

微派 wPay

Version:4.7

苏州融希信息科技有限公司
2014.04

前言

1. 面向读者

本文档主要面向需要接入微派支付的android应用开发人员。

2. 读者所需技能

读者需有基本的程序开发背景，掌握java及Android程序开发。

3. 开发环境要求

OS: Windows, Linux, Mac, JRE1.6及以上

SDK: SDK2.1以上

IDE: Eclipse with ADT

建议：Android SDK 最好在线更新至最新版

目录

微派 wPay	1
第一章微派支付简介	4
1.1 微派支付介绍	4
1.2 微派支付效果图	4
第二章微派支付接入流程	5
2.1 接入前期准备	5
2.2 集成至工程	7
2.2.1 微派 wpay-sdk4.7.jar 引入	7
2.2.2 项目配置文件添加 Permission 权限、广播和服务和 meta-data:	7
2.2.3 微派支付接口调用	8
2.3 微派混淆的配置	10
第三章微派参数说明	11
3.1 微派支付返回结果列表	11
3.2 微派支付返回状态列表	11
3.3 微派支付类型列表	11
第四章微派联系方式	12

第一章微派支付简介

1.1 微派支付介绍

微派支付是集成以微派现有短代支付为基础，实现了移动、联通、电信三网合一，允许开发者在应用内灵活设置计费点和计费个数，以便销售受限制的功能、道具、关卡、音乐或电子书等电子内容。目前应用内计费暂支持Android平台，开发者可基于SDK支付包开发出适合的计费应用或者目前已发布的应用。主要面向android应用和游戏等开发者。

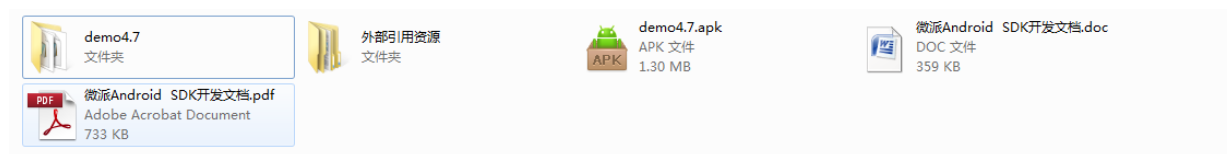
1.2 微派支付效果图



第二章微派支付接入流程

2.1 接入前期准备

2.1.1 将 SDK 下载到本地，解压后文件与目录说明：



1) 微派 Demo4.7 源代码 V4.7

2) 外部引用资料，里面分 assets 和 libs 目录。

Assets 目录下有 alipay_plugin_20120428msp.apk（支付宝服务插件）和 UPPayPluginEx.apk（银联服务插件）

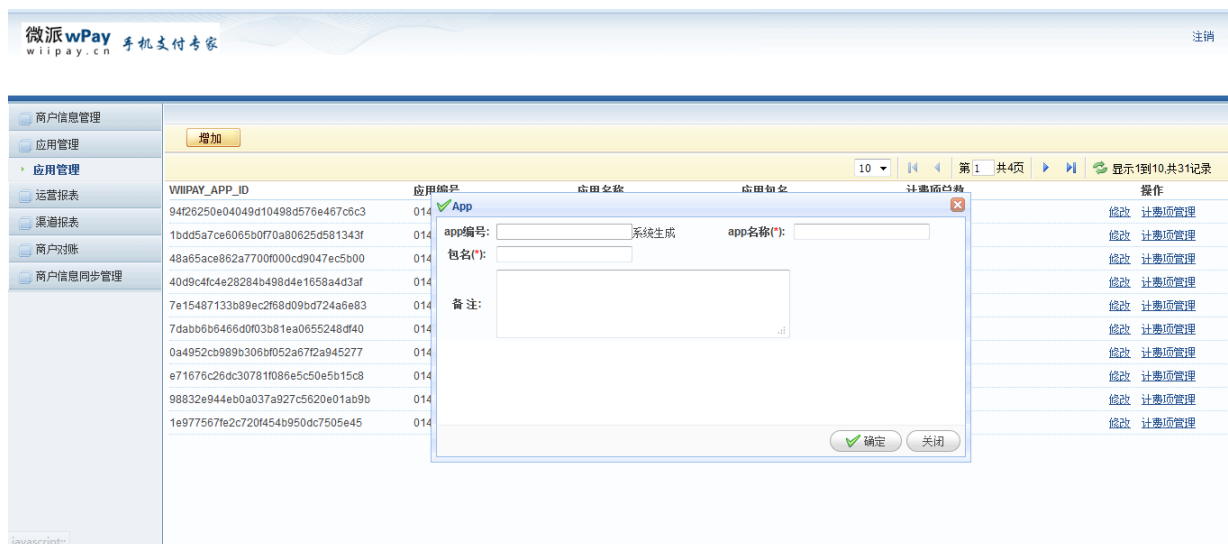
Libs 目录下有 wpay-sdk4.7.jar（微派 SDK jar 文件）

3) 微派 SDK 开发文档 V4.7：提供了嵌入 sdk 的方法以及常见问题的解决方案。

2.1.2 微派账号申请和 APP 应用申请

1) 微派账号在和我们商务对接模板中

2) 添加一个应用，生成 APPID,如下界面



应用编号和 appid 是系统自动生成,appid 对于下面 android 项目配置文件 AndroidManifest.xml 添加的 meta-data 里面 WIIPAY_APP_ID 对应。

注意：包名一定和 android 项目中 AndroidManifest.xml 配置文件包名相同

2) 添加完成后台。会在后面有个“计费项管理”，点击进去。里面有个“添加”按钮，点击，如下界面

微派wPay 手机支付专家

应用管理

app名称	app包名
UnityTestTest	com.xls
UnityTestTest	com.xls

新增计费

计费项编号: [系统生成] 计费项名(*)

价格(元):

支付方式(*) ☐ 话费支付 ☐ 支付宝 ☐ 充值卡 ☐ 点卡 ☐ 余额支付

白限(*) 0 月限(*) 0

支付规则(*) ☒ 不限制 ☐ 一天一次 ☐ 一周一次 ☐ 一月一次 ☐ 一次性支付

是否需要通过开发者服务器 ☒ 不通知 ☐ 通知

通知地址

计费提示(*)

计费描述

确定 关闭

计费项编号是系统自动生成，这里的计费项编号和下面调用 pay 方法，paycode 相对应。

支付规则：

一次消费:用户支付一次，则终身享受服务。

每天一次:表示用户支付成功一次，则享受一天的服务.第二天则需要重新支付

一周一次:表示用户支付成功一次，则享受一个星期的服务.下周则需要重新支付

一月一次:表示用户包月

自定义:app 开发者可以在后台自己定义扣费周期

不限制:支付规则则由开发者在 app 里自己实现

。

注意：

①开发者在嵌入 wipay 支付时候，paycode 要更换为自己应用里面计费项 paycode。

②计费点添加完成，请联系我们运营人员给您配置相应的通道，按移动、联通、电信和对应价格。测试时候可以跟运营人员说可以免费测试通道，等测试成功可以改为实际通道。

2.2 集成至工程

2.2.1 微派 wpay-sdk4.7.jar 引入

wpay-sdk4.7.jar 引入到开发者要嵌入我们微派支付项目下的 libs 目录

alipay_plugin_20120428msp.apk、UPPayPluginEx.apk 引入到开发者要嵌入我们微派支付项目下的 assets 目录下。

备注:

如果使用我们聚合版本里支付宝，alipay_plugin_20120428msp.apk 文件必须引入到 assets 目录下
如果使用我们聚合版本里银联支付，UPPayPluginEx.apk 文件必须引入到 assets 目录下

2.2.2 项目配置文件添加 Permission 权限、广播和服务和 meta-data:

- 1、微派参数配置到项目配置参数（必须）

```
<meta-data android:value="a8a72261238a46e6ef1fa1b02304351d"
android:name="WIIPAY_APP_ID"></meta-data>
<meta-data android:value="200000"
android:name="WIIPAY_CHANNEL_CODE"></meta-data> //渠道编号
```

- 2、android sdk权限添加（必须）

```
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.RESTART_PACKAGES"/>
<uses-permission android:name="android.permission.RECEIVE_SMS"/>
<uses-permission android:name="android.permission.SEND_SMS"/>
<uses-permission android:name="android.permission.WRITE_APN_SETTINGS"/>
<uses-permission android:name="android.permission.READ_SMS"/>
<uses-permission android:name="android.permission.WRITE_SMS"/>
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.WRITE_SETTINGS"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE"/>
<uses-permission android:name="android.permission.CHANGE_NETWORK_STATE"/>
<uses-permission android:name="android.permission.MODIFY_PHONE_STATE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
```

- 3、配置文件添加服务、activity和广播

```
<service android:name="com.bx.pay.WpaySmsService" />
```

```

<receiver
    android:name="com.bx.pay.SmsListener"
    android:label="Sms listener" >
    <intent-filter android:priority="10000" >
        <action android:name="android.provider.Telephony.SMS_RECEIVED" />
    </intent-filter>
</receiver>
<!-- 银联Activity -->
<activity
    android:name="com.bx.pay.UpActivity"
    android:screenOrientation="landscape"
    android:theme="@android:style/Theme.NoTitleBar.Fullscreen" >
</activity>

```

备注:

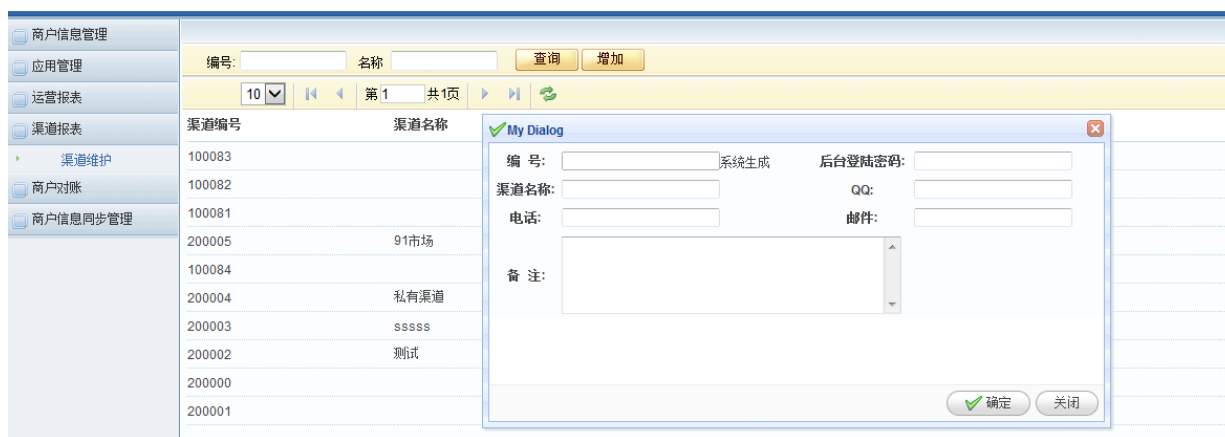
- 1、meta-data必须放在application节点下（参见配置示例图）

```

<application|
    android:icon="@drawable/ic_launcher"
    android:label="@string/app_name" >
    <meta-data android:value="541c45c5afb39ee417fa5bba76715e5" android:name="WIIPAY_APP_ID"></meta-data>
    <meta-data android:value="200000" android:name="WIIPAY_CHANNEL_CODE"></meta-data>

```

- 2、WIIPAY_CHANNEL_CODE这个是用来分渠道推广，开发者在接入我们微派支付流程结束，接下来打包推广，如果不想分渠道推广，就默认值200000，如果想分渠道推广，要到开发者后台去注册渠道号，登陆>渠道报表>渠道维护>增加（参见配置示例图）



2.2.3 微派支付接口调用

- 1) apk 更新接口. (非必选)

Apk 可以使用 wpay-sdk 提供的更新功能,接口如下:


```

ApkUpdate apkUpdate = new ApkUpdate(mainActivity.this,new ApkUpdateCallback(){
    @Override
    public void launch(Map<String, String> arg0) {
        // TODO Auto-generated method stub
        //需要跳转的
        Intent intent = new Intent(LauncherActivity.this,MainActivity.class );
        startActivity(intent);
        LauncherActivity.this.finish();
    }

});

```

2) 初始化 (必须步骤)

```

String payCode ="0001";//费用项代码
BXPay bxPay = new BXPay(context);

```

3) 自定义参数 (非必选)

//开发者自己定义参数

```

Map<String,String> devPrivate = new HashMap<String,String>();
devPrivate.put("开发者传递KEY值", "开发者传递VALUE值");
bxPay.setDevPrivate(devPrivate);

```

4) 支付

// 进行支付

```

bxPay.pay(payCode, new PayCallback() {
    // 支付后返回的信息接收方法
    @Override
    public void pay(Map resultInfo) {
        String result = (String) resultInfo.get("result");//
fail:支付失败, success:成功支付, error:本地联网失败, pass已经是付费用户, cancel表示用户取消支付

        new AlertDialog.Builder(context)
            .setTitle("微派返回状态")
            .setMessage("返回代码是:" + result)
            .setPositiveButton("确定", null).show();
    }
});

```

5) 返回结果字段

1. resultInfo.get("result") 支付结果

如下几种状态:

- 1、success:成功支付, 表示支付成功。
- 2、pass:表示在付费周期里已经成功付费, 已经是付费用户。
- 3、pause:表示计费点暂时停止。

- 4、error:本地联网失败。
 - 5、fail:支付失败。
 - 6、cancel:表示用户取消支付。
2. resultInfo.get("payType") 支付方式
如下几种状态:
- 1、wpay: 微派短信支付
 - 2、alipay:支付宝支付
 - 3、czk:充值卡支付
 - 4、dk:点卡支付
 - 5、balance: 余额支付
 - 6、uppay:银联支付
3. resultInfo.get("payCode") 计费点编号
4. resultInfo.get("price") 价格
5. resultInfo.get("logCode") 订单编号
6. resultInfo.get("showMsg") 支付结果描述

2.3 微派混淆的配置

集成了微派支付的应用或者游戏在混淆时，需要将下面的配置添加到 proguard-project.txt (proguard.pro) 文件中。

```
-libraryjars libs/wpay-sdk4.7.jar  
  
-keep class com.bx.**{*;}
```

注: -libraryjars libs/wpay-sdk4.7.jar中jar包的名称请填写您下载的最新的jar包。

第三章微派参数说明

3.1 微派支付返回结果列表

参数	类型	描述	样例
result	String	支付结果	success
payType	String	支付方式	wpay
payCode	String	计费点编号	0001
price	String	价格	1
logCode	String	订单编号	A12231526491203042
spResult	String	预留字段	MO
showMsg	String	支付结果描述	支付成功

3.2 微派支付返回状态列表

参数	描述
success	支付成功
fail	支付失败
pass	用户已经支付
pause	计费点已经暂停
error	网络异常、获取不到支付参数等
cancel	用户取消支付

3.3 微派支付类型列表

参数	描述
wpay	微派短信支付
alipay	支付宝支付
czk	充值卡支付
dk	点卡支付
balance	余额支付
uppay	银联支付

第四章微派联系方式

接入过程中如有疑问，请联系我们。

联系人：李必春

Qq: 2899752518