**速汇付-网关支付API接入文档**

[V1.0.1]

使用授权声明

本文档中所包含的信息属于商业机密信息，应严格控制使用范围。

如无电商付的书面许可，任何人员不得以任何介质方式持有或使用本文档的部分或全部内容。

**版本信息**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **编号** | **修订日期** | **修订内容** | **修订版本** | **修订人** |
| 01 | 2018-03-01 | 首次编写/创建 | V1.0.0 |  |
| 02 | 2018-04-21 | 修正错误字段的描述 | V1.0.1 |  |
|  |  |  |  |  |

# 网关支付概述

## 1.1 功能概述

网关支付是电商付针对互联网市场推出的安全支付服务。

商户可基于自己产品的特性和支付场景的需求，灵活定制支付业务，以提高用户（消费用户）的在线支付体验和支付效率。

用户（消费用户）只需使用网上支付功能，便能通过银行卡完成付款。

**备注：**

● 电商付在交易与支付过程中，未参与任何一方的资金结算或滞留；

● 电商付接入中国银行、民生银行、天下支付等10余家有结算资质的第三方，资金统一由第三方监管与结算；

● 商户在使用电商付的过程中，同时使用的是电商付聚合的10余家权威第三方。

## 1.2 阅读对象

● 与电商付签署合作意向（直接或间接签署）的企业或机构；

● 具有一定开发能力且了解支付业务的技术开发、管理人员等。

本文档对聚合支付进行了详细的描述，商户产品和技术人员通过该文档可以对聚合支付有全面了解，进而指导商户快速完成开发对接。

间接签署指的是：电商付的合作企业，在获取电商付的业务授权后，拓展的合作企业或机构。

## 1.3 准备工作

商户在与电商付（含业务服务商）完成线下协议签订后，由电商付平台管理员在“电商付支付接入系统”开立商户账户，且该账号电商付平台唯一。

通信安全协议采用 RSA 算法，商户订单信息采用**商户私钥证书签名**，**电商付公钥证书加密**。商户收到电商付返回的信息，用**商户私钥证书解密**，**电商付公钥证书验签**，确保双方交互的信息加密传输，安全可靠防抵赖。

## 1.4 业务术语

|  |  |
| --- | --- |
| **术语** | **解释** |
| 电商付商户 | 接入电商付支付系统且开通电商付 API接口服务的机构或企业 |
| 电商付商户号 | 电商付对每一位开通API接口商户，生成与使用唯一身份识别码-电商付商户号 |
| 商户用户 | 非电商付用户，指商户（自己的）业务系统的使用或消费用户-支付发起人 |
| 电商付商户交易单号 | 商户发起的一笔交易，通过电商付请求-下单即生成电商付唯一交易单号 |
| 商户交易单号 | 非电商付系统，商户的交易，在推送给电商付之前需要在商户（自己的）业务系统，生成商户（自己的）的唯一交易单号 |
| 交易状态 | 当前交易订单的状态（含是否支付），待支付-下单成功、支付成功、支付失败（含异常）、支付关闭，详见“**4.2 交易状态码**” |

# 开发规范与安全

## 2.1 商户交易单号规范

● 商户唯一订单号:商户订单号在电商付支付平台为唯一标识，商户订单号与自然日无关。

● 订单结果查询:商户号+订单日期+商户订单号查询唯一订单状态。

● 订单防重:单一商户在电商付支付系统内，商户订单号决定唯一订单，不可重复进行支付，商户重复发送已成功订单时会返回成功状态并示已成功请勿 重复支付。

## 2.2 重复订单处理规范

电商付允许商户提交重复的订单（即商户订单日期和订单号均不变）。由于网络的不稳定性等原因，相同订单的交易应答返回给商户订单系统时，极有可能先支付完成的订单后送达商户，对于同一笔订单收到两次支付结果，处理的原则是：

● 只要商户收到订单支付成功的应答，即使以后再次收到订单支付失败的应答，则商户订单系统不做任何处理；

● 商户订单系统对于某笔订单已经收到成功应答，有可能后续再次收到该笔订单的成功应答，则商户只需要回应速汇科技平台收到成功即可，商户的账务数据不应被修改。

## 2.3 日期规范

考虑到商户订单系统主机时间和电商付平台主机时间会有微小差别，并且考虑到地区时区差别，允许商户订单日期和电商付平台日期前后相差一天，但所有给商户的清算和对账数据都以电商付平台的清算日期为准。

## 2.4 后台异步通知

电商付支付接受商户支付请求并处理成功后，为确保交易结果能够及时通知到商户，除实时接口响应外，电商付支付还会增加后台异步通知到商户指定的服务器地址(notifyUrl)，异步通知24小时内最多通知8次，每次通知后间隔时间递增(如：0m~2m~5m~10m~1h~2h~6h~15h)，期间只要有一次收到商户响应接收成功(SUCCESS)，则该笔订单结果通知结束。

## 2.5 开发说明

电商付提供的 API 接口有统一的明文报文格式。明文部分为 json格式。明文报文格式区分为2个部分。businessHead部分存放公共请求头信息，businessContext部分存放业务信息。businessHead和businessContext也都是json格式。示例如下：

|  |
| --- |
| {  "businessHead": {  "h1": "head1",  "sign": "xxxxxxxxx"  },  "businessContext": {  "c1": "context1",  "c2": "context2"  }  } |

## 2.6 请求头详解

电商付的API接口，商户发起的请求默认包含公共请求报文头，电商付支付返回的响应默认包含公共响应报文头，后面快捷绑卡、解绑、支付等接口定义 不再单独列出公共报文头内容，请在后期开发中注意。如果返回的报文包含错误信息，则返回报文中的字段都可能为空(公共响应报文头不可空字段除外)。

## 2.7 签名详解

将businessContext参数值的参数按照参数名ASCII码从小到大排序（字典序拼）接成 json 字符串进行签名。暂时只支持RSA。

## 2.8 步骤详解

1.商户后台按照要求，生成对应订单信息，将订单信息拼接成 json 字符串

{

"businessHead": {

"merchantNumber": "PAY000000000001”,

"version": "V1.0",

"requestTime": "20180410164822"

},

"businessContext": {

"amount": "10",

"currency": "CNY",

"accName": “赵六”,

"cardNo": “000000000000000002”,

"cellPhone": “18700000002”,

"bankCode": "",

"bankName": "",

"bankBranchName": "",

"cardType": "SAVINGS",

"accType": "PRI",

"cvv2": "",

"validPeriod": "",

"orderNumber": "20180410164822",

"remark": "测试备注",

"bankRemark": "测试备注",

"remark1": "",

"remark2": "",

"remark3": ""

}

}

2.对businessContext的json字符串以参数名称按照ASCII码从小到大排序（a-z）（字典序拼）

{

"accName": “赵六”,

"accType": "PRI",

"amount": "10",

"bankBranchName": "",

"bankCode": "",

"bankName": "",

"bankRemark": "测试备注",

"cardNo": "000000000000000002",

"cardType": "SAVINGS",

"cellPhone": "18700000002",

"currency": "CNY",

"cvv2": "",

"orderNumber": "20180410165434",

"remark": "测试备注",

"remark1": "",

"remark2": "",

"remark3": "",

"validPeriod": ""

}

3.使用商户的RSA私钥对businessContext字符串进行RSA签名生成sign

eaKB5TJTobt0qUJvqln/rSI3H9mMGWBwuZIwxI1V4clN4D8SjigsOqgDK8alEOdAuBsizS4CP6VL4Mpvq/aVbwc4BtOFzwSSEz1cD38KG31gxYw9eb3DmpkMDOy1UCSjV9aaQ+inn1jacX8f8fd9Te40R0XdNWjsnxXV2IqKvMA=

4.sign拼接到businessHead中生成

{

"businessContext": {

"amount": "10",

"currency": "CNY",

"accName": “赵六”,

"cardNo": "000000000000000002",

"cellPhone": "18700000002",

"bankCode": "",

"bankName": "",

"bankBranchName": "",

"cardType": "SAVINGS",

"accType": "PRI",

"cvv2": "",

"validPeriod": "",

"orderNumber": "20180410165434",

"remark": "测试备注",

"bankRemark": "测试备注",

"remark1": "",

"remark2": "",

"remark3": ""

},

"businessHead": {

"merchantNumber": "PAY000000000001",

"version": "V1.0",

"requestTime": "20180416165434",

"sign": "eaKB5TJTobt0qUJvqln/rSI3H9mMGWBwuZIwxI1V4clN4D8SjigsOqgDK8alEOdAuBsizS4CP6VL4Mpvq/aVbwc4BtOFzwSSEz1cD38KG31gxYw9eb3DmpkMDOy1UCSjV9aaQ+inn1jacX8f8fd9Te40R0XdNWjsnxXV2IqKvMA="

}

}

5.使用电商付的RSA公钥进行RSA加密，生成context



6.构建发送的url

http://www.babaodao.com /api/pay/cardPay?context=

7.转发url，跳转到网银支付页面

8.用户支付完成后，点击返回商户页面，返回到商户提供的同步通知地址，商户收到同步参数，使用商户的RSA私钥对context进行RSA解密

{"businessContext":{"paySerialNo":"0220171016170712015236006501","merchantserialNo":"20180410164822","paymentType":"ISSU","effectiveTime":"","tradeState":"WAIT","stateExplain":"处理中","amount":101,"fee":100,"cardNo":"000000000000000002"},"businessHead":{"merchantNumber":"PAY000000000001","sign":"W/bkZwWrcvkOHVhgXhq0vRmiMOnHcQlEBQVEkxEPDlFRpj9ATtFX5oQNFhZF/w83cHeP3SWOBOd19LI5uQDgPeTqWBypvnVhH6i1U2tjXPIhFi4/jwsJSAj2aevz34PSfQbHMuxojIhl7gQpeAnypU+DA4KZYtsmGjZlrsBPudo="}}

9.使用电商付的RSA公钥，businessContext参数名称按照ASCII码从小到大排序（a-z）（字典序拼）的json 与sign进行签名比对，验签成功，返回商户自己的同步通知结果页（支付成功页面）

# API接口

## 3.1 公共报文头

● 公共请求报文

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| 字符集 | charset | 否 | String(2) | 00 | 00表示UTF-8，暂时只支持UTF-8 |
| 版本号 | version | 否 | String(10) | V1.0.0 | 默认为V1.0.0，采用向下兼容原则  传入版本为V1.1.0时，异步通知密文返回交易类型；明文返回商户单号。 |
| 电商付商户号 | merchantNumber | 是 | String(32) | PAY10017681024 | 商户在电商付注册与使用的商户编号 |
| 请求服务类型 | tradeType | 否 | String(20) | bankPayment | 各接口自定义，如该文档接口为：网关支付 |
| 请求时间 | requestTime | 是 | String(14) | 20180408161616 | 请求时间：yyyymmddhhmmss |
| 签名类型 | signType | 否 | String(10) | RSA | 暂时只支持RSA，必须大写的RSA |
| 签名信息 | sign | 是 | String |  | 使用商户证书对报文签名后值 |

● 公共响应报文

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| 字符集 | charset | 是 | String(2) | 00 | 00表示UTF-8，暂时只支持UTF-8 |
| 版本号 | version | 是 | String(10) | V1.0.0 | 默认为V1.0.0，采用向下兼容原则 |
| 电商付商户号 | merchantNumber | 是 | String(32) | PAY10017681024 | 商户在电商付注册与使用的商户编号 |
| 请求服务类型 | tradeType | 是 | String(20) | bankPayment | 各接口自定义，如该文档接口为：网关支付 |
| 响应时间 | responseTime | 是 | String(14) | 20180408161616 | 响应时间：yyyymmddhhmmss |
| 签名类型 | signType | 是 | String(10) | RSA | 暂时只支持RSA，必须大写的RSA |
| 签名信息 | sign | 是 | String |  | 使用商户证书对报文签名后值 |

## 3.2 网关支付(rpmbankPayment)

### 3.2.1接口说明

用户在商户页面进行支付时，先在商户网站选择支付的银行卡类型、付款的银行，选择后通过电商付的支付服务跳转到银行页面进行实际付款。

### 3.2.2 请求地址

正式环境请求URL：

测试环境请求URL：http://www.babaodao.com/api/pay/cardPay

请求方式：GET

请求服务类型：rpmbankPayment

### 3.2.3 详细参数说明

● 请求报文

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| 商户单号 | orderNumber | 是 | String(50) | 00 | 商户上送订单号，保持唯一值。 |
| 交易金额 | amount | 是 | String(20) | 1000 | 以分为单位，10.00元填入值为1000 |
| 币种 | currency | 是 | String(10) | CNY | CNY-人民币，默认为CNY |
| 商品名称 | commodityName | 是 | String(20) |  | 用于描述该笔交易或商品的主体信息 |
| 商品描述 | commodityDesc | 是 | String(500) |  | 用于描述该笔交易或商品的主体信息 |
| 商品备注 | commodityRemark | 否 | String(500) |  | 用于描述该笔交易或商品的主体信息 |
| 订单有效时间 | vaildTime | 否 | String(10) | 1800 | 以电商付服务器时间为准的订单有效时间长度。如果不填则采用默认值。单位：秒 |
| 支付类型 | payType | 是 | String(20) | UNION\_B2C\_SAVINGS | 详见”4.1：交易类型“ |
| 银行卡类型 | cardType | 是 | String(2) | SAVINGS | 银行卡类型。 储蓄卡：SAVINGS 信用卡：CREDIT |
| 银行编号 | bankNumber | 是 | String(10) | 1001 | 详见”4.3：银行代码“ |
| 页面返回地址 | returnUrl | 是 | String(255) | 0 | 支付成功后，跳转到商户定义的地址中（不能含有’字符，如果含有?&=字符, 必须先对该地址做URL编码） |
| 异步通知地址 | notifyUrl | 是 | String(255) | 0 | 网银支付结果后台通知地址。(不能含有’字符. 如果含有?&=字符, 必须先对该地址做URL编码) |

● 响应报文

无

## 3.3支付结果同步通知(rpmSyncNotify)

### 3.3.1 接口说明

接入的商户如果通过电商付完成了银行卡支付，不论支付结果是否成功，支付结果同步通知为电商付支付主动发起的请求，没有公共报文头定义。电商付都会通过该同步回调接口跳转到商户传给电商付的returnUrl页面，把支付结果信息传递给商户的显示页面。

### 3.3.2 请求地址

请求方式：GET

请求服务类型：rpmSyncNotify

### 3.3.3 详细参数说明

● 请求报文

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| 电商付单号 | payOrderNumber | 是 | String(50) |  | 电商付生成的订单号,维度唯一 |
| 商户单号 | orderNumber | 是 | String(50) |  | 商户上送订单号，保持唯一值。 |
| 交易金额 | amount | 是 | String(20) | 1000 | 以分为单位，10.00元填入值为1000 |
| 币种 | currency | 是 | String(10) | CNY | CNY-人民币，默认为CNY |
| 交易结果 | orderStatus | 是 | String(10) | 00 | 详见”附件4.2：交易状态码“ |

● 响应报文

无

## 3.4 异步通知（rpmAsyncNotify）

### 3.4.1接口说明

电商付接收商户支付请求并处理成功后，为确保交易结果能够及时通知到商户，除实时接口响应外，电商付还增加后台异步通知（发送通知至商户指定的服务器地址）；期间将会重复发送通知，直至商户接收成功。期间只要有一次收到商户响应接收成功(SUC)，则该笔订单结果通知结束

### 3.4.2请求地址

请求方式：POST

请求服务类型：unifiedPayAsyncNotify

### 3.4.3详细参数说明

● 通知报文

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| 电商付单号 | payOrderNumber | 是 | String(50) | 2017062616161001024 | 电商付生成的订单号,维度唯一 |
| 商户单号 | orderNumber | 是 | String(50) |  | 商户上送订单号，保持唯一值。 |
| 交易状态 | orderStatus | 是 | String(10) |  | 详见”附录4.2：交易状态码“ |
| 交易时间 | orderTime | 是 | String(14) |  | yyyyMMddHHmmss |
| 币种 | currency | 是 | String(10) | CNY | CNY-人民币，默认为CNY |
| 交易金额 | amount | 是 | String(20) | 1000 | 以分为单位，10.00元填入值为1000 |
| 交易手续费 | fee | 是 | String(20) | 1000 | 以分为单位，10.00元填入值为1000 |
| 交易类型 | payType | 否 | String | WECHAT\_NATIVE | 用户交易的支付方式，详见4.1 |

响应报文

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| 处理结果 | result | 是 | String(10) | SUC | 商户处理结果: SUC/FAILED |

注：商户在收到异步通知结果后，需要及时响应，电商付从而停止继续异步通知。

直接返回字符串SUC

## 3.5 支付结果查询（tradeQueryApi）

### 3.5.1 接口说明

用户交易或支付完成后，电商付商户可根据需要实时查询订单交易或支付结果。

### 3.5.2请求地址

正式环境请求URL：

测试环境请求URL：http://www.babaodao.com/api/query/queryTradeOrder

请求方式：POST

请求服务类型：tradeQueryApi

### 3.5.2详细参数说明

● 通知报文

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| 电商付交易单号 | payOrderNumber | 否 | String(32) | 0220171016170712015236006501 | 电商付平台生成的交易单号  与orderNumber二选一  优先级payOrderNumber>orderNumber |
| 商户交易单号 | orderNumber | 否 | String(64) | 10017681024 | 商户支付接口上送的商户叫订单号  与payOrderNumber二选一  优先级payOrderNumber>orderNumber |
| 交易类型 | payType | 否 | String(10) | WECHAT\_NATIVE | 详见“4.1 交易类型” |

● 响应报文

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **变量名** | **必填** | **类型** | **示例值** | **描述** |
| 电商付交易单号 | payOrderNumber | 是 | String(32) | 2017062616161001024 | 交易状态为：下单成功，该参数有值；  电商付支付交易单号 |
| 商户交易单号 | orderNumber | 是 | String(32) | 2017062616161001024 | 商户上传的交易单号 |
| 交易(支付)类型 | payType | 是 | String(16) | WECHAT\_NATIVE | 详见“4.1 交易类型” |
| 交易时间 | orderTime | 是 | String(14) | 2017080816161616001 | yyyyMMddHHmmss |
| 交易状态 | orderStatus | 是 | String(10) | SUC | 详见“附录2 交易状态编码” |
| 交易金额 | amount | 是 | String(12) | 1000 | 以分为单位，10.00元填入值为1000 |
| 币种 | currency | 是 | String(8) | CNY | CNY-人民币，默认为CNY |
| 交易手续费 | fee | 否 | Number(12) | 1000 | 交易状态为：支付成功，该参数有值；  以分为单位，10.00元填入值为1000 |

# 附录

## 4.1 交易类型

文档变量名：payType字段，将会层呈现对应字段名：

|  |  |  |
| --- | --- | --- |
| **交易类型** | **字段内容** | **说明** |
| B2B-网银 | UNION\_B2B |  |
| B2C-储蓄卡 | UNION\_B2C\_SAVINGS |  |
| B2C-信用卡 | UNION\_B2C\_CREDIT |  |
| 支付宝扫码 | ALI\_NATIVE |  |
| 支付宝h5 | ALI\_JSAPI |  |

## 4.2 交易状态码

文档变量名：orderStatus字段，将会层呈现对应字段名：

|  |  |  |
| --- | --- | --- |
| **状态** | **状态码** | **说明** |
| 待支付-下单成功 | WAIT | 交易下单成功，可还未支付；常见：未支付状态下，第三方无回执信息 |
| 支付成功 | SUC | 交易下单成功与支付成功，且电商付收到第三方成功回执 |
| 支付失败 | FAIL | 交易下单失败或该交易无法有效支付，例如：微信扫码支付，用户信息错误等 |
| 支付关闭 | CLOSE | 该状态由第三方自行撤销或关闭，且无回执；因此原则上电商付不存在该情况 |
| 处理中 | HANDLE | 支付处理中，可通过查询接口进行查询结果 |

## 4.3 银行代码

文档变量名：bankNumber字段，将会层呈现对应字段名：

备注：以下银行为支付支持银行。

|  |  |  |  |
| --- | --- | --- | --- |
| **支持的银行** | **银行代码** | **支持储蓄卡** | **支持信用卡** |
| 建设银行 | 1004 | 是 |  |
| 农业银行 | 1002 | 是 |  |
| 工商银行 | 1001 | 是 |  |
| 中国银行 | 1003 | 是 |  |
| 浦发银行 | 1014 | 是 |  |
| 光大银行 | 1008 | 是 |  |
| 平安银行 | 1011 | 是 |  |
| 兴业银行 | 1013 | 是 |  |
| 邮政储蓄银行 | 1006 | 是 |  |
| 中信银行 | 1007 | 是 |  |
| 华夏银行 | 1009 | 是 |  |
| 招商银行 | 1012 | 是 |  |
| 广发银行 | 1017 | 是 |  |
| 北京银行 | 1016 | 是 |  |
| 上海银行 | 1025 | 是 |  |
| 民生银行 | 1010 | 是 |  |
| 交通银行 | 1005 | 是 |  |
| 支付宝扫码 | 1020 | 是 |  |
| 支付宝h5 | 1021 | 是 |  |