图片 1

**中国建设银行对公专业结算综合服务平台服务接入方案**

中国建设银行北京开发中心

**2018**年**9**月

**修改记录**

| **编号** | **日期** | **描述** | **版本** | **作者** | **审核** | **发布日期** |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

本文档中所包含的信息属于机密信息，如无中国建设银行的书面许可，任何人都无权复制或利用。©Copy Right 2018 by China Construction Bank

[1. H5接入 5](#_Toc530128783)

[1.1. 交互流程 5](#_Toc530128784)

[2. SDK接入 7](#_Toc530128785)

[2.1. android端的接入 7](#_Toc530128786)

[2.1.1. 导入SDK包 7](#_Toc530128787)

[2.1.2. 配置AndroidManifest 7](#_Toc530128788)

[2.1.3. 接入方式 9](#_Toc530128789)

[2.2. IOS接入 10](#_Toc530128790)

[2.2.1. 导入SDK包及工程配置 10](#_Toc530128791)

[2.2.2. 接入方式 12](#_Toc530128792)

[3. 直连接口 13](#_Toc530128793)

[3.1. 直连接口调用说明 13](#_Toc530128794)

[3.1.1. 附件 13](#_Toc530128795)

[3.1.2. 身份认证 14](#_Toc530128796)

[3.1.2.1. SM2算法 14](#_Toc530128797)

[3.1.2.2. RSA算法 15](#_Toc530128798)

[3.1.3. 平台接口调用方法 16](#_Toc530128799)

[3.2. 接口描述 20](#_Toc530128800)

[3.2.1. 生成支付订单接口 21](#_Toc530128801)

[3.2.1.1. 访问路径 21](#_Toc530128802)

[3.2.1.2. 输入接口 21](#_Toc530128803)

[3.2.1.3. 输出接口 22](#_Toc530128804)

[3.2.2. 查询支付结果接口 23](#_Toc530128805)

[3.2.2.1. 访问路径 23](#_Toc530128806)

[3.2.2.2. 输入接口 23](#_Toc530128807)

[3.2.2.3. 输出接口 24](#_Toc530128808)

[3.2.3. 结果通知接口 24](#_Toc530128809)

[3.2.3.1. 输入接口 25](#_Toc530128810)

[3.2.3.2. 输出接口 25](#_Toc530128811)

[3.2.4. 支付订单退款接口 26](#_Toc530128812)

[3.2.4.1. 访问路径 26](#_Toc530128813)

[3.2.4.2. 输入接口 26](#_Toc530128814)

[3.2.4.3. 输出接口 27](#_Toc530128815)

[3.2.5. 退款结果通知接口 27](#_Toc530128816)

[3.2.5.1. 输入接口 28](#_Toc530128817)

[3.2.5.2. 输出接口 28](#_Toc530128818)

[3.2.6. 通知到货接口 28](#_Toc530128819)

[3.2.6.1. 访问路径 29](#_Toc530128820)

[3.2.6.2. 输入接口 29](#_Toc530128821)

[3.2.6.3. 输出接口 29](#_Toc530128822)

[3.2.7. 对账单推送接口 30](#_Toc530128823)

[3.2.7.1. 输入接口 30](#_Toc530128824)

[3.2.7.2. 文件模板 30](#_Toc530128825)

[3.3. 测试环境调用方式说明 31](#_Toc530128826)

[3.3.1. 测试环境域名 31](#_Toc530128827)

[3.3.2. 测试环境公私钥对 31](#_Toc530128828)

[3.4. 生产环境调用说明 31](#_Toc530128829)

[3.4.1. 生产环境域名 31](#_Toc530128830)

[3.4.2. 需市场方提供信息 32](#_Toc530128831)

[3.4.2.1. 市场方公钥 32](#_Toc530128832)

[3.4.2.2. 市场方结果通知回调地址 32](#_Toc530128833)

本文档仅适用于非跨商户合并订单支付、且分账规则预先设定的直联客户。

对于线上客户（即拟申请中国建设银行网络特约商户的客户），可采用服务调用方式完成支付。通过互联网访问对公专业结算综合服务平台的网址，并通过json串的形式传递参数。下面将介绍H5页面的接入方案。

# H5接入

客户自有平台通过互联网接入对公专业结算综合服务平台会涉及到调用以下几个接口，分别是**生成支付订单、查询支付结果、结果通知、支付订单退款、通知到货。**

## 交互流程



1. 首先需要市场自有平台调用对公专业结算综合服务平台的生成支付订单接口完成下单，对公专业结算综合服务平台会返回收银台的URL。
2. 市场自有平台跳转到收银台界面，消费者通过收银台完成支付。
3. 对公专业结算综合服务平台以服务器通知的形式将支付结果通知给市场自有平台，市场自有平台根据结果通知更新订单信息。浏览器上显示支付结果，并自动跳转回市场自有平台。
4. 如果市场自有平台未收到服务器通知，可以调用查询支付结果接口，主动发起查询，并根据查询结果更新订单状态。

# SDK接入

在接入SDK收银台之前，注意需要调用直连接口“生成支付订单接口”获取支付订单号和收银台URL。

下面将分android和ios两种移动端系统的SDK接入进行详细说明。

## android端的接入

### 导入SDK包

建行SDK包文件为ccbwlpay.aar,集成时将该文件复制到APP工程的libs文件夹下。右键关联到工程的类库中。

### 配置AndroidManifest

//建行支付所需权限

<uses-permission android:name="android.permission.INTERNET" />

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />

在AndroidManifest文件里注册建行支付需要的Activity

<activity android:name="android.ccb.llbt.sdklibrary.LoadCheckDeskActivity"  
 android:launchMode="singleTask"  
 android:theme="@style/Theme.AppCompat.NoActionBar"  
 android:screenOrientation="portrait" />

<activity android:name="com.ccb.ccbnetpay.activity.CcbH5PayActivity"

android:label="@string/app\_name"

android:configChanges="orientation|keyboardHidden|screenSize"

android:screenOrientation="portrait"/>

<activity android:name="com.ccb.ccbnetpay.activity.appresult.ResultActivity"

android:label="@string/app\_name"

android:configChanges="orientation|keyboardHidden|screenSize"

android:screenOrientation="portrait">

<intent-filter>

<action android:name="此处为THIRDAPPINFO"/>

<category android:name="android.intent.category.DEFAULT"/>

</intent-filter>

</activity>

<activity android:name="com.ccb.ccbnetpay.activity.CcbUnionPayActivity"

android:configChanges="orientation|keyboardHidden|screenSize"

android:screenOrientation="portrait"/>

<!-- wx sdk begin -->

绿色参数为必需配置

<activity android:name="com.ccbpay.test.wxapi.WXPayEntryActivity"

android:screenOrientation="portrait"

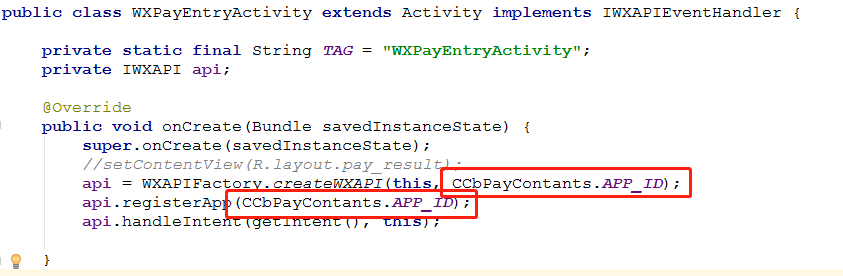
android:exported="true"

android:launchMode="singleTop"

android:theme="@android:style/Theme.Translucent.NoTitleBar"/> <!-- wx sdk end -->

微信平台规定接收支付结果的类WXPayEntryActivity，必须在商户应用APP的 包名.wxapi里，否则微信回调方法onResp(BaseResp resp)将不会调用。如demo的包名为com.ccbpay.test，需要在包里新建一个文件夹wxapi(必须是这个名称)，然后将WXPayEntryActivity复制到里面，修改该类的包路径package com.ccbpay.test.wxapi;即可；

在支付结果接收类WXPayEntryActivity中，需要先向微信注册APPID，该APPID为微信开放平台审核通过的应用APPID，在类中这两个地方填写即可，如下图



注册APPID

商户APP工程使用建行SDK微信支付前，需要先注册您的APPID（同上），在Application的子类中调用SDK中的init()方法，APP工程启动后即可完成注册，如下图所示



CCBWXPayAPI.*getInstance*().init(*context*, **"APPID"**);

### 接入方式

原生跳转方式

Intent intent =new Intent();  
intent.putExtra("Py\_Ordr\_No","105000007420264091719463501120");//订单号

intent.putExtra("url","");//订单号  
intent.setClass(MainActivity.this, CheckDeskActivity.class);//收银台界面

startActivity(intent);

## IOS接入

### 导入SDK包及工程配置

1、启动IDE（如Xcode），把iOS包中的压缩文件中的SDK文件夹拷贝到项目中并导入到项目工程中，如图1

添加到工程中并引用，以下文件(CCBGovPaySDK.framework, CCBNetPaySDK.framework,libWeChatSDK.a)和资源文件CCBGovPay.bundle拷贝到项目文件夹下，并导入到项目工程中(图1)。该SDK文件夹中会包含3个framework包和1个bundle资源文件。

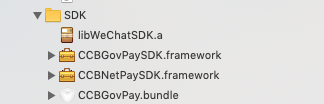


图1

2、检查TARGETS --> General --> Linked Frameworks and Libraries是否引用CCBGovPaySDK 和CCBNetPaySDK ，如(图2)。为了支持微信支付，请添加依赖库

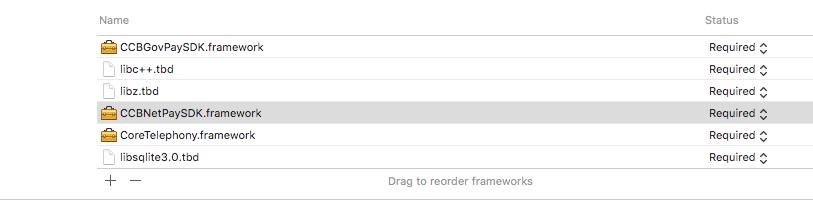


图2

3、在TARGETS --> Build Settings --> Linking --> Other Linker Flag路径下添加-ObjC，如图3。

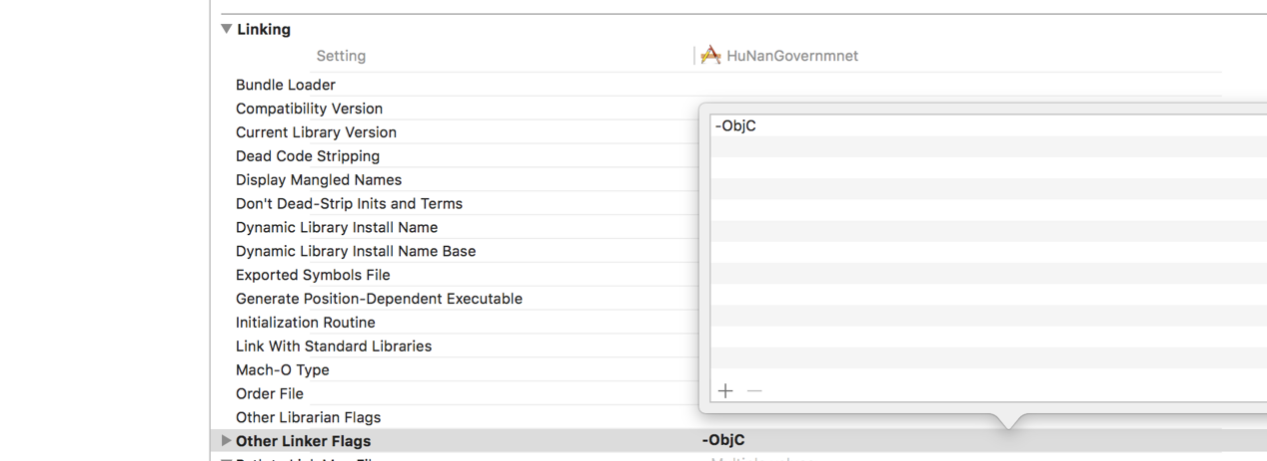


图3

4、在info.plist文件中的LSApplicationQueriesSchemes添加白名单mbspay（如果需要微信支付请加上 weixin），如图4。配置APP跳转后返回scheme(商品参数THIRDAPPINFO，其中固定为comccbpaycomccbgovpay)，如图5,如需调用微信支付，则需另外配置scheme，从微信开发平台注册app得到的APPID，如图6。配置ATS，如果Allow Arbitrary Loads已经设置为YES，则无需配置，如果为NO或没有设置，则需增加Exception Domains，如图7。

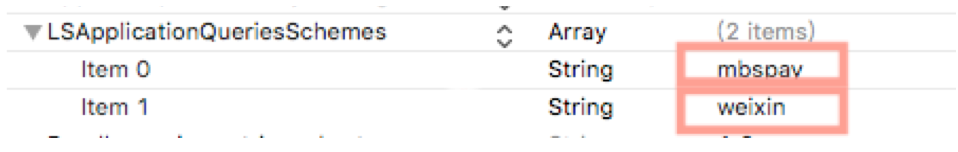


图4

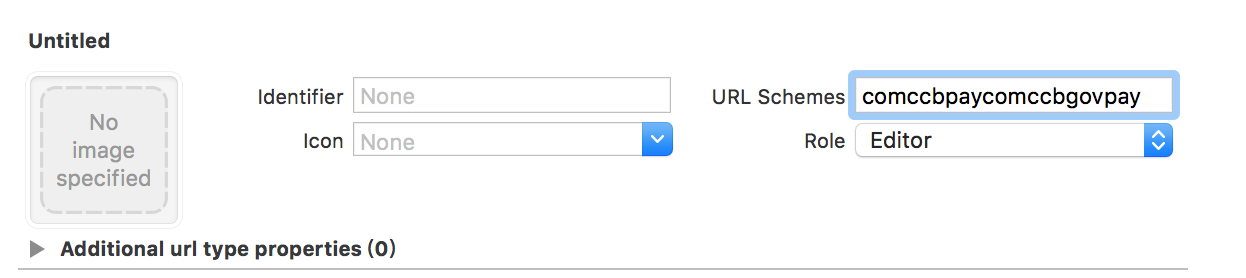


图5

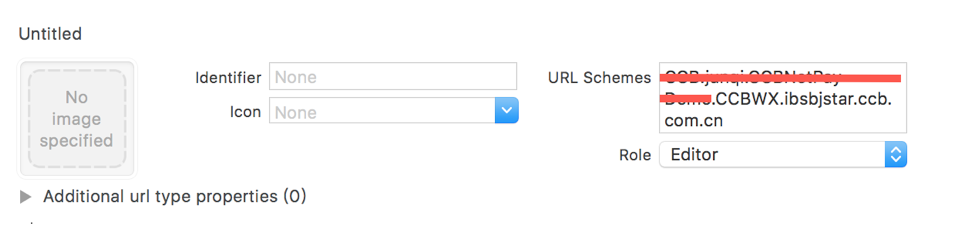


图6

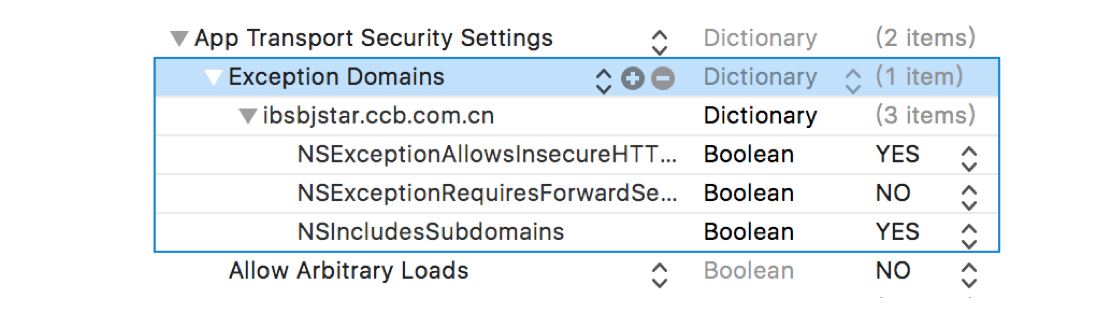


图7

5.注册appid，微信支付需在AppDelegate.m中的didFinishLauchingWithOptions:方法中 添加如下实现方法（传的参数为APPID，在微信开发平台注册app获得）：



**6支付宝支付采用的是保存二维码图片，然后用支付宝扫码的方式进行支付的方式。需配置访问相册权限说明：**

**NSPhotoLibraryAddUsageDescription**

### 接入方式

在接入页面中导入头文件：

#import

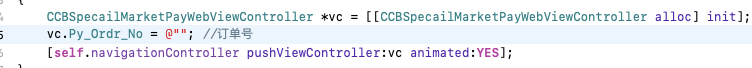
<CCBSpecialMarketPay/CCBSpecailMarketPayRootViewController.h> ,

在响应事件处：创建类CCBPayCheckstandWebViewController，

传入mainServiceURL参数，

从而调用CCBSpecialMarketPay.framework,进入支付流程

示例代码：



# 直连接口

## 直连接口调用说明

### 附件

附件中为对公专业结算综合服务平台接口调用的例子。



### 身份认证

开始调用接口前需要客户和平台（建行）各自调用建行开发的密钥生成器生成公钥和私钥，目前支持SM2算法和RSA算法来生成公私钥，推荐使用RSA算法，SM2为历史版本的支持（后期将不会再进行优化操作）。

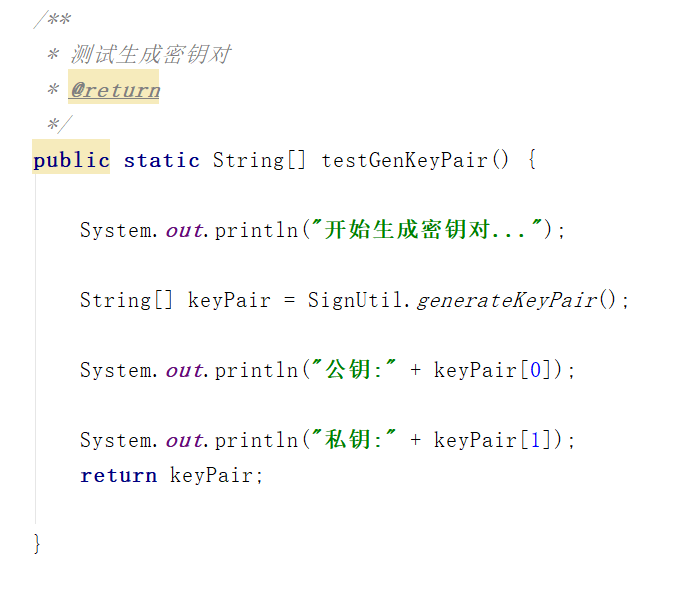
客户生成公钥和私钥后，客户将私钥自己留存，用于将来发送请求时生成签名信息；客户公钥则提供给平台（建行），平台通过该公钥对客户上送的签名信息验签。同时，平台（建行）也会生成一对公私钥，在返回请求数据时会用私钥来签名，公钥提供给客户用以验签。

#### SM2算法

如果采用SM2算法，需要将附件TestPay项目中的govpaysign.jar、bcprov-jdk15on-159.jar引用至java开发工程中。



可以在java代码中使用工具类SignUtil，这个工具类是签名包的封装类，签名、验签、生成密钥时都可使用这个类。SM2算法生成公钥和私钥可以使用SignUtil的SignuUtil.generateKeyPair()方法。具体代码可以参考SM2SignDemo中的testGenKeyPair()方法。

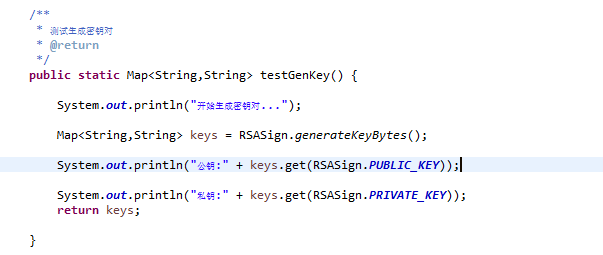


#### RSA算法

如果采用RSA算法，需要将附件TestPay项目中的mktpaysign.jar和commons-codec-1.11.jar引用到Java开发工程中。



采用RSA算法就可以在Java代码中使用工具类RSASignUtil,RSA算法生成公钥和私钥可以使用RSASignUtil的generateKeyBytes()方法。具体代码可以参考RSASignDemo中的testGenKey()方法。



### 平台接口调用方法

客户和平台生成各自的秘钥，并将公钥进行交换后，客户就可以进行平台接口的调用了。下面以生成支付订单接口为例说明平台接口的调用方式（具体代码可以参考TestPay工程中的TestPay.java）。

生成支付订单接口用于开发者生成支付订单信息。输入接口和输出接口中的参数都需要以json串的形式进行传递。

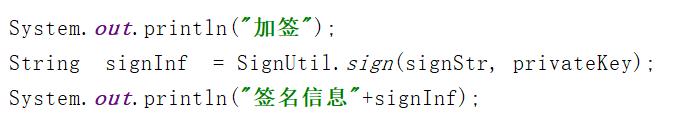
1. 签名信息的生成

在调用生成支付订单接口之前，首先要根据请求参数和客户之前生成的私钥生成签名信息，用于平台进行客户身份的验证。生成方法如下：

首先，将请求列表中除sign\_inf外的所有字段按照key=value&key=value的方式拼接成字符串signStr，其中请求字段中如果有列表格式的，则将列表中所有的对象字段按列表顺序依次拼接在字符串上。并且，key-value字符串拼接的先后顺序需要严格按照接口定义表格中字段先后顺序。如下图所示：



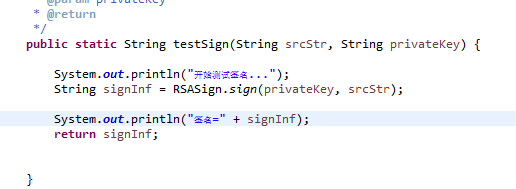
图中可以看到，需要发送的附加项列表和参与方列表，按照元素顺序依次拼接即可。得到signStr后，调用SignUtil的sign方法，传入两个参数signStr和客户私钥privateKey，生成签名信息signInfo，如下图所示：



如果要使用RSA算法，则需要在拼接字符串时添加Vno=2，添加字段Vno的目的是为了兼容SM2算法，虽然目前都支持，但建议使用RSA算法。Vno没值或为1时都使用旧算法SM2算法，为2时使用RSA算法进行加解密。



调用RSASign的sign()方法完成加签。



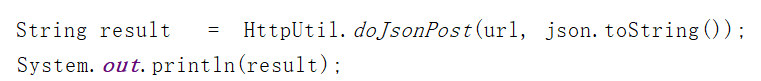
1. 发送请求

得到签名信息后，将请求的字段构造为json字符串(包括上面生成的签名信息)，然后将json字符串通过HttpUtil的doJsonPost()方法发送，得到返回信息。如下图所示：

生成需要发送的json字符串代码示例：



通过HttpUtil发送http请求，并得到返回的result，请求url为：



如果采用RSA算法，则需要将Vno=2添加到json串中：

1. 返回result的验证和解析

客户端收到平台返回的json格式结果信息后，解析json字符串，获得返回的各个字段，并将返回信息中除签名信息外的其他字段信息按照接口文档表格的顺序拼接成key=value&key=value格式。根据之前交换得到的平台公钥pubKey和发送来的签名信息signInfo，进行平台身份校验。校验采用SignUtil的verify()方法。如果校验结果为true则说明身份验证通过，可以进行下一步操作，否则身份验证失败。如下图所示：



如果采用RSA算法，检验则采用RSASign的*verifySign()方法。*

## 接口描述

接入接口有三个通用字段，服务响应状态、服务响应码、响应信息。此三个字段不参与签名验签操作，只用于表示交易的状态。比如交易成功时，返回的服务响应状态为00，交易失败时，服务响应状态为01，服务响应码对应的是错误代码，响应信息对应的是错误的描述信息。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Svc\_Rsp\_St | 服务响应状态 | VarChar2 | 2 | 必输 | 00-成功  01-失败 |
| Svc\_Rsp\_Cd | 服务响应码 | VarChar2 | 12 | 非必输 |  |
| Rsp\_Inf | 响应信息 | VarChar2 | 600 | 非必输 |  |

### 生成支付订单接口

用于开发者生成支付订单信息。输入接口和输出接口中的参数都需要以json串的形式进行传递，同时输入接口和输出接口中的参数都需要按照key=value&key=value的形式拼接字符串进行签名和验签，拼接字符串的字段顺序即为接口字段定义的顺序。可以参考附件里TestPay工程中的TestPay.java。

#### ****访问路径****

http://marketpay.ccb.com/online/direct/placeorder

#### ****输入接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Ittparty\_Stm\_Id | 发起渠道编号 | VarChar2 | 5 | 必输 | 默认送5个0 |
| Py\_Chnl\_Cd | 支付渠道代码 | VarChar2 | 25 | 必输 | 默认送25个0 |
| Ittparty\_Tms | 发起方时间戳 | VarChar2 | 23 | 必输 | yyyymmddhhmmssfff  年月日, 时分秒，毫秒 |
| Ittparty\_Jrnl\_No | 发起方流水号 | VarChar2 | 32 | 必输 | 该笔直连交易的客户方流水号（不允许重复） |
| Mkt\_Id | 市场编号 | VarChar2 | 30 | 必输 |  |
| Mkt\_Mrch\_Id | 商家编号 | VarChar2 | 30 | 必输 |  |
| Udf\_Id | 商家自定义编号 | VarChar2 | 30 | 非必输 |  |
| Pymd\_Cd | 支付方式代码 | VarChar2 | 2 | 必输 | 01 PC端  02 APP内唤醒  03 移动端H5页面（包含微信公众号/小程序/微信H5页面等）  04线下收银台 |
| Onln\_Ofln\_Indcd | 线上线下标志代码 | Char | 1 | 必输 | 1-线上支付（目前只支持线上方式） |
| Cmdty\_Ordr\_No | 商品订单号 | VarChar2 | 40 | 必输 | 不允许重复 |
| Opr\_No | 操作员号 | CHAR | 8 | 非必输 | 线上线下标志代码 为 1-线上支付 时，操作编号必输 |
| Usr\_Id | 用户ID | Char | 10 | 非必输 | 登陆市场管理方自有系统的用户ID |
| Ccy | 币种 | Char | 3 | 必输 | 156人民币 |
| Pgfc\_Ret\_Url\_Adr | 页面返回URL地址 | VarChar2 | 3000 | 非必输 | 支付完成后，跳转到的页面URL地址 |
| Clrg\_Rule\_Id | 分账规则编号 | VarChar2 | 32 | 必输 |  |
| Ordr\_Amt | 订单金额 | NUMBER | 19,8 | 必输 | 订单商品总金额，即应付金额 |
| Txnamt | 交易金额 | NUMBER | 19,8 | 必输 | 消费者实付金额 |
| Vno | 版本号 | NUMBER | 9 | 非必输 | 默认版本为1 |
| Prjlist |  |  |  |  | 加项减项列表 |
| ..Prj\_Id | 项目编号 | Char | 6 | 非必输 |  |
| ..Prj\_Nm | 项目名称 | VarChar2 | 60 | 非必输 |  |
| ..Pjcy\_Tp | 项目类型 | Char | 1 | 非必输 | 1-加项；2-减项 |
| ..Amt | 金额 | NUMBER | 19,8 | 非必输 |  |
| Parlist |  |  |  |  | 参与方列表 |
| .. Seq\_No | 顺序号 | NUMBER | 10 | 必输 | 参与方顺序号（默认从1开始） |
| ..Mkt\_Mrch\_Id | 商家编号 | VarChar2 | 30 | 必输 | 商家编号 |
| Sign\_Inf | 签名信息 | VarChar2 | 1500 | 必输 | 将上述参数进行拼接并用私钥生成签名 |

#### ****输出接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Cmdty\_Ordr\_No | 商品订单号 | VarChar2 | 40 | 必输 |  |
| Py\_Ordr\_No | 支付订单号 | VarChar2 | 40 | 必输 | 建行生成 |
| Ordr\_Gen\_Tm | 订单生成时间 | CHAR | 14 | 必输 | 生成订单时的时间 |
| Ordr\_Ovtm\_Tm | 订单超时时间 | CHAR | 14 | 必输 | 支付订单超时时间（订单生成后30分内未支付则订单超时） |
| Cshdk\_Url | 收银台URL | VarChar2 | 300 | 必输 | URL中存放商品订单号、支付订单号信息，数动态URL |
| Ordr\_Stcd | 订单状态代码 | VarChar2 | 2 | 必输 | 1待支付 |
| Sign\_Inf | 签名信息 | VarChar2 | 1500 | 必输 | 将上述参数进行拼接并用私钥生成签名 |

### 查询支付结果接口

#### ****访问路径****

[http:// marketpay.ccb.com /online/direct/enquireorder](http://marketpay.dev.jh/online/direct/enquireorder)

#### ****输入接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Ittparty\_Stm\_Id | 发起渠道编号 | VarChar2 | 5 | 必输 | 默认送5个0 |
| Py\_Chnl\_Cd | 支付渠道代码 | VarChar2 | 25 | 必输 | 默认送25个0 |
| Ittparty\_Tms | 发起方时间戳 | VarChar2 | 23 | 必输 | yyyymmddhhmmssfff  年月日, 时分秒，毫秒 |
| Ittparty\_Jrnl\_No | 发起方流水号 | VarChar2 | 32 | 必输 | 该笔直连交易的客户方流水号（不允许重复） |
| Mkt\_Id | 市场编号 | VarChar2 | 30 | 必输 |  |
| Cmdty\_Ordr\_No | 商品订单号 | VarChar2 | 40 | 非必输 | 商品订单号与支付订单号必输其一（当下单超时或异常时，可以通过商品订单查询） |
| Py\_Ordr\_No | 支付订单号 | VarChar2 | 40 | 非必输 | 商品订单号与支付订单号必输其一 |
| Vno | 版本号 | NUMBER | 9 | 非必输 | 默认版本为1 |
| Sign\_Inf | 签名信息 | VarChar2 | 1500 | 必输 | 将上述参数进行拼接并用私钥生成签名 |

#### ****输出接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Cmdty\_Ordr\_No | 商品订单号 | VarChar2 | 40 | 必输 |  |
| Py\_Ordr\_No | 支付订单号 | VarChar2 | 40 | 必输 | 建行生成 |
| Ordr\_Gen\_Tm | 订单生成时间 | CHAR | 14 | 必输 | 生成订单时的时间 |
| Ordr\_Ovtm\_Tm | 订单超时时间 | CHAR | 14 | 必输 | 支付订单超时时间。 |
| Ordr\_Stcd | 订单状态代码 | VarChar2 | 2 | 必输 | 1待支付 2成功 3失败  4全部退款 5部分退款 6失效 （一直未支付，超时过期，作为失效订单） |
| Sign\_Inf | 签名信息 | VarChar2 | 1500 | 必输 | 将上述参数进行拼接并用私钥生成签名 |

### 结果通知接口

订单成为终态后，对公专业结算综合服务平台主动通知客户。在未收到客户应答的前提下，会多次发送结果通知，发送频率为每5分钟发送一次，一共发送6次。**客户需要提前配置接收通知的URL并提供给结算平台。**可以参考附件里TestPay工程中的Tongzhi.java，TestRcv.java。

#### ****输入接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Cmdty\_Ordr\_No | 商品订单号 | VarChar2 | 40 | 必输 |  |
| Py\_Ordr\_No | 支付订单号 | VarChar2 | 40 | 必输 |  |
| Ordr\_Stcd | 订单状态代码 | VarChar2 | 2 | 必输 | 1待支付 2成功 3失败  4全部退款 5部分退款 6失效 （一直未支付，超时过期，作为失效订单） |
| Sign\_Inf | 签名信息 | VarChar2 | 1500 | 必输 | 将上述参数进行拼接并用私钥生成签名 |

#### ****输出接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Svc\_Rsp\_St | 服务响应状态 | VarChar2 | 2 | 必输 | 00-成功  01-失败 |
| Svc\_Rsp\_Cd | 服务响应码 | VarChar2 | 12 | 非必输 |  |
| Rsp\_Inf | 响应信息 | VarChar2 | 600 | 非必输 |  |
| Rcv\_Tm | 接收时间 | CHAR | 14 | 非必输 |  |

### 支付订单退款接口

客户可以通过此接口发起单笔订单的退款（全额退款和部分退款）。输入接口和输出接口中的参数都需要以json串的形式进行传递，同时输入接口和输出接口中的参数都需要按照key=value&key=value的形式拼接字符串进行签名和验签，拼接字符串的字段顺序即为接口字段定义的顺序。

针对当日生成的订单，退款是实时到账的。客户可以根据输出接口中的退款响应状态码Refund\_Rsp\_St和退款响应信息Refund\_Rsp\_Inf获得当日退款的结果和原因描述。针对隔日退款的订单，退款到账时间是发起退款请求的次日，所以需要在退款到账当日将退款结果主动通知给客户，接口具体描述可以参照1.3.5退款结果通知接口

#### ****访问路径****

http://marketpay.dev.jh/online/direct/refund

#### ****输入接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Ittparty\_Stm\_Id | 发起渠道编号 | VarChar2 | 5 | 必输 | 默认送5个0 |
| Py\_Chnl\_Cd | 支付渠道代码 | VarChar2 | 25 | 必输 | 默认送25个0 |
| Ittparty\_Tms | 发起方时间戳 | VarChar2 | 23 | 必输 | yyyymmddhhmmssfff  年月日, 时分秒，毫秒 |
| Ittparty\_Jrnl\_No | 发起方流水号 | VarChar2 | 32 | 必输 | 该笔直连交易的客户方流水号（不允许重复） |
| Mkt\_Id | 市场编号 | VarChar2 | 30 | 必输 |  |
| Py\_Ordr\_No | 支付订单号 | VarChar2 | 40 | 必输 | 下单时建行生成的支付订单编号 |
| Rfnd\_Amt | 退款金额 | NUMBER | 19,2 | 非必输 | 暂不支持部分退款 |
| Vno | 版本号 | NUMBER | 9 | 非必输 | 默认版本为1 |
| Sign\_Inf | 签名信息 | VarChar2 | 1500 | 必输 | 将上述参数进行拼接并用私钥生成签名 |

#### ****输出接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Refund\_Rsp\_St | 退款响应状态 | VarChar2 | 2 | 必输 | 00-退款成功（当日退款时会实时返回退款结果）  01-退款失败  02-退款中（隔日退款请求已受理，正在处理中） |
| Refund\_Rsp\_Inf | 退款响应信息 | VarChar2 | 600 | 非必输 | 退款处理结果描述 |
| Sign\_Inf | 签名信息 | VarChar2 | 1500 | 必输 | 将上述参数进行拼接并用私钥生成签名 |

### 退款结果通知接口

平台在接受到隔日退款请求后，会进行退款请求的处理，退款金额会在退款请求的次日到账。在退款请求次日处理完成之后，平台会主动通知客户退款到账结果。在未收到客户应答的前提下，会多次发送结果通知，发送频率为每5分钟发送一次，一共发送6次。

#### ****输入接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Py\_Ordr\_No | 支付订单号 | VarChar2 | 40 | 必输 |  |
| Refund\_Rsp\_St | 退款响应状态 | VarChar2 | 2 | 必输 | 00-退款成功  01-退款失败 |
| Refund\_Rsp\_Inf | 退款响应信息 | VarChar2 | 600 | 非必输 | 退款处理结果描述 |
| Sign\_Inf | 签名信息 | VarChar2 | 1500 | 必输 | 将上述参数进行拼接并用私钥生成签名 |

#### ****输出接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Svc\_Rsp\_St | 服务响应状态 | VarChar2 | 2 | 必输 | 00-成功  01-失败 |
| Svc\_Rsp\_Cd | 服务响应码 | VarChar2 | 12 | 非必输 |  |
| Rsp\_Inf | 响应信息 | VarChar2 | 600 | 非必输 |  |
| Rcv\_Tm | 接收时间 | CHAR | 14 | 非必输 |  |

### 通知到货接口

客户通过此接口告知到货成功，输入接口和输出接口中的参数都需要以json串的形式进行传递，同时输入接口和输出接口中的参数都需要按照key=value&key=value的形式拼接字符串进行签名和验签，拼接字符串的字段顺序即为接口字段定义的顺序。返回给客户相应的分账提示信息。

#### ****访问路径****

http:// marketpay.ccb.com/online/direct/noticeArrival

#### ****输入接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Ittparty\_Stm\_Id | 发起渠道编号 | VarChar2 | 5 | 必输 | 默认送5个0 |
| Py\_Chnl\_Cd | 支付渠道代码 | VarChar2 | 25 | 必输 | 默认送25个0 |
| Ittparty\_Tms | 发起方时间戳 | VarChar2 | 23 | 必输 | yyyymmddhhmmssfff  年月日, 时分秒，毫秒 |
| Ittparty\_Jrnl\_No | 发起方流水号 | VarChar2 | 32 | 必输 | 该笔直连交易的客户方流水号（不允许重复） |
| Mkt\_Id | 市场编号 | VarChar2 | 30 | 必输 |  |
| Py\_Ordr\_No | 支付订单号 | VarChar2 | 40 | 必输 | 下单时建行生成的支付订单编号 |
| **Vno** | 版本号 | NUMBER | 9 | 非必输 | 默认版本为1 |
| Sign\_Inf | 签名信息 | VarChar2 | 1500 | 必输 | 将上述参数进行拼接并用私钥生成签名 |

#### ****输出接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| Svc\_Rsp\_St | 服务响应状态 | VarChar2 | 2 | 必输 | 00-成功  01-失败 |
| Svc\_Rsp\_Cd | 服务响应码 | VarChar2 | 12 | 非必输 |  |
| Rsp\_Inf | 响应信息 | VarChar2 | 600 | 非必输 |  |

### 对账单推送接口

结算平台日终分账结束后向客户推送分账对账单文件。推送方式为HTTP请求方式，**客户需要提前配置接收通知的URL并提供给结算平台。**

#### ****输入接口****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **接口字段** | **名称** | **数据类型** | **数据长度** | **是否必输** | **备注** |
| File\_Smry\_Inf | 分账汇总文件摘要信息 | VarChar2 | 60 | 必输 |  |
| Sign\_Inf | 签名信息 | VarChar2 | 1500 | 必输 | 将上述参数进行拼接并用私钥生成签名 |

#### ****文件模板****

文件推送通过HttpClient 以Multipart方式发送,参考附件里TestPay工程中的FileUpLoadService.java文件里提供了推送样例代码以及接收样例代码。

接收样例代码请参考FileUpLoadService.java 文件里注释的部分。

解压缩后分为以下四个文件：



分账汇总文件命名：市场编号.分账日期.sum.txt

分账明细文件命名：市场编号.分账日期.det.txt

减项明细文件命名：市场编号.分账日期.sub.txt

对账明细文件命名：市场编号.日期.chk.txt

## 测试环境调用方式说明

### 测试环境域名

<https://marketpaytest.mytunnel.site/>

如“生成支付订单接口”的调用路径应变更为：

https://marketpaytest.mytunnel.site/ /online/direct/placeorder

### 测试环境公私钥对

市场方公私钥对：用私钥签名

公钥:04083E2CA7E71E51DB5374A49A3C07066390BD18C53B12A939D54C33E39E6916386F448B81D003BF76155EFEA565CD9818F6B84E846CB57CD4364BC715766D4FEC

私钥:20A6B71FEE52B30BD8669911BA1EEA9B530E24D0CFADF4694CBE10B358A884B6

平台方公私钥对：市场方用公钥验签

公钥:040F2C33AFE217CBC3FC4917BC948ACED230F52154C847DB5A8CABE36416F65C2F8DC395A4478B28D93DCB8D7D46143F98EF9B94C9B6A793516ABE4005C3089912

私钥:00D13AF485145540B12D72D2BD0850D7BD474DB0D14E56FDE10330C5EDAD3490E5

## 生产环境调用说明

### 生产环境域名

<http://marketpay.ccb.com>

如“生成支付订单接口”的调用路径应变更为：

http:// marketpay.ccb.com /online/direct/placeorder

### 需市场方提供信息

#### 市场方公钥

平台使用该公钥对市场方的签名进行验证

#### 市场方结果通知回调地址

具体请参见“结构通知接口”章节