

银联支付控件接入指南

3.1

中国银联

2024-10-21

目录

版本信息2

1 概述1

2 支付流程介绍1

3 测试帐号2

4 鸿蒙（HarmonyOS）客户端2

 4.1 SDK 包说明.....3

 4.2 接口说明.....3

 4.3 添加 SDK 包.....7

 4.4 工程配置.....7

 4.5 调用插件.....9

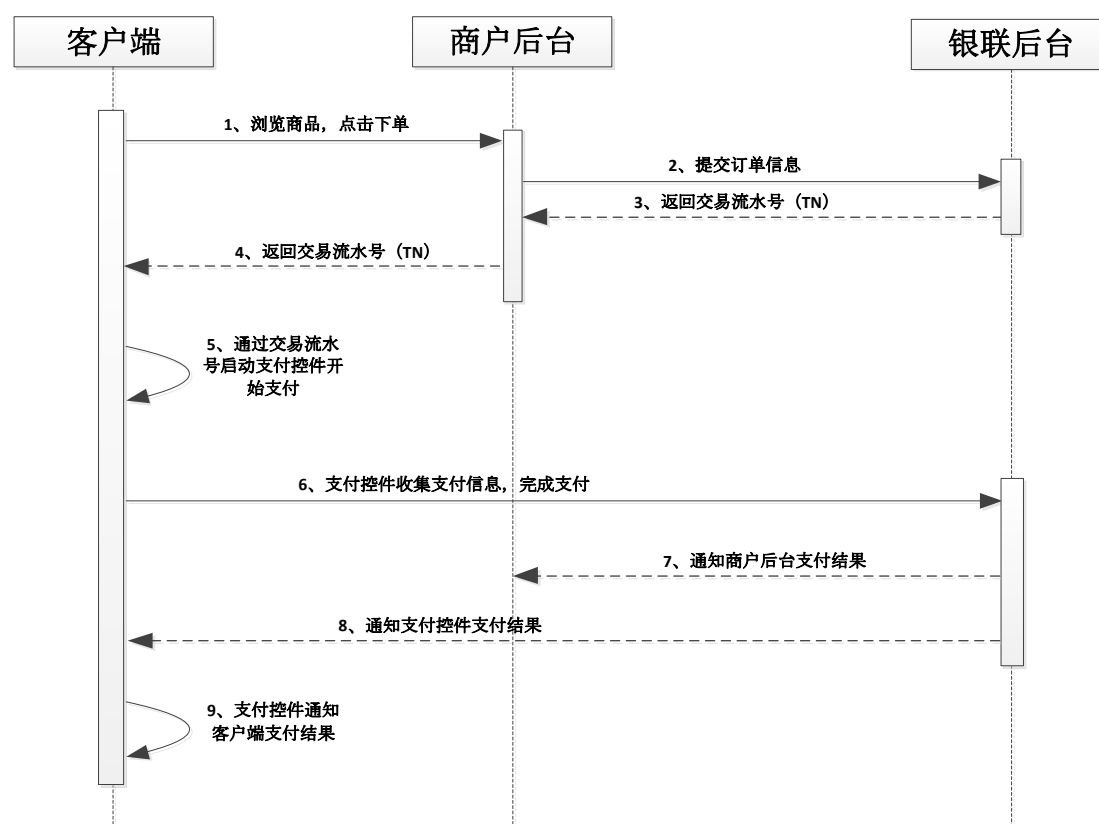
5 常见问题12

1 概述

银联线上收银台 SDK 支持调用云闪付 APP、银联手机闪付、银行 APP、Wap 支付等多种支付方式，可以为用户提供便捷多样的支付体验。

2 支付流程介绍

通过银联线上收银台 SDK 进行的交易流程如下图：



流程图说明：

- (1) 用户在商户 APP 中点击购买商品，商户 APP 客户端发起订单生成请求到商户后台；
- (2) 商户后台收到订单生成请求后，按照《05-中国银联全渠道支付平台-第 2 部分 产品接口规范-银联线上收银台》组织并推送订单信息至银联后台；

- (3) 银联后台接收订单信息并检查通过后，生成对应交易流水号（TN），并返回交易流水号（TN）至商户后台（应答要素：交易流水号等）；
- (4) 商户后台接收到交易流水号（TN），将交易流水号（TN）返回给商户 APP 客户端；
- (5) 商户 APP 客户端通过交易流水号（TN）调用线上收银台 SDK；
- (6) 线上收银台 SDK 调起银联线上收银台页面，用户选择具体支付方式后，跳转至该支付方式进行支付；
- (7) 支付成功后，银联后台将支付结果通知给商户后台；
- (8) 银联将支付结果通知这笔交易对应的支付工具；
- (9) 支付工具显示支付完成页并将支付结果通过线上收银台 SDK 返回给商户 APP 客户端；

注：本文档主要关注上述流程中（5）、（9）部分的实现

3 测试帐号

提供测试使用卡号、手机号信息（**此类信息仅供测试，不会发生正式交易**）

测试卡信息见此 FAQ：<https://open.unionpay.com/tjweb/support/faq/mchlist?id=4>（如地址发生变动，进主页找到商户常见问题页面，搜“测试卡”）

4 鸿蒙（HarmonyOS）客户端

本小节涉及到 SDK 的接口说明、及其接口调用细节，需要读者具有一定鸿蒙（Ha

rmonyOS) 编程经验。

4.1 SDK 包说明

商户开发者在获取到银联提供的开发包后请检查 SDK 文件所在目录 paycontrol_merchant_oohos/Products，以下部分提及的文件均在该目录中。银联支付控件为静态共享包 (HAR)，以下简称 libuppayment，包含以下文件：

- libuppayment.har

4.2 接口说明

4.2.1 支付接口

```
startPay(tn: string, mode: string, scheme: string,  
         context: common.UIAbilityContext, seType?: string):  
void
```

各个参数定义如下表：

参数名称	类型	含义
tn	string	必填项； 交易流水号，商户后台向银联后台提交订单信息后，由银联后台生成并下发给商户后台的交易凭证；
mode	string	必填项； 接入模式，标识商户以何种方

		式调用支付控件,该参数提供 以下两个可选值: "00"代表接入生产环境(正式 版本需要); "01"代表接入开发测试环境 (测试版本需要);
scheme	string	必填项; 商户自定义协议,商户在调用 支付接口完成支付后,用于引 导支付控件返回而定义的协 议
context	common.UIAbilityContex t	必填项; 用户展示支付页面
seType	string	可选项; 手机 pay 支付类别,见表 1

手机品牌与seType对应关系表格如下：

表1 手机品牌与seType对应表格

手机品牌	seType
Huawei Pay	04

4.2.2 返回结果接口

```
handlePaymentResult(url: string, resultCallback: UPPayResultCallback);
```

各参数定义如下表：

参数名称	类型	含义
url	NSURL*	必填项； 支付结果 url，传入后由 SDK 解析 并通过 resultCallback 回调商户客 户端； url 格式为：xxxx://uppayresult?code=success&data= 其中 xxxx 为商户自定义 scheme， uppayresult 为银联结果标识串， 商户可依据是否包含 uppayresult 用来判断是否为银联支付结果调用
resultCallback	UPPayResultCallback	必填项； 结果回调函数，商户客户端定义的结果处理方法，此方法本文档 4.5 第三部分将详细说明；

UPPayResultCallback 参数说明

```
UPPayResultCallback = (result: UPPayResult) => void
```

参数名称	类型	含义
result	UPPayResult	必填项； 支付结果

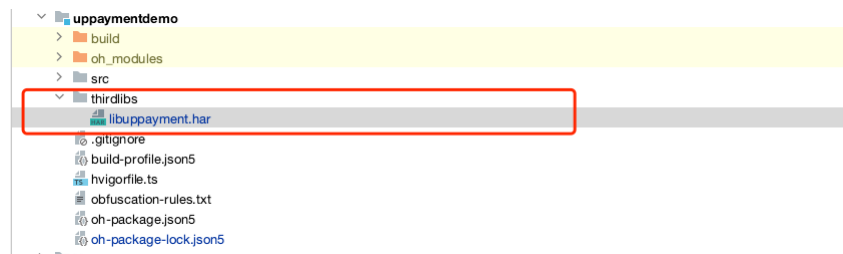
UPPayResult 参数说明

参数名称	类型	含义
code	UPPayResultStatus	必填项； 支付状态码，枚举值： SUCCESS：支付成功 FAIL：支付失败 CANCEL：支付取消
data	string null	可选项； 支付数据，预留字段

4.3 添加 SDK 包

使用 libuppayment 需要将 libuppayment.har 文件添加到商户应用的工程中，添加后

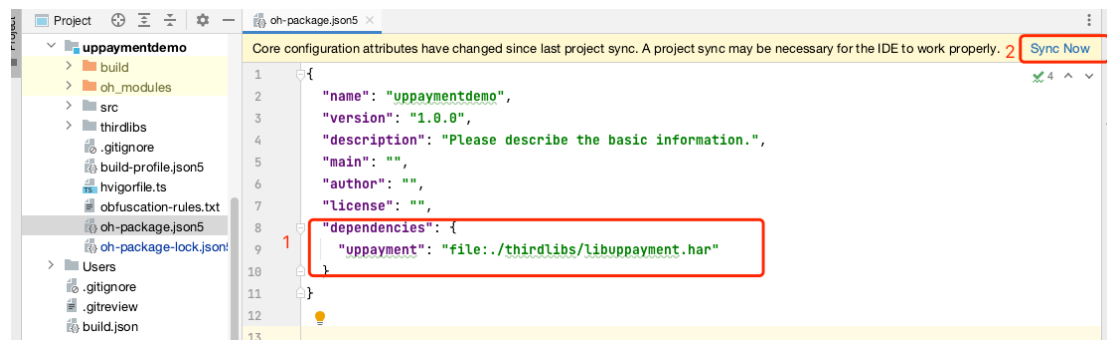
如图：



4.4 工程配置

1、 添加 libuppayment 后，在工程 oh-package.json5 文件中添加依赖，如下图红

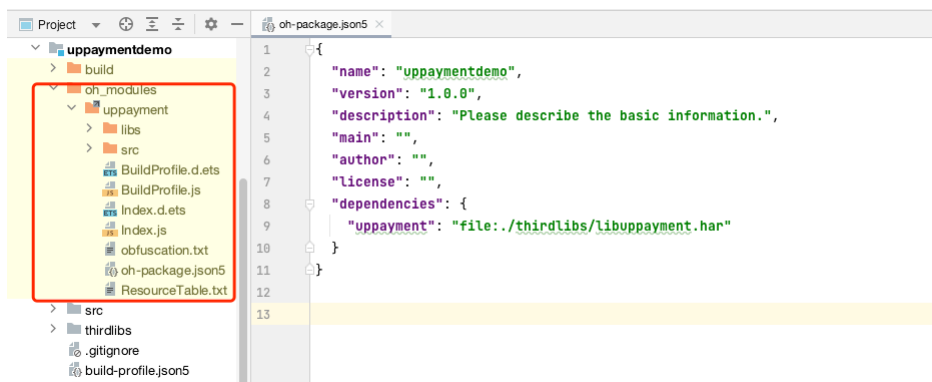
框 1 所示：



2、 在 oh-package.json5 依赖后，点击 Sync Now 按钮，如上图红框 2 所示。点击

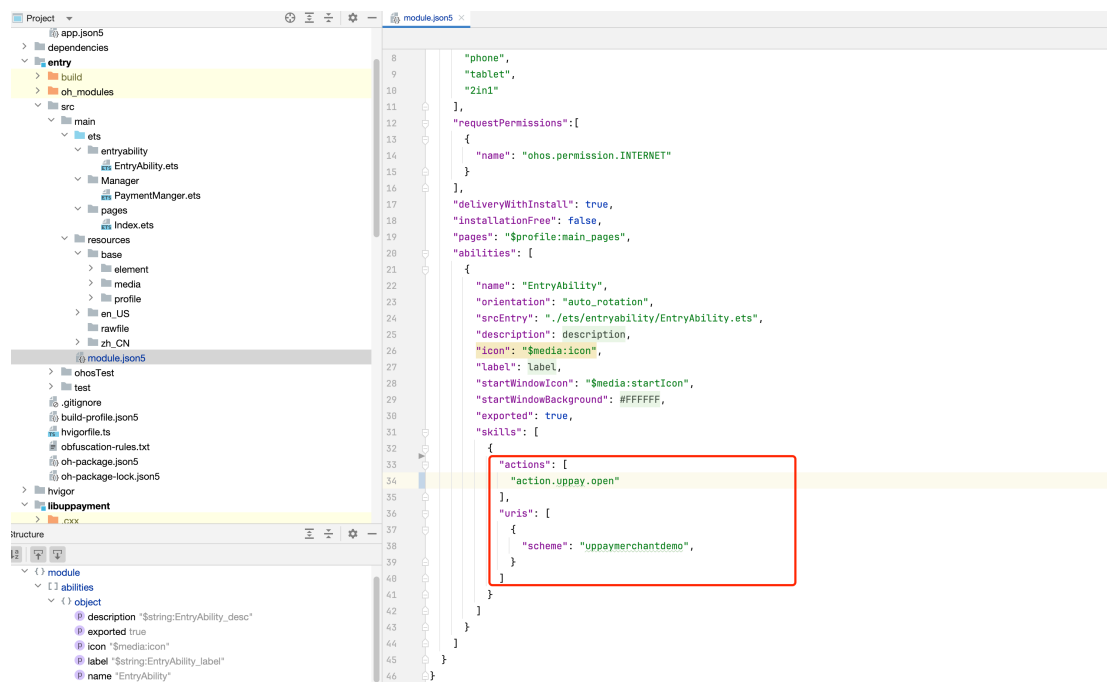
完成后，IDE 完成自动完成依赖配置，IDE 左侧工程结构窗口中会在 oh_module

s 中新增 uppayment 目录，如下图所示：



3、module.json5 配置

在 module.json5 中的 skills 下添加模块的 uris 和 actions，如下图所示：



注意：uris 中的 scheme 名称由商户自定义。actions 请统一添加为：

action.uppay.open。

4.5 调用插件

在需要调用支付控件接口的代码文件内引入 libuppayment。

1、支付接口调用

商户 App 从商户服务器获取 tn，当 tn 不为空时，调用支付接口。

```
import {UPPayment, UPPayResult, UPPayResultCallback} from 'uppayme
nt
...
...
...
let context = getContext() as common.UIAbilityContext;
let mode = '01'; //01: 银联测试环境, 该环境中不发生真实交易;00: 银联正式环境
let upPayment = new UPPayment();

this.upPayment.startPay(tn, mode, 'uppaymerchantedemo', context, '');
```

2、返回结果接口调用

支付完成后，获取支付控件支付结果，并添加相应处理逻辑，需在 Ability 中的 onNewWant 和 onCreate 方法中调用 `handlePaymentResult`，支付成功完成后返回商户客户端支付结果。

控件返回的结果信息仅作为参考，商户订单是否成功支付应该以商户后台主动到全渠道查询的结果或者收到全渠道支付结果通知为准。

收到控件返回结果后，建议直接到自己的后台查状态，如果后台此时未收到全渠道的后台通知，则主动发起查询接口到全渠道查状态。这样可既保证APP中展示的订单状态和后台记录一致，也可及时更新后台记录的订单状态。

结果处理方法示例代码：

```
onNewWant(want: Want, launchParam: AbilityConstant.LaunchParam) {
  if (want?.uri && want.uri.includes('uppayresult') && this.uiContext) {
    this.upPaymen.handlePaymentResult(want.uri, (data: UPPayResult) => {
      switch (data.code) {
        case UPPayResultStatus.CANCEL:
          this.uiContext?.showAlertDialog({
            title: '支付结果',
            message: '取消!',
            alignment: DialogAlignment.Center
          });
          break;
        case UPPayResultStatus.FAIL:
          this.uiContext?.showAlertDialog({
            title: '支付结果',
            message: '失败!',
            alignment: DialogAlignment.Center
          });
          break;
        case UPPayResultStatus.SUCCESS:
          this.uiContext?.showAlertDialog({
            title: '支付结果',
            message: '成功!',
            alignment: DialogAlignment.Center
          });
          break;
        default:
          this.uiContext?.showAlertDialog({
            title: '支付结果',
            message: '取消!',
            alignment: DialogAlignment.Center
          });
          break;
      }
      console.log('支付结果: ' + JSON.stringify(data));
    });
  }
}
```

3、canOpenLink 白名单配置

在 module.json5 中的 querySchemes 下添加白名单，如下图所示：

```
{
  "module": {
    "name": "entry",
    "type": "entry",
    "description": module description,
    "mainElement": "EntryAbility",
    "deviceTypes": [...],
    "requestPermissions": [
      {"name": "ohos.permission.INTERNET"...}
    ],
    "deliveryWithInstall": true,
    "installationFree": false,
    "pages": "$profile:main_pages",
    "querySchemes": [
      "uppaywallet",
      "uppaysdk",
    ],
  },
}
```

注意：querySchemes 中只需要添加 scheme 名称，不需要添加 ://。

需添加的各支付工具 scheme 如下：

序号	支付 APP scheme	对应支付 APP
1	uppaysdk	云闪付 APP
2	uppaywallet	
3	uppayvendor	华为 Pay

5 常见问题

1、商户需按照以下链接内容配置 IP 及地址白名单

<https://open.unionpay.com/tjweb/support/faq/mchlist?id=228>

测试	生产
cn : https://appcashier.test.cup.com.cn	https://appcashier.cup.com.cn

2、其他问题请参见 <https://open.unionpay.com> 帮助中心-FAQ