**商米云支付**

**接入规范说明书**

V 1.5.0

**上海商米科技集团股份有限公司**

**修改历史记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **更新说明** | **修订** |
| 2019-06-11 | 1.0 | 初稿 | 吴代坤 |
| 2019-06-24 | 1.1 | 接口参数限制调整 | 张雷 |
| 2019-07-10 | 1.2 | 接口名称、参数等数据整体调整、改进，与终端L3接口尽量保持一致 | 张雷 |
| 2019-07-16 | 1.3 | 增加支付回调接口数据说明 | 张雷 |
| 2019-11-05 | 1.4 | 增加小程序/公众号支付相关接口，并优化各个接口的返回参数是否是必须的返回标识,返回参数类型  新增场景描述，明确字段含义，新增QA问答，按照场景结构来展现文档接口 | 吴代坤、于奎强 |
| 2020-02-24 | 1.4.1 | 鉴于对接反馈，为减少客户对接周期，新增统一下单接口，现将小程序、公众号支付的下单能力合并到统一下单接口中，并后续在统一下单接口中扩展更多场景的下单能力，客户已对接的1.4版本单一接口的能力依旧保留，不影响正常使用 | 吴代坤 |
| 2020-06-08 | 1.4.2 | 新增开发自助申请开发appId的介绍说明 | 张雷 |
| 2020-10-22 | 1.4.3 | 新增支付宝、微信侧付款码支付能力；扩展已有的小程序公众号支付能力，增加支付宝侧的生活号、小程序支付能力 | 程云波 |
| 2020-11-24 | 1.4.3 | 新增B扫C包名上报，新增B扫C经纬度上报 | 姜梦林 |
| 2021-06-24 | 1.5.0 | 升级新版本支付网关路由 | 姜梦林 |

目录

**[1. 文档说明](#_Toc1212240401_WPSOffice_Level1)** **[0](#_Toc1212240401_WPSOffice_Level1)**

[1.1. 功能描述](#_Toc811550990_WPSOffice_Level2) [0](#_Toc811550990_WPSOffice_Level2)

[1.2. 阅读对象](#_Toc1068846833_WPSOffice_Level2) [0](#_Toc1068846833_WPSOffice_Level2)

[1.3. 业务术语](#_Toc408015076_WPSOffice_Level2) [0](#_Toc408015076_WPSOffice_Level2)

[1.4. 技术服务](#_Toc594097461_WPSOffice_Level2) [0](#_Toc594097461_WPSOffice_Level2)

**[2. 签名加密](#_Toc811550990_WPSOffice_Level1)** **[0](#_Toc811550990_WPSOffice_Level1)**

[2.1. 签名生成方式](#_Toc1344552124_WPSOffice_Level2) [0](#_Toc1344552124_WPSOffice_Level2)

[2.2. 请求数据结果验签](#_Toc2064614334_WPSOffice_Level2) [0](#_Toc2064614334_WPSOffice_Level2)

[2.3. 支付回调验签](#_Toc932343312_WPSOffice_Level2) [0](#_Toc932343312_WPSOffice_Level2)

**[3. 统一支付](#_Toc1068846833_WPSOffice_Level1)** **[0](#_Toc1068846833_WPSOffice_Level1)**

[3.1. 应用场景](#_Toc1853356272_WPSOffice_Level2) [0](#_Toc1853356272_WPSOffice_Level2)

[3.2. 请求参数](#_Toc108563769_WPSOffice_Level2) [0](#_Toc108563769_WPSOffice_Level2)

[3.3. 请求数据报文样例](#_Toc1417649280_WPSOffice_Level2) [0](#_Toc1417649280_WPSOffice_Level2)

[3.4. 返回参数列表](#_Toc100385495_WPSOffice_Level2) [0](#_Toc100385495_WPSOffice_Level2)

[3.5. 响应数据报文样例](#_Toc1404351570_WPSOffice_Level2) [0](#_Toc1404351570_WPSOffice_Level2)

**[4. 交易查询](#_Toc408015076_WPSOffice_Level1)** **[0](#_Toc408015076_WPSOffice_Level1)**

[4.1. 应用场景](#_Toc2091556460_WPSOffice_Level2) [0](#_Toc2091556460_WPSOffice_Level2)

[4.2. 请求参数](#_Toc629605477_WPSOffice_Level2) [0](#_Toc629605477_WPSOffice_Level2)

[4.3. 请求数据报文样例](#_Toc1127323170_WPSOffice_Level2) [0](#_Toc1127323170_WPSOffice_Level2)

[4.4. 返回参数列表](#_Toc1819784356_WPSOffice_Level2) [0](#_Toc1819784356_WPSOffice_Level2)

[4.5. 响应数据报文样例](#_Toc653570718_WPSOffice_Level2) [0](#_Toc653570718_WPSOffice_Level2)

**[5. 交易退款](#_Toc594097461_WPSOffice_Level1)** **[0](#_Toc594097461_WPSOffice_Level1)**

[5.1. 应用场景](#_Toc184203021_WPSOffice_Level2) [0](#_Toc184203021_WPSOffice_Level2)

[5.2. 请求参数](#_Toc1376238620_WPSOffice_Level2) [0](#_Toc1376238620_WPSOffice_Level2)

[5.3. 请求数据报文样例](#_Toc2043608150_WPSOffice_Level2) [0](#_Toc2043608150_WPSOffice_Level2)

[5.4. 返回参数列表](#_Toc68726932_WPSOffice_Level2) [0](#_Toc68726932_WPSOffice_Level2)

[5.5. 响应数据报文样例](#_Toc1894827685_WPSOffice_Level2) [0](#_Toc1894827685_WPSOffice_Level2)

**[6. 付款码支付](#_Toc1344552124_WPSOffice_Level1)** **[0](#_Toc1344552124_WPSOffice_Level1)**

[6.1. 场景介绍](#_Toc1333900432_WPSOffice_Level2) [0](#_Toc1333900432_WPSOffice_Level2)

[6.2. 业务流程](#_Toc1282769591_WPSOffice_Level2) [0](#_Toc1282769591_WPSOffice_Level2)

[6.3. 统一支付](#_Toc920183704_WPSOffice_Level2) [0](#_Toc920183704_WPSOffice_Level2)

[6.4. 交易查询](#_Toc1497771081_WPSOffice_Level2) [0](#_Toc1497771081_WPSOffice_Level2)

[6.5. 交易退款](#_Toc235248233_WPSOffice_Level2) [0](#_Toc235248233_WPSOffice_Level2)

[6.6. 支付信息回调](#_Toc299657904_WPSOffice_Level2) [0](#_Toc299657904_WPSOffice_Level2)

**[7. 公众号支付](#_Toc2064614334_WPSOffice_Level1)** **[0](#_Toc2064614334_WPSOffice_Level1)**

[7.1. 场景介绍](#_Toc501240313_WPSOffice_Level2) [0](#_Toc501240313_WPSOffice_Level2)

[7.2. 业务流程](#_Toc1915077057_WPSOffice_Level2) [0](#_Toc1915077057_WPSOffice_Level2)

[7.3. 公众号域名绑定](#_Toc215195763_WPSOffice_Level2) [0](#_Toc215195763_WPSOffice_Level2)

[7.4. 统一支付](#_Toc432727193_WPSOffice_Level2) [0](#_Toc432727193_WPSOffice_Level2)

[7.5. 交易查询](#_Toc1466304009_WPSOffice_Level2) [0](#_Toc1466304009_WPSOffice_Level2)

[7.6. 交易退款](#_Toc1796629938_WPSOffice_Level2) [0](#_Toc1796629938_WPSOffice_Level2)

[7.7. 支付信息回调](#_Toc191807499_WPSOffice_Level2) [0](#_Toc191807499_WPSOffice_Level2)

**[8. 小程序支付](#_Toc932343312_WPSOffice_Level1)** **[0](#_Toc932343312_WPSOffice_Level1)**

[8.1. 场景介绍](#_Toc335681546_WPSOffice_Level2) [0](#_Toc335681546_WPSOffice_Level2)

[8.2. 业务流程](#_Toc360202953_WPSOffice_Level2) [0](#_Toc360202953_WPSOffice_Level2)

[8.3. 小程序域名绑定](#_Toc174630178_WPSOffice_Level2) [0](#_Toc174630178_WPSOffice_Level2)

[8.4. 统一支付](#_Toc1546739844_WPSOffice_Level2) [0](#_Toc1546739844_WPSOffice_Level2)

[8.5. 交易查询](#_Toc767011173_WPSOffice_Level2) [0](#_Toc767011173_WPSOffice_Level2)

[8.6. 交易退款](#_Toc1959935317_WPSOffice_Level2) [0](#_Toc1959935317_WPSOffice_Level2)

[8.7. 支付信息回调](#_Toc381211486_WPSOffice_Level2) [0](#_Toc381211486_WPSOffice_Level2)

[8.7.1.1. 接口文档参考支付信息回调](#_Toc1077726201_WPSOffice_Level2) [0](#_Toc1077726201_WPSOffice_Level2)

**[9. 支付信息回调](#_Toc1853356272_WPSOffice_Level1)** **[0](#_Toc1853356272_WPSOffice_Level1)**

[9.1. 接口请求参数](#_Toc1467181409_WPSOffice_Level2) [0](#_Toc1467181409_WPSOffice_Level2)

[9.2. 接口数据示例](#_Toc1510706209_WPSOffice_Level2) [0](#_Toc1510706209_WPSOffice_Level2)

[9.3. 回调返回信息](#_Toc740096182_WPSOffice_Level2) [0](#_Toc740096182_WPSOffice_Level2)

**[10. 金融刷卡查询](#_Toc108563769_WPSOffice_Level1)** **[0](#_Toc108563769_WPSOffice_Level1)**

[10.1. 场景介绍](#_Toc571247450_WPSOffice_Level2) [0](#_Toc571247450_WPSOffice_Level2)

[10.2. 业务流程](#_Toc1703990060_WPSOffice_Level2) [0](#_Toc1703990060_WPSOffice_Level2)

[10.3. 金融交易查询](#_Toc119022028_WPSOffice_Level2) [0](#_Toc119022028_WPSOffice_Level2)

**[11. 错误码](#_Toc1417649280_WPSOffice_Level1)** **[0](#_Toc1417649280_WPSOffice_Level1)**

[11.1. 系统错误码](#_Toc1095949239_WPSOffice_Level2) [0](#_Toc1095949239_WPSOffice_Level2)

[11.2. 业务错误码](#_Toc651619554_WPSOffice_Level2) [0](#_Toc651619554_WPSOffice_Level2)

**[12. QA](#_Toc100385495_WPSOffice_Level1)** **[0](#_Toc100385495_WPSOffice_Level1)**

[12.1. 常见问题](#_Toc1750728025_WPSOffice_Level2) [0](#_Toc1750728025_WPSOffice_Level2)

[12.2. 接口报错问题](#_Toc1812468628_WPSOffice_Level2) [0](#_Toc1812468628_WPSOffice_Level2)

[12.3. 支付回调问题](#_Toc104698101_WPSOffice_Level2) [0](#_Toc104698101_WPSOffice_Level2)

[12.4. 其他问题](#_Toc871876614_WPSOffice_Level2) [0](#_Toc871876614_WPSOffice_Level2)

1. 文档说明
   1. 功能描述

提供商米云交易相关API接口，以满足软件商的对接需求。

* 1. 阅读对象

本文档是面向具有一定的开发能力、了解计算机开发语言、SQL数据库语言、信息安全的基本概念的开发和维护人员。

* 1. 业务术语

|  |  |
| --- | --- |
| 术语 | 解释 |
| M | 必填字段 |
| O | 选填字段 |

* 1. 技术服务

在开发或使用商米接口时，产生疑问或遇到技术问题，可以联系商米相关技术支持人员并提交问题，获得在线支持。  
 开发过程中使用到的app\_id及app\_key可通过注册开发者平台账号并创建自己的应用来获得。开发者平台地址：https://partner.sunmi.com

1. **签名说明**
   1. 签名生成方式
      1. 签名必须参数

接口请求和生成签名必须依赖：app\_id、app\_key、privateKey此三个参数需要在开发者平台进行申请。

* + 1. 签名方式

请求方式为：

POST

请求必传header：

Sunmi-Timestamp // 当前时间戳

Sunmi-Sign // 签名内容  
Sunmi-Nonce // 6位随机数  
Sunmi-Appid // 申请的APPID

请求数据格式为 ：

application/json

请求body为json结构：

{

"merchant\_no":"业务json-body"

}

签名方式使用HMAC-SHA256 ：

HMAC-SHA256( json-body + appid + timestamp + nonce, appKey)

签名方式使用 RSA（收银台能力需要使用该签名，参考开发者平台能力说明） ：

Sha256WithRsa( json-body + appid + timestamp + nonce,RsaPrivateKey)

* 1. 请求数据结果验签

需要对请求结果的返回值进行签名验证，验签的签名生成方式参考2.1对应的签名方式进行验签。

* 1. 支付回调验签

需要对支付回调数据进行签名验证，验签对方式采用Sha256WithRsa 公钥验签，公钥需要从开发者平台获取appid对应对商米rsa公钥。

1. **统一支付**
   1. 应用场景

统一支付接口，开发者可以通过该接口完成各种类型交易订单的支付。

* 1. 请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 类型 | 必填 | 说明 |
| 接口说明：统一下单接口，支持各种场景统一下单，持续扩展 | | | |
| 请求地址：https://openapi.sunmi.com/v2/sunmipay/payment/trade/pay | | | |
| 请求方式：POST | | | |
| Content-type：application/json | | | |
| **请求header 基本参数** | | | |
| Sunmi-Timestamp | String(10) | M | 当前时间戳 |
| Sunmi-Sign | String | M | 签名内容，RSA |
| Sunmi-Nonce | String(6) | M | 6位随机数 |
| Sunmi-Appid | String(32) | M | 申请对APPID |
| **请求Body基本参数** | | | |
| merchantNo | String(20) | M | 商户号（商户进件成功商米创建商户的商户号) |
| totalAmount | String(20) | M | 交易金额格式：精确到分 最低1分钱 |
| orderId | String(80) | M | 商户订单号 |
| payScene | String(10) | M | 交易场景：  BC-付款码支付（B扫C）  JSAPI-公众号支付【微信公众号】【支付宝生活号】  MINI-小程序支付【微信小程序】【支付宝小程序】 |
| payType | String(10) | M | 交易类型：  WX-微信  ALI-支付宝  UNION 银联支付 |
| notifyUrl | String(250) | M | 支付结果通知地址 |
| terminalIp | String(20) | M | 终端IP：正确的用户端IP |
| orderTitle | String(128) | O | 订单描述【如不填写，默认为商户名称】 |
| expireTime | String(10) | O | 交易有效时间（如不填写，默认值为3，单位分钟，取值区间3~120） |
| miniAppId | String(32) | O | jsapi/公众号，商户公众号**AppID、小程序AppID**  （公众号和小程序支付必填） |
| userId | String(128) | O | jsapi/公众号，支付宝userId、微信openId【用户标识】  （公众号和小程序支付必填） |
| authCode | String(80) | O | B扫C，支付授权码【扫码支付使用】，B扫C交易时必传 |
| packageName | String(20) | O | B扫C，应用包名，B扫C交易时必传 |
| appVersion | String(80) | O | B扫C，应用版本号，B扫C交易时必传 |
| longitude | String(20) | O | B扫C，经度,121.615459404，B扫C交易时必传 |
| latitude | String(20) | O | B扫C，维度,31.4056441552，B扫C交易时必传 |

* 1. 请求数据报文样例

付款码支付场景：

{

"merchantNo": "8202915737",

"totalAmount": "0.01",

"orderId": "SFD42343242364723482",

"payScene": "BC",

"payType": "WX",

"orderTitle": "测试商品",

"expiredTime": "3",

"terminalIp": "192.168.1.10",

"userAuthCode":"8B2121CCCCEEA964A2B6",

"notifyUrl": "https://www.baidu.com",

}

公众号场景：

{

"merchantNo": "8202915737",

"totalAmount": "0.01",

"orderId": "SFD42343242364723482",

"payScene": "JSAPI",

"payType": "WX",

"miniAppId": "wx2e5f6590644de622", //微信公众号appId、

"userId": "oG-NZ5LHRuFlGF7bKEbXtjxkHS3s",//微信的openId

"appVersion": "", //小程序版本号，选填

"orderTitle": "测试商品" //选填,

"expiredTime": "3",

"terminalIp": "192.168.1.10",

"notifyUrl": "", //支付回调地址

}

小程序场景：

{

"merchantNo": "8202915737",

"totalAmount": "0.01",

"orderId": "SFD42343242364723482",

"payScene": "MINI",

"payType": "WX",

"miniAppId": "wx2e5f6590644de622", //小程序appId

"userId": "oG-NZ5LHRuFlGF7bKEbXtjxkHS3s",//微信的openId

"appVersion": "", //小程序版本号，选填

"orderTitle": "测试商品" //选填,

"expiredTime": "3",

"terminalIp": "192.168.1.10",

"notifyUrl": "", //支付回调地址

}

* 1. 返回参数列表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 类型 | 必填 | 说明 |
| data | String | O | 结果返回集 |
| code | Int(5) | M | 错误码 |
| msg | String | M | 错误信息 |
| **data响应参数** | | | |
| orderId | String | M | 商户订单号 |
| misId | String | M | 商米订单号 |
| platformId | String | M | 支付通道订单号 |
| merchantNo | String | M | 商户号 |
| amount | String | M | 交易金额 单位为分 |
| payTime | String | M | 交易时间,unix时间戳 |
| platform | String | M | 支付平台：  wxpay：微信支付  alipay：支付宝支付  unionpay：银联钱包支付 |
| cloudPayInfo | String | M | 以JSON字符串格式将结果返回,小程序调起支付数据签名字段列表,具体见下方cloudPayInfo参数 |
| buyerPayAmount | String | M | 实付金额 |
| resultMsg | String | M | 支付订单状态描述 |
| state | Int | M | 订单状态：  1-交易创建，交易提交到支付通道，待付款  2-订单未付款  3-交易成功，包含：收款、退款等成功  4-交易结束，不可退款，如撤单后原订单即为此状态  5-交易失败 |
| **data参数中cloudPayInfo明细参数【微信小程序、公众号】** | | | |
| appId | String | M | 微信分配的小程序ID |
| timeStamp | String | M | 时间戳 从1970年1月1日00:00:00至今的秒数,即当前的时间 |
| nonceStr | String | M | 随机串 |
| packageStr | String | M | 数据包，对应官方的package字段，统一下单接口返回的 prepay\_id 参数值，提交格式如：prepay\_id=wx2017033010242291fcfe0db70013231072 |
| signType | String | M | 签名方式 MD5 |
| paySign | String | M | 签名数据 |
| **data参数中cloudPayInfo明细参数【支付宝小程序、生活号】** | | | |
| tradeNo | String | M | 支付宝小程序生活号下单调起支付参数 |
| **data参数中cloudPayInfo明细参数【微信、支付宝C扫B场景返回支付二维码参数】** | | | |
| codeUrl | String | M | 微信、支付宝C扫B场景支付二维码Url |

* 1. 响应数据报文样例

{

"data":{

"cloudPayInfo":{

"appId":"wx154f9a63aef2864f",

"nonceStr":"W2ExnBJZfrYcqAargcKxD0FxkjRTJ81M",

"packageStr":"prepay\_id=wx11102037411915c588dec8290435820050",

"paySign":"rFlfcgmJY5WTHceTL1hL0rwsEftL0vxpY5XOUBLAL9uEi6pCw7IGf1qk1gJNRDkMO+paBRlUu9e0XI2m+5h5iyqsI6K8BjwOOBTIN115sZWFGT91+nohtnfUrnltmjwA0f90473khIuVQtqs7oEew+qFrDNYgHJenegwZHa+2tJ7rKVDakUIxOSuwg+Wj7j5WHZ82XkLBGjjX8nbwx00eG+FPF9eAqMuy2oLW4wOeyksfecesBjVl9h6Bks7C+BZmu87Csfxxr6tI/J9jVY94eECyJCHbt19iBJk9RfoM4zDEEAhKhpHsCU7c2APJAOL2JzgscpPNNbu8iHZA0HJwQ==",

"signType":"RSA",

"timeStamp":"1539224437"

},

"merchantNo":"82324238742",

"orderId":"18092110499722771691",

"misId":"18092110499722771691",

"platformId":"18092110499722771691",

"amount":"200",

"payTime":"20180921104914",

"platform":"wxpay",

"buyerPayAmount":"200",

"resultMsg":"用户支付中",

"state":1

},

"code":"1",

"msg":""

}

注意：当响应参数中出现有“用户支付中”的情况时，需要通过【查询接口】获取最终交易结果，建议在10秒（含）后开始查询！！！

* 1. 业务响应码

|  |  |
| --- | --- |
| **code(返回码)** | **msg(返回码描述)** |
| 14100 | 缺少业务参数 |
| 14101 | 交易单号重复 |
| 14103 | 商户不合法 |
| 14104 | 商户未激活通道 |
| 14105 | 设备不合法 |
| 14106 | 包名不合法或未绑定 |
| 14170 | 请求通道接口失败，请稍后重试 |

1. 交易查询
   1. 应用场景

该接口支持非金融刷卡类订单的查询，软件商(商户)可以通过该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：

1、当软件商(商户)其后台、网络、服务器等出现异常，支付结果未知时；

2、软件商终端扫码支付，返回系统错误或交易状态未知情况；

3、其他需要确定最终交易状态的情况。

* 1. 请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 类型 | 必填 | 说明 |
| 接口说明：根据订单号，设备SN，商户号等获取订单交易明细，支持非金融刷卡类交易的订单查询 | | | |
| 请求地址：https://openapi.sunmi.com/v2/sunmipay/payment/trade/query | | | |
| 请求方式：POST | | | |
| Content-type：application/json | | | |
| **请求header 基本参数** | | | |
| Sunmi-Timestamp | String(10) | M | 当前时间戳 |
| Sunmi-Sign | String | M | 签名内容，RSA |
| Sunmi-Nonce | String(6) | M | 6位随机数 |
| Sunmi-Appid | String(32) | M | 申请对APPID |
| **请求Body基本参数** | | | |
| msn | String(64) | M | 商户设备SN信息(设备号msn、merchantNo必传一个，两个都传以msn为主，两个需要有绑定关系) |
| merchantNo | String(20) | M | 商米商户号(设备号msn和、merchantNo必传一个，两个都传以msn为主，两个需要有绑定关系) |
| misId | String(80) | O | 商米收银台订单号（misId、orderId、platformId必传一个） |
| orderId | String(80) | O | Saas软件交易订单号（misId、orderId、platformId必传一个） |
| platformId | String(80) | O | 支付平台订单号（misId、orderId、platformId必传一个） |

* 1. 请求数据报文样例

1、设备类交易查询，建议根据设备SN查询，商户号也可以，二选一

{

"msn": "V206D88200036",

"orderId": "40000734324823473432"，

}

2、非设备上交易查询，只能通过商户号查询

{

"merchantNo": "836234846239482342",

"orderId": "40000734324823473432"

}

* 1. 返回参数列表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 类型 | 可空 | 说明 |
| **基本响应参数** | | | |
| data | Object | O | 结果返回集 |
| code | Int(5) | M | 错误码 |
| msg | String | M | 错误信息 |
| **data响应参数** | | | |
| misId | String | M | 商米订单号 |
| orderId | String | O | Saas系统支付订单号 |
| businessId | String | O | 业务流水号 |
| platformId | String | O | 第三方平台订单号 |
| amount | Int | M | 交易金额 单位为分 |
| buyerPayAmount | Int | M | 用户实付金额 |
| refunded | Int | M | 当前订单退款总额 |
| orderType | Int | M | 订单类型，0 表示收款，1 表示退款 |
| platform | String | M | 支付平台：  wxpay：微信支付  alipay：支付宝支付  unionpay：银联钱包支付 |
| payTime | Int | M | 交易产生的时间，unix时间戳 |
| merchantNo | String | O | 商户号 |
| msn | String | O | 产生交易的设备sn号 |
| state | Int | M | 订单状态：  1-交易创建，交易提交到支付通道，待付款  2-订单未付款  3-交易成功，包含：收款、退款等成功  4-交易结束，不可退款，如撤单后原订单为此状态  5-交易失败 |
| resultMsg | String | O | 支付订单状态描述 |

* 1. 响应数据报文样例

{

"code":1,

"data":{

"amount":2,

"businessId":"",

"buyerPayAmount":2,

"misId":"wqwqqqwwwwww78",

"msn":"TL05D91940002",

"merchantNo":"",

"orderType":0,

"orderId":"",

"payTime":1561454238,

"platform":"alipay",

"platformId":"SDPPDS434890114",

"refunded":"2",

"resultMsg":"Success",

"state":3

},

"msg":"成功"

}

* 1. 业务响应码

|  |  |
| --- | --- |
| **code(返回码)** | **msg(返回码描述)** |
| 14100 | 缺少业务参数 |
| 14103 | 商户不合法 |
| 14104 | 商户未激活通道 |
| 14105 | 设备不合法 |
| 14106 | 包名不合法或未绑定 |
| 14130 | 交易不存在 |
| 14170 | 请求通道接口失败，请稍后重试 |

1. 交易退款
   1. 应用场景

当正常交易发生一段时间后，由于买家或者卖家的原因需要退款时，可以通过该退款接口发起退款申请，退款请求且验证成功后，会将支付款原路退还到买家帐号。

根据请求参数支持全额和部分退款:

1、通过商米设备上的收银台产生的交易退款，建议根据设备SN、商户号进行退款

2、云端交易只能使用商户号进行退款

* 1. 请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 类型 | 必填 | 说明 |
| 接口说明：云端退款接口，支持全额和部分退款 | | | |
| 请求地址：https://openapi.sunmi.com/v2/sunmipay/payment/trade/refund | | | |
| 请求方式：POST | | | |
| Content-type：application/json | | | |
| **请求header 基本参数** | | | |
| Sunmi-Timestamp | String(10) | M | 当前时间戳 |
| Sunmi-Sign | String | M | 签名内容，RSA |
| Sunmi-Nonce | String(6) | M | 6位随机数 |
| Sunmi-Appid | String(32) | M | 申请对APPID |
| **请求Body基本参数** | | | |
| orderId | String(80) | M | Saas系统退款单号 |
| businessId | String(80) | M | Saas系统业务订单号 |
| amount | String(20) | M | 退款金额，单位为分 |
| msn | String(64) | O | 商户设备SN信息(设备号msn和、merchantNo必传一个，两个都传以msn为主，两个需要有绑定关系) |
| merchantNo | String(80) | O | 商户号(设备号msn和、merchantNo必传一个，两个都传以msn为主，两个需要有绑定关系) |
| oriMisId | String(80) | O | 原收银台流水号（oriMisId、oriPlatformId、oriOrderId必传一个） |
| oriPlatformId | String(80) | O | 原第三方支付平台订单号（oriMisId、oriPlatformId、oriOrderId必传一个） |
| oriOrderId | String(80) | O | 原Saas系统支付订单号（oriMisId、oriPlatformId、oriOrderId必传一个） |

* 1. 请求数据报文样例

{

"businessId":"453975342343",

"orderId":"sda1212",

"msn":"V206D88200036",

"amount":"1",

"oriOrderId":"1489834523642384"

}

* 1. 返回参数列表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 类型 | 可空 | 说明 |
| **基本响应参数** | | | |
| data | Object | O | 结果返回集 |
| code | String | M | 错误码 |
| msg | String | M | 错误信息 |
| **data响应参数** | | | |
| misId | String | M | 收银台流水号 |
| orderId | String | M | Saas系统支付订单号 |
| businessId | String | O | 业务流水号 |
| platformId | String | M | 第三方平台订单号 |
| oriMisId | String | M | 原收银台流水号 |
| oriOrderId | String | O | 原软件订单号 |
| amount | Int | M | 退款金额 单位为分 |
| orderType | Int | M | 订单类型，0 表示收款，1 表示退款 |
| platform | String | M | 支付平台：  wxpay：微信支付  alipay：支付宝支付  unionpay：银联钱包支付 |
| payTime | Int | M | 退款时间，unix时间戳 |
| merchantNo | String | O | 商户号 |
| msn | String | O | 产生交易的设备sn号 |
| state | Int | M | 订单状态：  1-交易创建，交易提交到支付通道，待付款  2-订单未付款  3-交易成功，包含：收款、退款等成功  4-交易结束，不可退款，如撤单后原订单即为此状态  5-交易失败 |
| resultMsg | String | O | 支付订单状态描述 |

* 1. 响应数据报文样例

{

"code":1,

"data":{

"amount":1,

"businessId":"",

"misId":"F96BDF328214623929EC91",

"msn":"TL05D91940002",

"merchantNo":"34789374374",

"orderId":"PS201905221509421",

"orderType":1,

"oriMisId":"wqwqqqwwwwwwweqwhhh",

"oriOrderId":"",

"payTime":1561460091,

"platform":"alipay",

"platformId":"SDPPDS434916053",

"resultMsg":"成功",

"state":3

},

"msg":"成功"

}

* 1. 业务响应码

|  |  |
| --- | --- |
| **code(返回码)** | **msg(返回码描述)** |
| 14100 | 缺少业务参数 |
| 14103 | 商户不合法 |
| 14104 | 商户未激活通道 |
| 14105 | 设备不合法 |
| 14106 | 包名不合法或未绑定 |
| 14130 | 交易不存在 |
| 14131 | 交易已完结 |
| 14132 | 交易金额超限 |
| 14133 | 当日交易不支持部分退款 |
| 14170 | 请求通道接口失败，请稍后重试 |

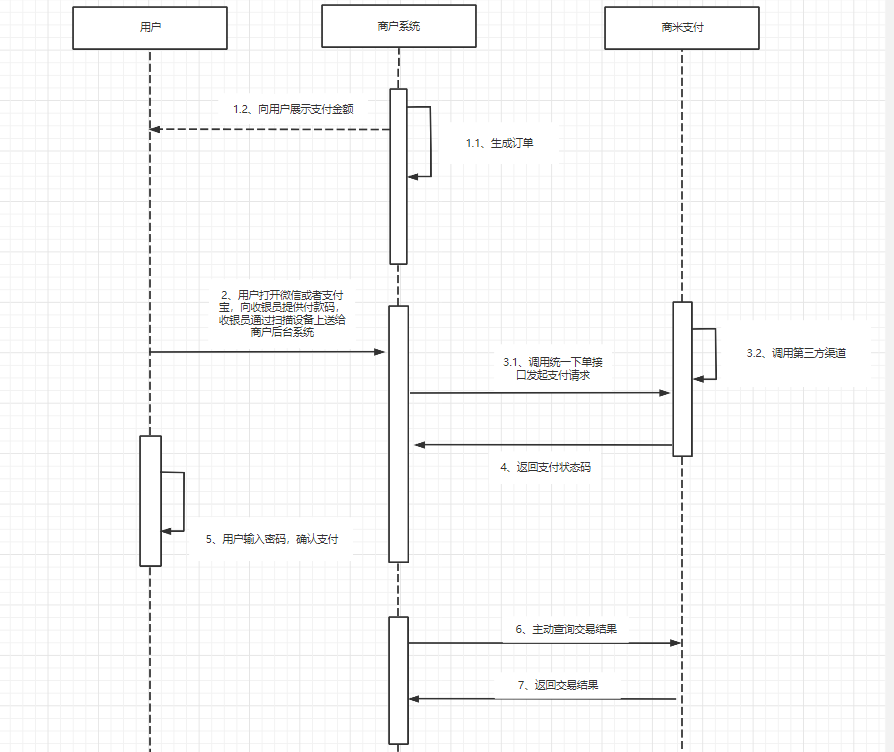
1. 付款码支付
   1. 场景介绍

软件商(商户)通过自己的线下门店，对消费者的消费进行收款。收银员通过设备上的软件系统生成支付订单，用户确认支付金额，收银员使用扫码设备扫描用户的支付宝、微信的条码或二维码，软件系统提交支付。用户输入密码之后，完成支付。

* 1. 业务流程

付款码支付适用于有开发能力或者购买已对接过商米支付的软件通过扫码用户支付码进行收款。若商户使用软件商的软件完成支付，软件商需要配置商户的信息到自己的系统中并在支付时将商户信息通过接口透传。

主要流程如下：



* 1. 统一支付
     1. 应用场景

软件商(商户)通过自己的线下门店，对消费者的消费进行收款。收银员通过设备上的软件系统调用统一支付接口生成支付订单，用户确认支付金额，收银员使用扫码设备扫描用户的支付宝、微信的条码或二维码，软件系统提交支付。用户输入密码之后，完成支付。

* + 1. 接口文档参考[统一支付](#统一支付)
  1. 交易查询
     1. 接口文档参考[交易查询](#交易查询)
  2. 交易退款
     1. 接口文档参考[交易退款](#交易退款)
  3. 支付信息回调
     1. 接口文档参考[支付信息回调](#支付信息回调)

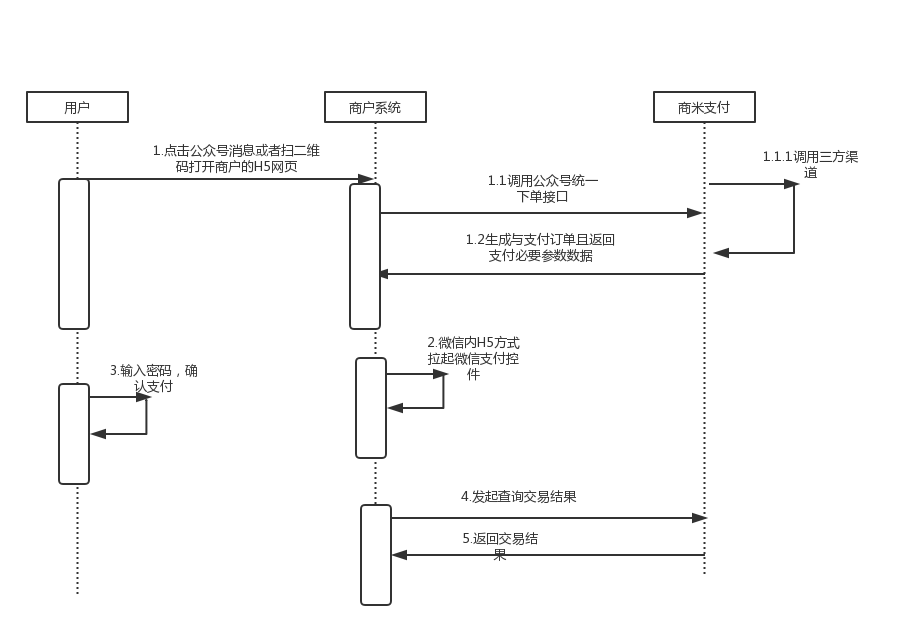
1. 公众号支付
   1. 场景介绍

软件商(商户)有一个在线售卖业务(如：在线商城)，欲通过微信公众号或者支付宝生活号开放给用户，用户在微信公众号/支付宝生活号里完成下单支付流程。

* 1. 业务流程

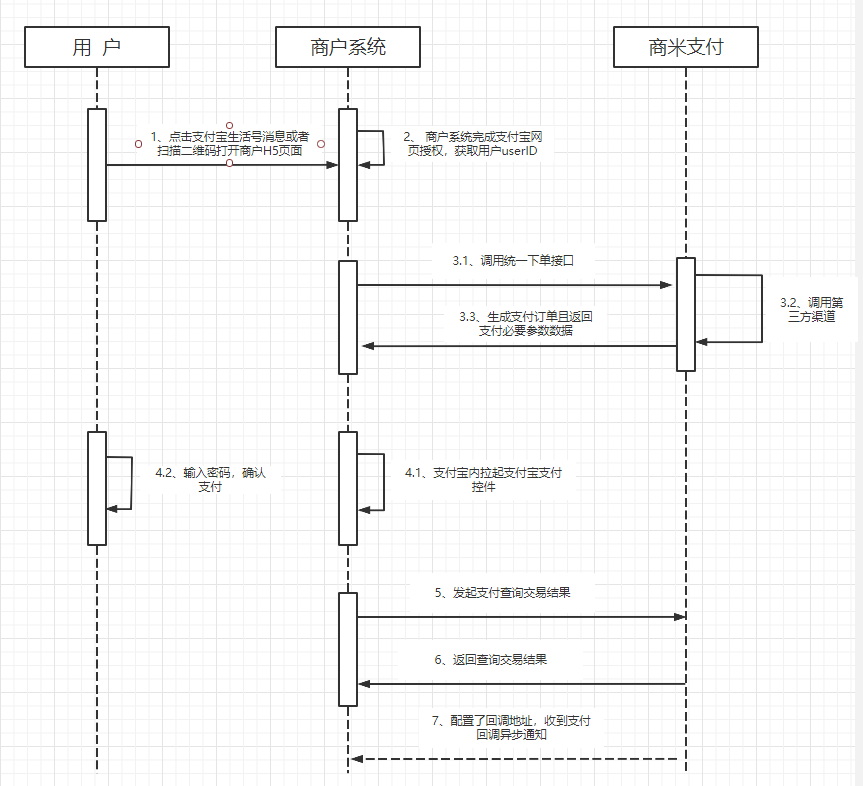
该公众号支付对应的是微信支付/支付宝支付的常规的普通模式，适用于有开发能力的商户或者有软件商代开发的直连商户收款。如果商户需要软件商帮助做公众号支付，需要以商户为主体注册公众号，将appid、appkey及mch\_id等参数托管到软件商。

以下是主要交互流程：

微信公众号：  


支付生活号：

备注：需要商户自行创建支付宝生活号，并且配置相应信息，完成网页授权，具体可参照支付宝官方文档。



* 1. 公众号域名绑定
     1. 应用场景

软件商(商户)有一个在线售卖业务(如：在线商城)，欲通过微信公众号开放给用户，用户在微信公众号里完成下单支付流程 。软件商(商户)如果要对接本文档的公众号下单接口，需要调本接口，完成绑定公众号的支付授权目录。

备注：该接口目前仅限于微信

* + 1. 请求参数列表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 类型 | | 必填 | 说明 |
| 接口说明：绑定公众号(或小程序)交易的支付授权目录 | | | | |
| 请求地址：https://openapi.sunmi.com/v2/sunmipay/payment/trade/payBind | | | | |
| 请求方式：POST | | | | |
| Content-type：application/json | | | | |
| 数据格式：请求表单格式 返回JSON格式 | | | | |
| **请求header 基本参数** | | | | |
| Sunmi-Timestamp | | String(10) | M | 当前时间戳 |
| Sunmi-Sign | | String | M | 签名内容，RSA |
| Sunmi-Nonce | | String(6) | M | 6位随机数 |
| Sunmi-Appid | | String(32) | M | 申请对APPID |
| **请求Body基本参数** | | | | |
| merchantNo | String(20) | | M | 商户号（商户进件成功商米创建商户的商户号，可通过商户终端获取) |
| configType | String(255) | | M | 1授权目录（公众号）  2小程序appId |
| miniAppId | String(32) | | O | 小程序appId或公众号的appid，添加授权目录或绑定小程序appid时必传 |
| payUrl | String(250) | | O | 授权目录； configType为1时必传，调起微信支付的页面的域名。payUrl要给最终目录的前一个目录才能匹配到，如：支付页面是http://xxxx.com/xxx/xxx.php，那么值需要给http://xxxx.com/xxx/ |

* + 1. 请求参数报文样例

**{**

**"merchantNo": "820291573790006",**

**"configType": "3",**

**"attentionAppId": "wx2e5f6590644de622",**

**}**

* + 1. 返回参数列表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 类型 | 可空 | 说明 |
| **基本响应参数** | | | |
| data | Object | O | 结果集 |
| code | String | M | 错误码 |
| msg | String | M | 错误信息 |

* + 1. 响应数据报文样例

成功响应：

{

"code":"1",

"data":"Success",

"msg":""

}

失败响应：

{

"code":"14106",

"data":{},

"msg":""

}

* + 1. 业务响应码

|  |  |
| --- | --- |
| **code(返回码)** | **msg(返回码描述)** |
| 14100 | 缺少业务参数 |
| 14103 | 商户不合法 |
| 14104 | 商户未激活通道 |
| 14106 | 包名不合法或未绑定 |
| 14170 | 请求通道接口失败，请稍后重试 |

* 1. 统一支付
     1. 应用场景

**1、微信公众号**

软件商(商户)有一个在线商城，欲通过公众号开放给用户，用户在公众号里完成下单支付流程。

1. 先调用公众号域名绑定，绑定授权目录以及公众号的APPID
2. 调用统一支付接口获取调微信支付的凭证参数，使用微信订单凭证参数通过JavaScript调用getBrandWCPayRequest接口，唤起微信支付。

3、微信调用参见：https://pay.weixin.qq.com/wiki/doc/api/jsapi.php?chapter=7\_7&index=6

**2、支付宝生活号：**

软件商（商户）有一个在线商户欲通过生活号开放给用户，用户在支付宝内完成下单支付流程：

1. 支付宝生活号开放平台配置好回调目录以及APPID信息。
2. 完成网页授权之后，调用此接口获取支付的凭证参数，利用参数通过javaScript调用 ap.tradePay()接口，唤起支付宝支付。
3. 支付宝拉起控件参见：

https://myjsapi.alipay.com/alipayjsapi/util/pay/tradePay.html

* + 1. 接口文档参考[统一支付](#统一支付)
  1. 交易查询
     1. 接口文档参考[交易查询](#交易查询)
  2. 交易退款
     1. 接口文档参考[交易退款](#交易退款)
  3. 支付信息回调
     1. 接口文档参考[支付信息回调](#支付信息回调)

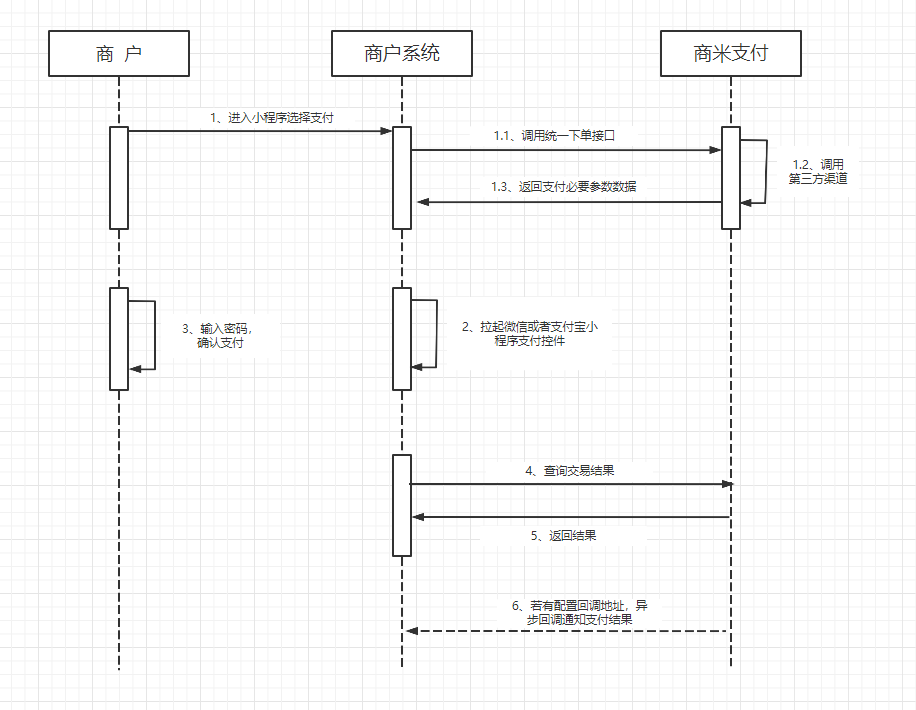
1. 小程序支付
   1. 场景介绍

软件商(商户)有一个在线售卖业务(如：在线商城)，欲通过微信(支付宝)小程序开放给用户，用户在微信小程序里完成下单支付流程。

* 1. 业务流程

该小程序支付对应的是微信（支付宝）支付的常规的普通模式，适用于有开发能力的商户或者有软件商代开发的直连商户收款。如果商户需要软件商帮助做小程序支付，需要以商户为主体注册小程序支付，将appid、appkey及mch\_id等参数给到软件商。

以下是主要交互流程：



* 1. 小程序域名绑定
     1. 应用场景

软件商(商户)有一个在线售卖业务(如：在线商城)，欲通过微信小程序开放给用户，用户在微信小程序里完成下单支付流程 。软件商(商户)如果要对接本文档的小程序下单接口，需要调本接口，完成绑定小程序appid。

注意：小程序支付只需要绑定小程序appid即可，示例如下：

{

"merchantNo": "820291573790006",

"configType": "2",

"miniAppId": "wxhsdfks24324534",

}

注意：微信小程序场景才需要调用此接口

* + 1. 接口文档参考[公众号域名绑定](#公众号域名绑定)
  1. 统一支付
     1. 应用场景

软件商(商户)有一个在线商城，欲通过小程序开放给用户，用户在小程序里完成下单支付流程。

1. 微信小程序
   1. 先调用公众号域名绑定，绑定小程序APPID
   2. 调用统一支付接口获取调起微信支付的凭证参数，拿这些凭证参数在小程序代码里调用wx.requestPayment唤起微信支付。具体唤起防范可参照支付文档：https://pay.weixin.qq.com/wiki/doc/api/wxa/wxa\_api.php?chapter=7\_7&index=5
2. 支付宝小程序
   1. 先调用【小程序域名绑定接口】，绑定小程序APPID
   2. 调用统一支付接口获取调起微信支付的凭证参数，拿这些凭证参数在小程序代码里调用my.tradePay()方法唤起支付宝支付。具体唤起方法可参照支付宝文档：https://opendocs.alipay.com/mini/006lmj
      1. 接口文档参考[统一支付](#统一支付)
   3. 交易查询
      1. 接口文档参考[交易查询](#交易查询)
   4. 交易退款
      1. 接口文档参考[交易退款](#交易退款)
   5. 支付信息回调
      * 1. 接口文档参考[支付信息回调](#支付信息回调)
3. 支付信息回调
   1. 接口请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 类型 | 必填 | 说明 |
| 接入方交易时提供接收回调的地址，开发者需要提供该接口的实现 | | | |
| 请求方式：POST | | | |
| Content-type：application/json | | | |
| **请求header 基本参数** | | | |
| Sunmi-Timestamp | String(10) | M | 当前时间戳 |
| Sunmi-Sign | String | M | 签名内容，RSA |
| Sunmi-Nonce | String(6) | M | 6位随机数 |
| Sunmi-Appid | String(32) | M | 申请对APPID |
| Sunmi-NotifyType | String(20) | M | 通知类型  交易通知：sunmipay\_payment\_trade |
| **基本请求参数** | | | |
| amount | Int | M | 交易金额，单位分 |
| businessId | String(80) | M | Saas业务流水号 |
| buyerPayAmount | Int | M | 用户付款金额 |
| misId | String(80) | M | 收银台订单号 |
| msn | String(64) | M | 产生交易的设备终端SN号 |
| orderId | String(80) | M | 商户订单号 |
| orderType | Int | M | 订单类型  0 表示消费  1 表示退款  2 表示撤单  3 表示预授权  4 表示预授权完成  5 表示撤销，支付通道为盛付通，则一般为后台撤单  6 表示订单查询 |
| payTime | Int | M | 支付时间，unix时间戳 |
| platform | String(8) | M | 支付平台：  wxpay：微信支付  alipay：支付宝支付  unionpay：银联钱包支付 |
| buyerLogonId | String(6) | M | platform-wxpay ：openId platform-alipay：userId |
| platformId | String(80) | M | 支付通道订单号 |
| refunded | String(20) | M | 退款总额 |
| resultMsg | String(128) | M | 错误信息 |
| state | Int | M | 订单状态  1-交易创建，交易提交到支付通道，待付款  2-订单未付款  3-交易成功，包含：收款、退款等成功  4-交易结束，不可退款，如撤单后原订单即为此状态  5-交易失败 |
| 以下字段只有此订单为退款订单时才会有 | | | |
| origMisId | String(80) | O | 原交易收银台订单号 |
| origOrderId | String(80) | O | 原交易商户订单号 |
| origPlatformId | String(80) | O | 原交易支付通道订单号 |
| origBusinessId | String(80) | O | 原交易Saas业务流水号 |

* 1. 接口数据示例

{

"amount":1,

"businessId":"",

"buyerPayAmount":1,

"misId":"test\_5F5EC7A54CDCC58BADC425",

"msn":"D804P89V00033",

"orderId":"15632714718760",

"orderType":0,

"payTime":1563271502,

"platform":"alipay",

"platformId":"2019071622001419071050445382",

"refunded":"0",

"resultMsg":"Success",

"state":3

}

* 1. 回调返回信息

接收方接收到接口请求后，返回字符串”success”7个字符，表示接收者已经接收到回调通知，若未收到回调响应将会重试进行回调，回调间隔频次：1s/15s/30s/3m/10m/10m/1h/2h/6h/15h，所有回调会在24h17m内回调完成。

1. 金融刷卡查询
   1. 场景介绍

商户使用商米金融设备进行刷卡，通过金融交易查询接口可以查询金融刷卡交易的支付状态。目前仅支持盛付通通道。

* 1. 业务流程

商户采用商米金融设备进行刷卡支付，这类订单最终的交易状态可以通过本章的交易查询接口进行状态的确认。

* 1. 金融交易查询
     1. 应用场景

软件商(商户)可以通过该接口主动查询金融类刷卡交易的订单状态，进而完成下一步的业务逻辑。  
需要调用查询接口的情况：  
 1、当软件商(商户)其后台、网络、服务器等出现异常，交易支付结果未知；  
 2、金融刷卡时返回系统错误或交易状态未知情况；其他需要确定最终交易状态的情况。

* + 1. 请求参数列表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名称 | 类型 | | 必填 | 说明 |
| 接口说明：根据交易流水号（商户订单号）参数获取刷卡类订单交易明细 | | | | |
| 请求地址：https://openapi.sunmi.com/v2/sunmipay/payment/trade/swipeQuery | | | | |
| 请求方式：POST | | | | |
| Content-type：application/json | | | | |
| **请求header 基本参数** | | | | |
| Sunmi-Timestamp | | String(10) | M | 当前时间戳 |
| Sunmi-Sign | | String | M | 签名内容，RSA |
| Sunmi-Nonce | | String(6) | M | 6位随机数 |
| Sunmi-Appid | | String(32) | M | 申请对APPID |
| **请求Body基本参数** | | | | |
| msn | String(64) | | M | 商户设备SN信息 |
| tradeDate | String(8) | | O | 指定交易日期。若为空，仅限当天交易(yyyyMMdd) |
| orderId | String(80) | | M | 交易流水号（商户订单号）（orderId、platformId必传其中一个，或者两个都提供） |
| platformId | String(80) | | O | 支付通道订单号（orderId、platformId必传其中一个，或者两个都提供） |

* + 1. 请求参数报文样例

{

"msn": "V206D88200036",

"tradeDate": "20190520",

"orderId": "201906300600003",

}

* + 1. 返回参数列表

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名称 | 类型 | 必填 | 说明 |
| **基本响应参数** | | | |
| data | String | O | 结果返回集 |
| code | Int(5) | M | 错误码 |
| msg | String | M | 错误信息 |
| **data响应参数** | | | |
| orderId | String | M | 软件订单号 |
| oriTradeRef | String | O | 原交易参考号 |
| platformId | String | M | 支付通道订单号，如盛付通订单号 |
| amount | Int | M | 交易金额，单位为分 |
| orderType | Int | M | 订单类型  0 表示消费  1 表示退款  2 表示撤单  3 表示预授权  4 表示预授权完成  5 表示撤销，支付通道为盛付通，则一般为后台撤单  6 表示订单查询 |
| payTime | Int | M | 交易时，unix时间戳 |
| merchantNo | String | M | 商户号 |
| state | Int | M | 订单状态：  1-交易创建，交易提交到支付通道，待付款  2-订单未付款  3-交易成功，包含：收款、退款等成功  4-交易结束，不可退款，如撤单后原订单即为此状态  5-交易失败 |
| resultMsg | String | O | 错误信息 |
| TransCurrency | String | M | 交易货币类型，如CNY |
| cardType | String | O | 卡类型 |
| cardGov | String | O | 发卡组织 |
| cardHash | String | O | 卡hash值 |
| shortCard | String | O | 卡号后4位 |
| terminalNo | String | M | 终端号 |
| traceNo | String | O | 终端流水号 |
| tradeRef | String | O | 交易参考号 |
| batchNo | String | O | 批次号 |

* + 1. 响应数据报文样例

{

"code": 1,

"data": {

"TransCurrency": "CNY",

"amount": 19900,

"batchNo": "000001",

"cardGov": "CUP",

"cardHash": "22005e1c89abc6c6b7090534c31180af38643346",

"cardType": "SCC",

"merchantNo": "820102179970017",

"orderType": 0,

"oriTradeRef": "",

"orderId": "201906300600003",

"payTime": 1561892597,

"platformId": "331916913",

"resultMsg": "交易成功",

"shortCard": "4402",

"state": 3,

"terminalNo": "10040784",

"traceNo": "000293",

"tradeRef": "454881473"

},

"msg": "成功"

}

* + 1. 业务响应码

|  |  |
| --- | --- |
| **code(返回码)** | **msg(返回码描述)** |
| 14100 | 缺少业务参数 |
| 14103 | 商户不合法 |
| 14104 | 商户未激活通道 |
| 14105 | 设备不合法 |
| 14130 | 交易不存在 |
| 14170 | 请求通道接口失败，请稍后重试 |

* + 1. 卡类列表

|  |  |
| --- | --- |
| **cardType** | **备注** |
| CC | 贷记卡 |
| DC | 借记卡 |
| SCC | 准贷记卡 |
| OA | 扫码：微信，支付宝，盛付通 |

* + 1. 发卡组织

|  |  |
| --- | --- |
| **cardGov** | **备注** |
| WX | 微信 |
| ZFB01 | 支付宝 |
| SFT01 | 盛付通 |
| CUP | 银行卡 |
| CUP01 | 银联扫码支付 |
| CUP02 | 银联静态扫码支付 |
| CUP03 | 银联动态扫码支付 |

1. 错误码
   1. 系统错误码

|  |  |
| --- | --- |
| **code(返回码)** | **msg(返回码描述)** |
| 20000 | 网关校验缺少参数 |
| 20001 | 请求超过有效期 |
| 30000 | 开发者身份验证失败 |
| 30001 | 开发者权限不足 |
| 40000 | 签名验证失败 |
| 40001 | 签名失败 |
| 50000 | 服务器异常 |
| 50001 | 网关异常 |

* 1. 业务错误码

|  |  |
| --- | --- |
| **code(返回码)** | **msg(返回码描述)** |
| 1 | 请求成功 |
| 14100 | 缺少业务参数 |
| 14101 | 交易单号重复 |
| 14102 | 交易金额有误 |
| 14103 | 商户不合法 |
| 14104 | 商户未激活通道 |
| 14105 | 设备不合法 |
| 14106 | 包名不合法或未绑定 |
| 14130 | 交易不存在 |
| 14131 | 交易已完结 |
| 14132 | 退款金额超限 |
| 14133 | 当日交易不支持部分退款 |
| 14170 | 请求通道接口失败，请稍后重试 |

1. QA
   1. 常见问题
      1. app\_id、app\_key怎么拿到？

答：通过商米开发者平台完成应用申请，partner.sunmi.com

* + 1. 退款场景支持哪几种？

答：B扫C、Jsapi、小程序交易（全额退款、部分退款）【当天到账】

* + 1. 刷卡交易查询接口支持的应用场景？

答：只支持金融刷卡类的交易查询。

* + 1. 支付回调接口能够配置几个？

答：

1. 支持通过开发者平台根据引用配置统一的交易回调地址
2. 支持通过统一交易接口传入回调地址进行回调
   * 1. 小程序支付、公众号支付的场景描述？

答：

1. 小程序支付目前只支持微信小程序支付，不支持支付宝小程序支付
2. 公众号支付目前仅支持微信公众号支付，支付宝的服务窗/生活号支付暂未开放
3. 小程序、公众号支付目前仅支持【一个商户一个APPID】模式对接，不支持【服务商APPID】模式以及多个商户使用一个APPID模式
   * 1. 小程序支付，公众号支付对接流程是怎样的？

答：

1. 商务合作确认，通过商米进件小程序完成正式商户开户
2. 通过开发者平台注册账号并创建应用，配置公私钥和回调地址
3. 调用【公众号域名绑定接口】绑定相关信息，之后开始调用相关相关
   1. 接口报错问题
      1. 接口报【订单不存在】

答：

1. 确认订单号，misId、orderId、msn是否传值错误
2. 确认订单产生的来源，设备SN，是否在A设备上产生的交易，调用查询接口入参传了B设备的SN号。
   * 1. 接口报【设备商户不合法】

答：

a) 确认设备SN号是否传值正确；

b) 确该设备是否绑定正确的商户，找技术支持人员确认。

* + 1. 接口报【退款金额有误】

答：

a) 入参传值错误，不符合规范，金额最小单位为分，注意最小传值为1，不是0.01；

b) 订单已经全部退过款，订单已是结束状态

* + 1. 接口报【系统异常】

答：联系技术支持人员确认，找到开发人员配合查找定位问题。

* + 1. 接口报【软件商不合法】

答：联系技术支持人员确认，找到开发人员配合查找定位问题。

* 1. 支付回调问题
     1. 支付回调验签出现错误

答：

a) 测试环境的支付回调验签使用文档最下方的商米公钥。

b) 线上环境的支付回调公私密钥信息会在申请的时候创建好，回调地址配置好之后邮件发送

* 1. 其他问题
     1. 订单相关

答：

a) code为1业务请求成功。

b) status、msg表示订单状态和对应的描述信息，对应文档下方的业务错误码描述。

c) code为1表示网络请求成功，订单状态的判断参考status状态。

* + 1. 查询相关

答：

1. P系列金融设备的查询
2. 扫码类交易查询接口调用[交易查询](#交易查询)

b) 刷卡类交易查询接口调用[金融交易查询](#金融交易查询)

c) 默认查询优先使用商米订单号【misId】做查询，对应Saas软件商自己的orderId

* + 1. 退款相关

答：

1. 如果有退款不到账的问题，联系技术支持人员，提供相关商户信息，设备信息，订单信息给到技术人员排查；
2. 退款到账时间，扫码类当日退款走撤销，实时到账，其他退款有部分延时到账，最晚第二天凌晨商户账户清洁算后到账。