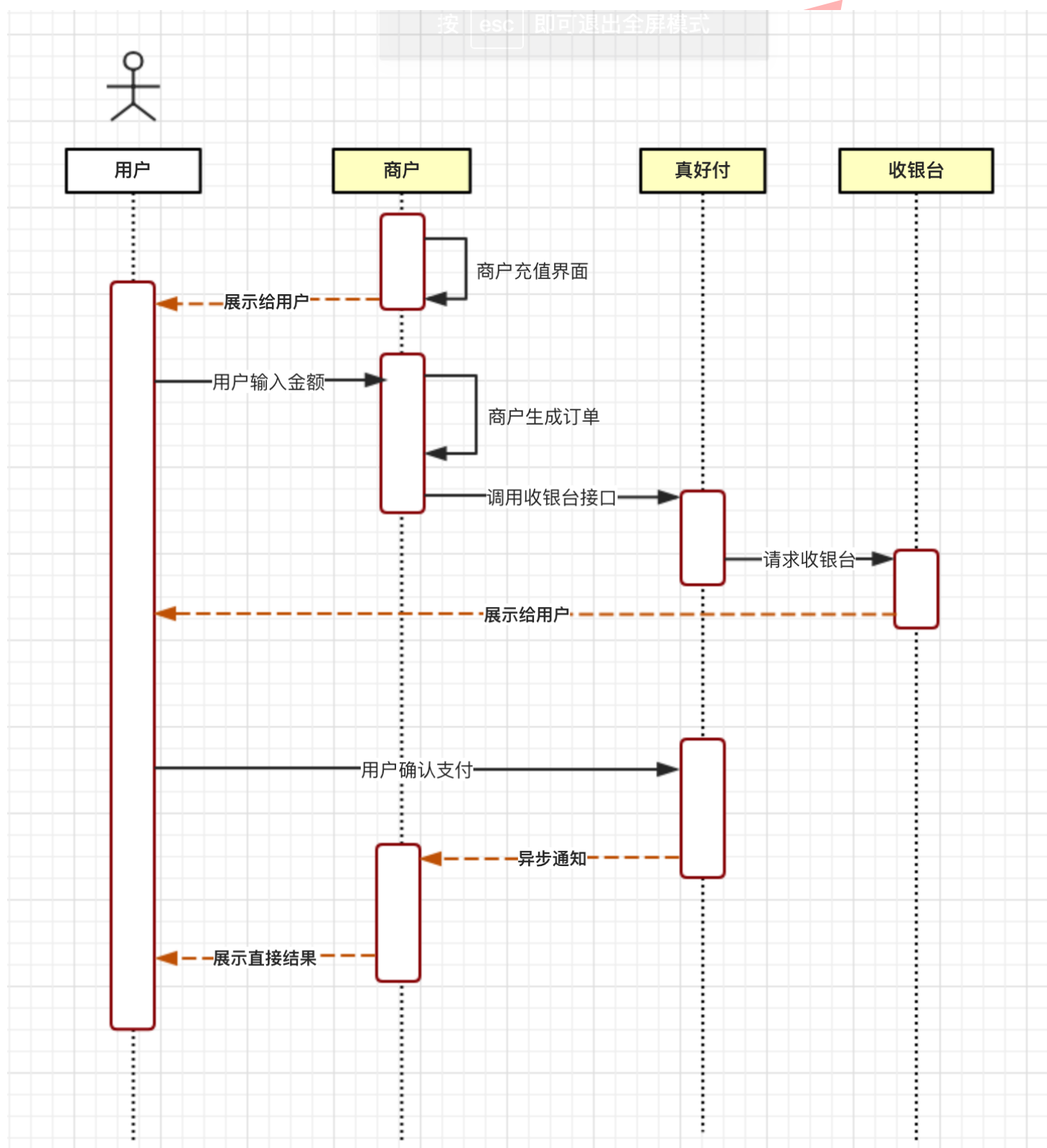
    "timestamp": 15012312312342,
    "sign": "afasdf34fasdfasd"
  }
}

```

2.2. 收银台接口（商户系统发起请求）

商户发起支付请求，调用支付系统获取微信支付二维码，商家→支付平台

2.2.1. 业务流程



2.2.2. 接口调用地址

<http://pay1.527460.cn/cashier>

2.2.3. 接口调用方式

2.2.4.参数说明

字段名	变量名	必填	类型	示例值	描述
随机字符串	nonceStr	Y	String(32)	Er123sd2	随机字符串，不长于 32 位
交易发起时间	startTime	Y	String(14)	20160125091010	交易时间，格式为 yyyyMMddHHmmss，如 2009 年 12 月 25 日 9 点 10 分 10 秒表示为 20091225091010
商户号	merchantNo	Y	String(32)		
商户订单号	outOrderNo	Y	String(32)	A906002474110810	每笔交易的流水号，需要保证每笔交易不重复，32 个字符内、可包含字母；异步通知、获取二维码、查询订单详情均需要使用此字段
交易金额	amount	Y	Int	10010	交易金额，单位为分
客户端 ip	client_ip	Y	String		
时间戳	timestamp	Y	Long	151231432342	接口请求时间戳
订单描述	description	N	String(128)		
异步通知地址	notifyUrl	N	String(255)		支付成功后，支付平台通知商户系统订单成功的地址。支付平台优先使用此字段，如果此字段为空，将使用商户后台绑定的异步通知地址。如果两个都为空，将不通知商户系统订单支付成功。
支付成功回调地址	returnUrl	N	String(255)		支付成功后，支付平台跳转到商户的支付成功页面。
扩展字段	extra	N	String(255)		异步通知接口将原样返回
产品编码	productCode	Y	String(32)	1101	支付类型，具体参考《附录 1：产品编码说明》
银行编码	bankCode	N	String(32)	CCB	网银支付，必填，具体参考《附录 2：银行编码说明》
验签字段	sign	Y	String(32)		MD5 签名，具体参考《附录 3：MD5 签名算法》

2.2.5.返回说明

跳转到收银台页面进行收款

2.3. 订单查询接口（商户系统发起请求）

该接口提供所有微信支付订单的查询，商户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。

需要调用查询接口的情况：

- ◆ 当商户后台、网络、服务器等出现异常，商户系统最终未接收到支付通知；
- ◆ 调用支付接口后，返回系统错误或未知交易状态情况；

2.3.1.接口调用地址

<http://pay1.527460.cn/query>

2.3.2.接口调用方式

2.3.3. 参数说明

字段名	变量名	必填	类型	示例值	描述
商户订单号	outOrderNo	Y	String(32)	A906002474110810	每笔交易的流水号，需要保证每笔交易不重复，32个字符内、可包含字母；异步通知、获取二维码、查询订单详情均需要使用此字段
随机字符串	nonceStr	Y	String(32)	Er123sd2	随机字符串，不长于 32 位
商户号	merchantNo	Y	String(32)	15012312312342	
时间戳	timestamp	Y	Long	151231432342	接口请求时间戳
验签字段	sign	Y	String(32)		MD5 签名，具体参考《附录 3: MD5 签名算法》

2.3.4. 返回格式示例

```
{
  "status": 200,
  "msg": "",
  "data": {
    "orderStatus": 1,
    "outOrderNo": "12asdfas131234F",
    "merchantNo": 15012312312342,
    "payTime": "20180326220909",
    "nonceStr": "afasdf34fasdfasd",
    "timestamp": "afasdf34fasdfasd",
    "sign": "afasdf34fasdfasd"
  }
}
```

2.3.5. 参数说明

字段名	变量名	必填	类型	示例值	描述
接口状态码	status	Y	Int	200	200: 接口正常返回 其他: 接口异常
状态信息	msg	Y	String(100)		交易时间，格式为 yyyyMMddHHmmss，如 2009 年 12 月 25 日 9 点 10 分 10 秒表示为 20091225091010
返回的数据	data	Y	JSON		
订单状态	orderStatus	Y	Int	1	0 未支付 1 已支付: 未通知商户或者已经通知商户，但是商户没有返回成功 2 关闭: 订单超时未支付 3 完成: 已经通知商户，并且商户返回成功
商户订单号	outOrderNo	Y	String(32)	A906002474110810	每笔交易的流水号，需要保证每笔交易不重复，32 个字符内、可包含字母；异步通知、获取二维码、查询订单详情均需要使用此字段

商户号	merchantNo	Y	String(32)	15012312312342	
支付时间	payTime	Y	String(14)	20160125091010	支付时间，格式为 yyyyMMddHHmmss，如 2009 年 12 月 25 日 9 点 10 分 10 秒表示为 20091225091010
随机字符串	nonceStr	Y	String(32)	Er123sd2	随机字符串，不长于 32 位
时间戳	timestamp	Y	Long	151231432342	接口响应时间戳
验签字段	sign	Y	String(32)		MD5 签名，具体参考《附录 3：MD5 签名算法》

2.4. 异步通知接口（支付平台接口发起请求）

支付完成后，支付平台会把相关支付结果和用户信息发送给商户，商户需要接收处理，并返回应答。
对后台通知交互时，如果支付平台收到商户的应答不是成功或超时，支付平台认为通知失败，支付平台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，但支付平台不保证通知最终能成功。（通知频率为 5/10/30/60/300/1800/1800/3600/3600/7200，单位：秒）

注意：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。

2.4.1.接口调用方式

POST

2.4.2.参数说明

字段名	变量名	必填	类型	示例值	描述
支付系统订单号	orderNo	Y	String(32)		
通知时间	notifyTime	Y	String(14)	20160125091010	支付时间，格式为 yyyyMMddHHmmss，如 2009 年 12 月 25 日 9 点 10 分 10 秒表示为 20091225091010
支付时间	payTime	Y	String(14)	20160125091010	支付时间，格式为 yyyyMMddHHmmss，如 2009 年 12 月 25 日 9 点 10 分 10 秒表示为 20091225091010
订单金额	amount	Y	Int	10010	交易金额，单位为 分
订单状态	orderStatus	Y	Int	1	0 未支付 1 已支付：未通知商户或者已经通知商户，但是商户没有返回成功 2 关闭：订单超时未支付 3 完成：已经通知商户，并且商户返回成功
支付系统下单时间	createTime	Y	String(14)	20160125091010	支付时间，格式为 yyyyMMddHHmmss，如 2009 年

					12月25日9点10分10秒表示为20091225091010
商户订单号	outOrderNo	Y	String(16)		每笔交易的流水号，需要保证每笔交易不重复，32个字符内、可包含字母；异步通知、获取二维码、查询订单详情均需要使用此字段，原样返回
交易发起时间	startTime	Y	String(14)	20160125091010	支付时间，格式为yyyyMMddHHmmss，如2009年12月25日9点10分10秒表示为20091225091010
随机字符串	nonceStr	Y	String(32)		随机字符串，不长于32位
订单描述	description	N	String(255)		
客户端ip	clientIp	Y	String(255)		
通知类型	notifyType	Y	String		AUTO，自动 MANUAL 人工补单
扩展字段	extra	N	String		原样返回
验签字段	sign	Y	String(32)		MD5签名，具体参考《附录3：MD5签名算法》

2.4.3.返回说明

- （1）异步通知过程在整个支付流程中一定存在。商户系统在收到下行通知过程后，需向支付平台接口返回“SUCCESS”。支付平台接口根据该返回值判断商户系统是否已经收到结果。若返回结果不是“SUCCESS”，则认为是通知失败。
- （2）当支付平台接口在下行异步通知过程中返回在线支付结果时，商户系统在收到数据后，应该通过sign值判断是否是有效的返回数据，防止数据在网络传输过程中被恶意篡改。

2.4.4.返回示例

SUCCESS

2.4.5.注意事项

- ① 在收到支付平台接口订单下行异步通知结果时，商户系统需首先验证订单通知的合法性，如果不合法，则不要更新商户系统上的订单状态。
- ② 商户系统内订单状态和订单实际金额务必以此次支付平台接口订单异步通知的结果为准。
- ③ 商户系统在成功提交订单后，在未没有收到支付平台接口异步通知结果时，请不要更新商户系统上的订单状态
- ④ 在下行异步通知过程中商户系统返回结果值“SUCCESS”回支付平台接口时，“SUCCESS”并不是将支付平台接口在下行异步通知的结果原样返回。商户系统返回值“SUCCESS”时表示商户系统已经成功接收到了结果（不论结果是什么，总之是收到了），而其他值表示因为某些原因商户系统并不认为支付平台接口的返回是有效的。

附录 1：产品编码说明

产品编码	产品名称
1101	微信扫码支付
1102	微信 H5 支付
1103	微信 APP 支付
1201	支付宝扫码支付
1202	支付宝 H5 支付
1203	支付宝 APP 支付
1301	QQ 钱包 H5 支付
1302	QQ 钱包扫码支付
1401	京东 H5 支付
1402	京东扫码支付
1501	快捷支付
1601	网银支付
1701	云闪付
1801	美团扫码支付
1802	美团 H5 支付
1901	百度钱包扫码支付
1902	百度钱 H5 支付
3001	代付

附录 2： 银行编码说明

银行名称	银行简码
中国农业银行	ABC
中国银行	BOC
中国建设银行	CCB
中国光大银行	CEB
兴业银行	CIB
中信银行	CITIC
招商银行	CMB
中国民生银行	CMBC
交通银行	COMM
华夏银行	HXB
中国工商银行	ICBC
中国邮政储蓄银行	POST
浦发银行	SPDB

附录 3：MD5 签名算法

1. 算法描述

为保证数据传输过程中的数据真实性，完整性和不可抵赖，我们需要对数据进行数字签名，在接收签名数据之后进行签名校验。签名算法为 MD5。签名是为了防止从商户系统提交的支付请求以及通知数据被非法篡改。根据接口报文内容，把除签名字段以外的所有非空字段内容按照报文字段的先后顺序(参数名字字段升序排序)，依次按照“字段名=字段值”的方式用“&”符号连接，最后将密钥连接在最后，使用 MD5 算法计算数字签名，填入签名字段。接受方应按相应步骤验证签名。

2. 算法示例

2.1. 请求参数

```
clientIp=103.251.99.164
remark=
nonceStr=f6e825bc4fef4679b01755421b8795a5
createTime=20180411155020
description=1804110033547100
extra=1804110033547100
trandNo=null
notifyType=MANUAL
notifyTime=20180411155020
orderNo=2877452431755264
orderStatus=PAY
outOrderNo=1804110033547100
payTime=20180411003351
startTime=20180411003334
amount=1000
```

2.2. 待签名参数

```
amount=1000&clientIp=103.251.99.164&createTime=20180411011032&description=1804110033547100&extra=1804110033547100&nonceStr=55343b4a48084bf18937b0aae5fa68cf&notifyTime=20180411011032&notifyType=AUTO&orderNo=2877452431755264&orderStatus=PAY&outOrderNo=1804110033547100&payTime=20180411003351&startTime=20180411003334&key=1234567890
```

注意事项：没有值的参数不需要参与签名，也无需包含到待签名数据中，计算签名前需要将参数按字段顺序升序排序，待签名字符串最后需将密钥（密钥不需要排序）拼接在最后；密钥（KEY）与每个商户一一对应，不需要传递。

2.3. 签名计算结果

```
2536B69E5E31A71B2D71A4083C30E670
```

2.4. 最终参数

```
description=test&nonce_str=111111&notify_url=aaaa&transAmountt=1&transDate=20160125091010&transNo=201601250910101223&sign=30CFAE4A0617880A7227C53AC0528AF0
```

3. 代码示例

(1)java 版算法示例

```

public static String  buildSgin(SortedMap<String,Object> map, String secretKey){
    if(map==null||map.size()==0){
        return "";
    }

    StringBuffer sb = new StringBuffer();
    Iterator<Map.Entry<String, Object>> it = map.entrySet().iterator();
    while (it.hasNext()) {
        Map.Entry<String, Object> entry = it.next();
        String k = entry.getKey();
        Object v = entry.getValue();
        if (null != v && !"".equals(v) && !"sign".equals(k)
            && !"key".equals(k)&&!"signature".equals(k)) {
            sb.append(k + "=" + v + "&");
        }
    }
    sb.append("key=" + secretKey);
    String sign = EncryptUtils.MD5(sb.toString());
    return sign.toUpperCase();
}

```

(2)php 版本算法示例

```

$params = $_POST;
$arr = $params;
unset($arr['key']);
ksort($arr);
$buff = "";
foreach ($arr as $x => $x_value){
    if($x_value != '' && !is_array($x_value)){
        $buff .= "{$x}={$x_value}&";
    }
}
$buff.="key={$params['key']}";
unset($params['key']);
$params['sign'] = strtoupper(md5($buff));

```