

# 证海支付收银台接口文档 V1.0

## 1. 版权声明

本文档的版权属于证海支付, 任何人或组织未经许可, 不得擅自修改、拷贝或 以其它方式使用本文档中的内容。

## 2. 引言

### 名词定义

**商户机构:** 提供商品或服务的商业机构, 是电子支付中的收款方。 包含电子商务网站、B2B 电子交易市场、B2C 电子交易市场、公共事业单位、政府部门等, 是支付服务的需求方。

**用户:** 除机构外的电子支付交易的参与方, 包含企业用户和个人用户。

**电子交易市场:** 为交易方提供撮合服务的中间人, 它也是交易规则的制定者, 它在交易的过程中起到桥梁衔接的作用。电子交易市场也简称: 交易市场。

**非确认支付:** 在电子商务支付交易过程中, 付款方首先把资金支付到支付平台, 支付平台按照交易规则和协议, 再将资金结算给收款方。在这种模式下, 付款方支付成功后, 即完成支付过程。

### 编写目的

该文档旨在描述证海支付支付平台的交易接口说明, 以帮助商户机构顺利接入, 同时也作为软件开发人员的参考文档。

### 文档说明

电子商务支付中, 机构连接用户和支付平台, 在和支付平台进行连接时, 由支付平台提供数据加密传输和身份验证的接口程序。

支付平台的接口系统接收机构送来的订单信息后进行处理支付交易处理完成后将结果返回给机构交易系统。同时为机构提供订单查询、结算、下载对账单等功能。

本操作手册用来指导机构技术人员来实现与支付平台系统的顺利连接请相关技术人员详细阅读本手册。

### 数据安全责任划分

用户端, 用户到机构网络传输, 机构端的数据安全由机构完全负责。

机构到支付平台网络传输,机构平台端的数据安全由支付平台完全负责。

### 3.消息结构

HTTP 标准的 POST 协议。编码统一采用 UTF-8 编码。

基本数据元说明:都是 UTF-8 编码的字符串。

### 4 签名机制

MD5 签名

签名例子详见 8。

### 5.接口描述

交易类型

机构与支付平台的交易分为 2 类:

#### 1、异步交易

支付通知交易属于异步交易。机构提交的订单支付成功后,支付平台会向机构发送通知 报文。机构收到通知报文后再进行后续的交易处理。

#### 2、同步交易

机构产生交易报文后,向支付平台发送交易请求,支付平台受理请求后,立即回复一个 响应。同步交易包括: 订单支付交易,订单支付查询交易,下载对账单等

### 6.接口定义

#### 6.1 收银台置单接口

PC 收银台: PC 收银台上集成了快捷支付,扫码支付, B2B/B2C 网银支付,适用于 PC 及专用设备等支付场景

移动收银台: 移动收银台上继承了一键(快捷)支付,微信和支付宝的 H5 支付,适用于手机站, APP 等支付场景

## 交易步骤

商户系统调用该接口,在平台服务后台生成平台交易订单，平台返回正确的支付交易会话标识后，商户凭借标识发起请求到收银台。该接口包含如下三个步骤：

1. 商户调用创建订单接口在平台创建交易单并返回订单 token。
2. 商户根据支付的模式生成收银台 URL。
3. 用收银台 URL 将用户跳转到收银台进行支付。

## 置单接口请求报文

备注:使用接口请求地址

字段名	是否必填	类型	说明
SignMethod	是	String(8)	签名方法，默认：MD5
Signature	是	String(32)	签名信息，MD5 后 32 位小写
TxCode	不可空	String(6)	交易编码：setupord
Version	是	String(8)	1.0
MerNo	是	String(8)	商户号
TxSN	是	String(24)	商户交易流水号(唯一)
Amount	是	String(15)	金额，单位:分
PdtName	是	String(24)	商品名称
Remark	否	String(32)	备注值可不传但如果传参数参与排序生成 Signature.
ReturnUrl	否	String(120)	支付成功页面回调地址 值可不传但如果传参数参与排序生成 Signature.
NotifyUrl	是	String(12)	后台异步通知 URL
ReqTime	是	String(14)	请求时间 格式:yyyyMMddHHmmss
TimeoutExpress	否	String(4)	订单有效时间，单位分钟，默认 24 小时,值可不传但如果传参数参与排序生成 Signature.

## 置单接口响应报文

字段名	是否必填	类型	说明
SignMethod	是	String(8)	签名方法，默认：MD5
Signature	是	String(32)	签名信息，MD5 后 32 位小
RspCod	是	String(6)	响应码
RspMsg	是	String(8)	响应消息
MerNo	是	String(8)	商户号
TxSN	是	String(24)	商户交易流水号(唯一)
Amount	是	String(15)	金额，单位：分
PdtName	是	String(24)	商品名称
Remark	否	String(32)	备注
ReturnUrl	否	String(120)	前台通知地址
NotifyUrl	是	String(120)	后台异步通知 URL
ReqTime	是	String(14)	请求时间 格式:yyyyMMddHHmmss
Status	是	Char(1)	交易状态 1：成功 3：失败
PlatTxSN	是	String(32)	证海支付流水号
Token	是	String(32)	用来以参数方式传给收银台来完成支付

## 收银台 URL 拼接说明

### 拼接参数

字段名	是否必	类型	说明
SignMethod	是	String(8)	签名方法，默认：MD5
Signature	是	String(32)	签名信息，MD5 后 32 位小写
Version	是	String(8)	1.0
MerNo	是	String(8)	商户号
Token	是	String(32)	订单 token
ExpireTime	否	String(14)	操作收银台的有效时间，到期后不可操作 格式:yyyyMMddHHmmss 不传：默认：请求支付链接 10 分钟后关闭
ProductId	否	String(8)	产品 ID:传入该参数后直接调用支付工具，不进入收银台。 值可不传但如果传参数参与排序生成 Signature.
DirectBankId	否	String(16)	但产品 ID 设置为 B2C 跳银行时必填，值参考银行编码。值可不传但如果传参数参与排序生成 Signature.

CardType	否	String(16)	限制交易的卡类型，对快捷有效 枚举值：DEBIT：借记卡 CREDIT：信用卡,值可不传但如果传参数参与排序生成 Signature.
UserId	是	String(64)	用户标识,用户在商户系统唯一标识
UserIdType	是	String(32)	用户标识类型: USEID:用户 ID PHONE:用户手机号 ID_CARD:用户身份证号
ExtInfo	否	String(128)	扩展参数值可不传但如果传参数参与排序 生成 Signature.

备注:使用收银台 URL 地址进行拼接

例如:

拼接后的 URL = 收银台 URL 地址? MerNo=商户号&Version=1.0&UserId=商户系统用户标识

&UserIdType=用户标识类型&Token=置单接口返回的 token&SignMethod=MD5&Signature=MD5  
签名

### ProductId 产品 ID 说明

商户自己定制收银台,用户选择支付后，直接跳转入支付工具，在不同的支付环境支持如下：

ProductId	支付说明	PC 网站	移动网站	手机 APP
0601	微信扫码	支持	支持	支持
0602	支付宝扫码	支持	支持	支持
0614	快捷 H5 支付	支持	支持	支持
0615	H5 银联在线	支持	支持	支持
0621	H5 微信 APP 支付	不支持	支持	支持
0622	H5 支付宝 APP 支付	不支持	支持	支持
0611	B2C 网银跳银行	支持	不支持	不支持
0612	B2C 网银跳收银台	支持	不支持	不支持

## 6.2 支付订单交易查询

备注:使用接口请求地址

## 请求报文

字段名	可否为	类型	说明
SignMethod	不可空	String(8)	签名方法, 默认值: MD5
Signature	不可空	String(32)	签名信息, MD5 后 32 位小写
Version	不可空	String(8)	1.0
TxCode	不可空	String(6)	交易编码: <a href="#">qrypayord</a>
MerNo	不可空	String(8)	商户号
TxSN	不可空	String(24)	商户交易流水号(唯一)

## 响应报文

字段名	是否可空	类型	说明
SignMethod	不可空	String(8)	签名方法, 默认值: MD5
Signature	不可空	String(32)	签名信息, MD5 后 32 位小写
RspCod	不可空	String(6)	响应码
RspMsg	不可空	String(8)	响应消息
MerNo	不可空	String(8)	商户号
TxSN	不可空	String(24)	商户交易流水号(唯一)
Amount	不可空	String(15)	金额, 单位: 分
PdtName	不可空	String(14)	商品名称
Remark	可空	String(32)	备注
Status	不可空	Char(1)	交易状态 0: 未处理 1: 成功 2: 处理中 3: 失败
PlatTxSN	不可空	String(24)	证海支付流水号
TxTime	不可空	String(14)	交易时间 格式: yyyyMMddHHmmss
BankAccountNumber	可空	String(20)	银行账户号码
PayFee	可空	String(15)	手续费, 单位: 分

## 6.3 异步通知

支付平台发起异步通知到商户(机构). 商户(机构)接收并处理该请求。

处理 完成后向支付平台返回” success” 。支付平台收到该信息后, 做为

商户(机构) 收到异步通知的凭证。如果支付平台未收到该信息,以及网络问题引起的通知失败,支付平台会按照 5 分钟, 15 分钟, 30 分钟, 60 分钟的间隔再次发起异步通知。如果 60 分钟发送的异步通知未收到确认信息,不会再次发送。(共发送 5 次)

响应报文

字段名	可否为空	类型	说明
SignMethod	不可空	String(8)	签名方法,默认值: MD5
Signature	不可空	String(32)	签名信息,MD5 后 32 位小写
Version	不可空	String(8)	1.0
MerNo	不可空	String(8)	商户号
TxSN	不可空	String(24)	商户交易流水号(唯一)
Amount	不可空	String(15)	金额,单位: 分
PdtName	不可空	String(24)	商品名称
Remark	不可空	String(32)	备注
Status	不可空	Char(1)	交易状态 0: 未处理 1: 成功 2: 处理中 3: 失败
PlatTxSN	不可空	String(24)	证海支付流水号
TxTime	不可空	String(14)	交易时间 格式:yyyyMMddHHmmss
BankTxSN	可空	String(64)	渠道侧流水号
BankAccountNumber	可空	String(32)	银行卡号码
PayFee	可空	String(15)	手续费,单位: 分
PlatTxMsg	不可空	String(16)	交易描述

## 7. 附录

### 7.1 测试商户以及地址

名称	值
商户号	11570058
商户密钥	a164972347dd49d3012297f00fca0ec8
接口请求地址	<a href="http://apipay.58ccpsy.com/mertrade/api">http://apipay.58ccpsy.com/mertrade/api</a>
收银台 URL 地址	<a href="http://apipay.58ccpsy.com/payshow/index">http://apipay.58ccpsy.com/payshow/index</a>

### 7.2 B2C 银行编码列表

银行编码	银行名称
ICBC	工商银行
ABC	农业银行
BOC	中国银行
CCB	建设银行
BOCOM	交通银行
CITIC	中信银行
CEB	光大银行
HXB	华夏银行
CMBC	民生银行
GDB	广东发展银行
PAB	平安银行
CMB	招商银行
BOB	北京银行
SPDB	浦发银行
BOS	上海银行
CIB	兴业银行
PSBC	邮政储蓄银行

## 8. 报文签名详解

### 以跳转跳转收银台预下单为例

#### 1: 生成待签名字符串

在请求参数列表中，除去 SignMethod、Signature 两个参数外，其他需要使用到的参数皆是要签名的参数。



对请求参数数组里的每一个值从 a 到 z 的顺序排序,若遇到相同首字母,则看第二个字母,以此类推。排序完成之后,把所有的值直接连接起来。

如:

Amount=1000&MerNo=11570058&NotifyUrl=http://XXX.XXX.com/asyncNotifyProcess&PdtName=测试&ProductId=0612&Remark=测试&ReqTime=20180326194355&ReturnUrl=http://XXX.XXX.com/return&TxCode=210112&TxSN=20180326194355

## 2: MD5 签名

当拿到请求时的待签名字符串后,需要把密钥拼接到待签名字符串后面,形成新的字符串,利用 MD5 的签名函数对这个新的字符串进行签名运算,得到 MD5 的结果赋值于参数 Signature.

Signature =MD5 (待签名字符串+密钥)

待签名字符串+密钥 例如:

Amount=1000&MerNo=11570058&NotifyUrl=http://XXX.XXX.com/asyncNotifyProcess&PdtName=测试&ProductId=0612&Remark=测试&ReqTime=20180326194355&ReturnUrl=http://XXX.XXX.com/return&TxCode=210112&TxSN=20180326194355e084f8d4f53180b43c2ac522ec9fe8cb

红色部分是密钥。

Signature 为 e084f8d4f53180b43c2ac522ec9fe8cb

## 3:对请求参数值处理

SignUtil.getWebForm 函数就是进行这个处理的。

3.1 对参数名为: { "CodeUrl", "ImgUrl", "Token\_Id", "PayInfo", "sPayUrl", "PayUrl", "NotifyUrl", "ReturnUrl" }的值先进行 Base64 编码然后把其中的” +” 替换为 “%2b”

Java 函数:

```
/**
 * 转换特殊请求字段
 *
 * @param paramMap
 */
public static void convertReqData(Map<String, String> paramMap, String[]
base64Keys, String charset) {
    if (base64Keys == null)
        return;
    for (int i = 0; i < base64Keys.length; i++) {
        String key = base64Keys[i];
        String value = (String) paramMap.get(key);
        if (value != null) {
            try {
                String text =
Base64.encode(value.getBytes(charset)).replace("+", "%2b");
                // 更新请求参数
                paramMap.put(key, text);
            } catch (Exception e) {
                // 处理异常
            }
        }
    }
}
```

```

        } catch (Exception e) {
        }
    }
}
}

```

### 3.2 对所有参数值调用 `URLEncoder.encode` 进行 utf-8 编码

`URLEncoder.encode`

请求的原始报文:

请求原始报文:

Amount=1000&MerNo=11570058&NotifyUrl=http://XXX.XXX.com/asyncNotifyProcess&PdtName=测试&ProductId=0612&Remark=测试&ReqTime=20180326194355&ReturnUrl=http://XXX.XXX.com/return&SignMethod=MD5&Signature=3d15b0f45237f3e3784c9bf10240cba8&TxCode=210112&TxSN=20180326194355

经过 3.1,3.2 处理后发送的实际报文:

TxCode=210112&ReqTime=20180326194355&PdtName=%E6%B5%8B%E8%AF%95&MerNo=11570058&Amount=1000&NotifyUrl=aHR0cDovL1hYWc5YWFGuY29tL2FzeW5jTm90aWZ5UHJvY2Vzcw%3D%3D&Signature=3d15b0f45237f3e3784c9bf10240cba8&TxSN=20180326194355&ProductId=0612&ReturnUrl=aHR0cDovL1hYWc5YWFGuY29tL3JldHVybg%3D%3D&SignMethod=MD5&Remark=%E6%B5%8B%E8%AF%95

## 4 接收数据后，对返回参数值进行处理

`SignUtil.parseResponse` 函数就是进行如下处理。

### 4.1 对所有参数值调用 `URLDecoder.decode` 进行 utf-8 解码

`URLDecoder.decode`

4.2 对参数名为: { "CodeUrl", "ImgUrl", "Token\_Id", "PayInfo", "sPayUrl", "PayUrl", "NotifyUrl", "ReturnUrl" } 的值先把其中的 "%2b" 替换为 "+" 然后 Base64 解码。

```

/**
 * 转换特殊返回字段
 *
 * @param paramMap
 */
public static void convertRespData(Map<String, String> paramMap, String[]
base64Keys) {
    if (base64Keys == null)
        return;
    for (int i = 0; i < base64Keys.length; i++) {
        String key = base64Keys[i];
        String value = (String) paramMap.get(key);
    }
}

```

```

        if (value != null) {
            try {
                String text = new String(Base64.decode(value.replace("%2b",
                    "+")));

                // 更新请求参数
                paramMap.put(key, text);
            } catch (Exception e) {
            }
        }
    }
}

```

返回的原始报文

Status=1&PlatTxSN=S880753905806846&RspMsg=%E4%BA%A4%E6%98%93%E6%88%90%E5%8A%9F&Amount=1000&PayChannel=&ProductId=0612&ReturnUrl=aHR0cDovL1hYWc5YWFGuY29tL3JldHVybG%3D%3D&SignMethod=MD5&Remark=%E6%B5%8B%E8%AF%95&ReqTime=20180326194355&PdtName=%E6%B5%8B%E8%AF%95&PayUrl=aHR0cDovL2FwaS41MXJsYi5jb20vdHJhZGVtdWJtaXQuaHRtbD9hY2Nlc3NNb2RlPTAxJmFjY2Vzc1R5cGU9MCZiYWNRVXJsPWWh0dHA6Ly9hcGlwYXkuNTFyY29tL2FzeW5jLzkyOTAzMdA0ODE2MDExMy50cnNwJmJpelR5cGU9MDAwMDAwJmJvZG9wOTI5dWMzVnRaUT09JmN1cnJlbnN5PUNOWSZtZXJJZD05MjkwMzAwNDgxNjAxMTMmbWVvT3JkZXJJZD1TODgwNzUzOTA1ODA2ODQ2JnBheVR5cGU9MDIwMSZzaWduTWV0aG9kPU1ENSZzaWduYXR1cmU9N1lnWVBJS1hvR1dqYUVCUKIvRkFuZz09JnN1YmplY3Q9U0dGeVpIZGhjbVFNWVdOalpYTnpiM0pwWlhNPSZ0eG5BbXQ9MTAwMCZ0eG5TdWJUeXB1PTAwJnR4blRpbWU9MjAxODAzMjYxOTQzNTUmdHhuVHlwZT0wMSZ2ZXJzaW9uPTEuMC4w&MerNo=11570058&NotifyUrl=aHR0cDovL1hYWc5YWFGuY29tL2FzeW5jTm90aWZ5UHJvY2Vzcw%3D%3D&Signature=**1991d7618a89c0429b1603eb29106c77**&TxSN=20180326194355&PayMethod=1&RspCod=00000

经过 4.1,4.2 处理后的返回报文:

Amount=1000&MerNo=11570058&NotifyUrl=http://XXX.XXX.com/asyncNotifyProcess&PayChannel=&PayMethod=1&PayUrl=http://api.51rlb.com/tradeSubmit.html?accessMode=01&accessType=0&backUrl=http://apipay.51rlb.com/async/929030048160113.trsp&bizType=000000&body=Y29uc3VtZQ==&currency=CNY&merId=929030048160113&merOrderId=S880753905806846&payType=0201&signMethod=MD5&signature=7YgYPIKXoGWjaEBRB/FAng==&subject=SGFyZhdhcmUgYWNjZXNzb3JpZXM=&txnAmt=1000&txnSubType=00&txnTime=20180326194355&txnType=01&version=1.0.0&PdtName=测试&PlatTxSN=S880753905806846&ProductId=0612&Remark=测试&ReqTime=20180326194355&ReturnUrl=http://XXX.XXX.com/return&RspCod=00000&RspMsg=交易成功&SignMethod=MD5&Signature=**1991d7618a89c0429b1603eb29106c77**&Status=1&TxSN=20180326194355

## 5: 验证签名

5.1 计算签名:用 4 处理后的返回参数通过 1,2 步骤计算出签名

待签名字符串+密钥

Amount=1000&MerNo=11570058&NotifyUrl=http://XXX.XXX.com/asyncNotifyProcess&PayChannel=&PayMethod=1&PayUrl=http://api.51rlb.com/tradeSubmit.html?accessMode=01&accessType=0&backUrl=http://apipay.51rlb.com/async/929030048160113.trsp&bizType=000000&body=Y29uc3VtZQ==&currency=CNY&merId=929030048160113&merOrderId=S880753905806846&payType=0201&signMethod=MD5&signature=7YgYPIKXoGWjaEBRB/FAng==&subject=SGFyZHdhcmUgYWVjZXNzb3JpZXM=&txnAmt=1000&txnSubType=00&txnTime=20180326194355&txnType=01&version=1.0.0&PdtName=测试&PlatTxSN=S880753905806846&ProductId=0612&Remark=测试&ReqTime=20180326194355&ReturnUrl=http://XXX.XXX.com/return&RspCod=000000&RspMsg=交易成功&Status=1&TxSN=20180326194355**1991d7618a89c0429b1603eb29106c77**

计算出的签名为:**1991d7618a89c0429b1603eb29106c77**

5.2 比较签名:计算出的签名和返回的签名比较是否一致。