

Paymax Android SDK 使用文档

一、下载

在 <https://www.paymax.cc> 下载Android SDK，下载列表如下包含：

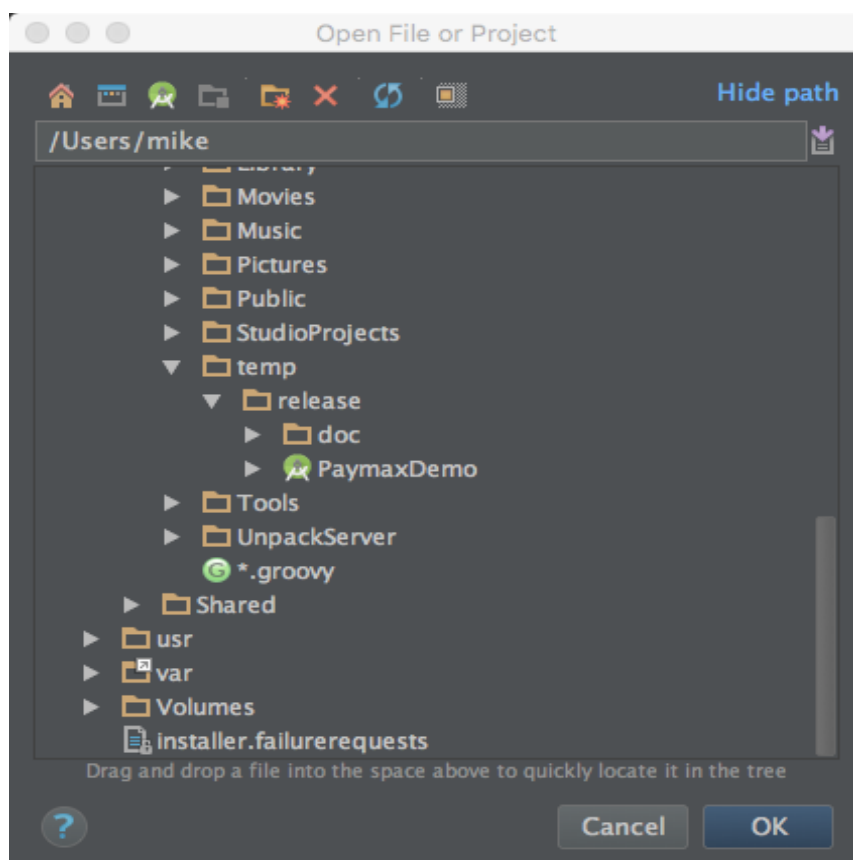


注：要求jdk版本最低为1.7

二、快速体验

Paymax SDK 为开发者提供了demo 程序，可以快速体验 Paymax 接入流程。下载 Paymax SDK 之后将整个目录导入到您的 Android Studio 中。

使用 Android Studio 时，请选择 **File** → **Open...** → **PaymaxDemo**



三、快速集成

导入 Paymax SDK

Android Studio

1. 在你的项目里创建 `libs` 目录:将项目切换到project模式, 定位到你的module, 右击 `New` → `Directory` → 输入 `libs` → `OK`
2. 将下载的 `paymax.jar` 复制、粘贴到 `libs` 目录
3. 同时将下载的微信 `libammsdk.jar`、支付宝 `alipaySdk-20160825` 官方jar文件复制、粘贴到 `libs` 目录
4. 找到module的 `build.gradle` → `dependencies` 填写

```
compile fileTree(dir: 'libs', include: ['*.jar'])
```

5. 如果需要使用拉卡拉支付与人脸识别, 找到 `release` → `libs` 目录, 将其放置到与您项目并行目录下, 并在您项目的 `gradle` 文件进行如下配置

```
repositories {  
    flatDir {  
        dirs '../libs'  
    }  
}  
  
dependencies {  
    compile fileTree(dir: 'libs', include: ['*.jar'])  
    compile(name: 'facesdk-release', ext: 'aar')  
    compile(name: 'lkl-pay-sdk-release', ext: 'aar')  
    compile(name: 'camera-release', ext: 'aar')  
    compile(name: 'idcardcaptorsdk-release', ext: 'aar')  
    compile(name: 'livenessdetectorsdk-release', ext: 'aar')  
    compile(name: 'livenessdetectionviewsdk-release', ext: 'aar')  
    compile files('../libs/paymax.jar')  
}
```

权限声明

```

<!-- 微信和支付宝公共权限 -->
<uses-permission android:name="android.permission.INTERNET" />

<!-- 微信所需权限 -->
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />

<!-- 支付宝所需权限 -->
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />

<!-- 拉卡拉 -->
<uses-permission android:name="android.permission.RECEIVE_USER_PRESENT" />
<uses-permission android:name="android.permission.WAKE_LOCK" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW" />
<uses-permission android:name="android.permission.MOUNT_UNMOUNT_FILESYSTEMS" />
<uses-permission android:name="android.permission.VIBRATE" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.WRITE_SETTINGS" />

<!--人脸识别-->
<uses-permission android:name="android.permission.CAMERA" />

<uses-feature android:name="android.hardware.camera" />
<uses-feature android:name="android.hardware.camera.autofocus" />

```

注册 activity

```

<!--人脸识别-->
<activity
    android:name="com.swwx.facesdk.ui.FaceLivenessActivity"
    android:screenOrientation="portrait" />

<!-- 注册微信 -->
<activity
    android:name="com.swwx.paymax.PaymentActivity"
    android:launchMode="singleTop"
    android:theme="@android:style/Theme.Translucent.NoTitleBar" />

<activity-alias
    android:name=".wxapi.WXPayEntryActivity"
    android:exported="true"
    android:targetActivity="com.swwx.paymax.PaymentActivity" />

```

<!-- 注册支付宝 -->

```
<activity
    android:name="com.alipay.sdk.app.H5PayActivity"
    android:configChanges="orientation|keyboardHidden|navigation|screenSize"
    android:exported="false"
    android:screenOrientation="behind"
    android:windowSoftInputMode="adjustResize|stateHidden" >
</activity>
```

<!-- 注册拉卡拉 -->

```
<activity
    android:name="com.lkl.pay.ui.activity.SDK_StartJarActivity"
    android:screenOrientation="portrait"
    android:theme="@android:style/Theme.Translucent" />
<activity
    android:name="com.lkl.pay.ui.activity.SDK_LoginActivity"
    android:screenOrientation="portrait" >
</activity>
<activity
    android:name="com.lkl.pay.ui.activity.SDK_MsgLoginActivity"
    android:screenOrientation="portrait" >
</activity>
<activity
    android:name="com.lkl.pay.ui.activity.cardPay.BindCreditCardActivity"
    android:screenOrientation="portrait" >
</activity>
<activity
    android:name="com.lkl.pay.ui.activity.cardPay.BindDebitCardActivity"
    android:screenOrientation="portrait" >
</activity>
<activity
    android:name="com.lkl.pay.ui.activity.cardPay.HtmlProtocolActivity"
    android:screenOrientation="portrait" >
</activity>
<activity
    android:name="com.lkl.pay.ui.activity.cardPay.InputCardNoActivity"
    android:screenOrientation="portrait" />
<activity
    android:name="com.lkl.pay.ui.activity.cardPay.InputMessageCodeActivity"
    android:screenOrientation="portrait" />
<activity
    android:name="com.lkl.pay.ui.activity.forget.FindLoginPwdActivity"
    android:screenOrientation="portrait" />
<activity
```

```

        android:name="com.lkl.pay.ui.activity.forget.SetLoginPwdActivity"
        android:screenOrientation="portrait" />
<activity
    android:name="com.lkl.pay.ui.activity.register.SetPayPwdActivity"
    android:screenOrientation="portrait" />
<activity
    android:name="com.lkl.pay.ui.activity.payResult.SuccessActivity"
    android:screenOrientation="portrait" />
<activity
    android:name="com.lkl.pay.ui.activity.payResult.FailureActivity"
    android:screenOrientation="portrait" />
<activity
    android:name="cn.cloudcore.iprotect.plugin.CKbdActivity"
    android:launchMode="singleTask"
    android:configChanges="orientation"
    android:theme="@android:style/Theme.Translucent"
/>

```

四、人脸识别

发起支付前，最好先通过后台接口判断是否需要调起人脸识别。商户后台通过调用 `Paymax Server SDK` 向paymax服务器发起请求，获得结果码的方式进行判断（已经识别过的用户可不用再次识别验证），详细步骤请参考Demo

如果需要调用，请参考如下内容：

```

/**
 * 调用人脸识别接口
 *   uId           商户用户号
 *   authorization  商户号(用户信息->开发信息->Live Secret Key)
 *   realName      姓名
 *   idCardNo      身份证
 *   act           上下文
 *
 */
FaceRecoSDK.startReco(String uId, String authorization, String realName, String idCardNo, Activity act);

```

根据人脸识别结果进行相应处理，如下为处理情况，仅供参考：

```

/**
 * 人脸识别结果处理
 * @param msg
 */

protected void onActivityResult(int requestCode, int resultCode, Intent data)
{
    switch (resultCode) {

        case Activity.RESULT_OK:
            //活体检测成功
            break;

        case Activity.RESULT_CANCELED: {
            if (data != null) {
                int code = data.getIntExtra(FaceLivenessActivity.RESULT_VERIFY_CODE, ResponseHelper.UNKONW_ERROR);
                String msg = data.getStringExtra(FaceLivenessActivity.RESULT_VERIFY_MESSAGE);

                switch (code) {
                    case ResponseHelper.CODE_LIVENESS_INITIALIZE_FAIL:
                        msg = "活体检测初始化失败";
                        break;

                    case ResponseHelper.CODE_LIVENESS_FAIL:
                        msg = "活体检测取样失败";
                        break;

                }

            }
            break;
        }
    }
}

```

详细处理结果展示如下：

Code	
CODE_VERIFY_SUCCESS	人脸识别一致
CODE_INCONFORMITY	人脸识别不一致
CODE_VERIFY_FAILED	签名验证失败
REQUEST_TIMESTAMP_EXPIRE	时间戳过期或者提前
ILLEGAL_ARGUMENT	非法参数
ILLEGAL_DATA	请求数据非法
IDCARDNO_ERROR	身份证号码不一致
REALNAME_IDCARD_NOT_SAME	姓名与身份证号不一致
SERVER_ERROR	服务器内部异常

五、获得 Charge

Charge 对象是一个包含支付信息的 JSON 对象，是 `Paymax SDK` 发起支付的必要参数。该参数需要请求用户服务器获得，服务端生成 charge 的方式参考 [Paymax 官方文档]。SDK 中的 demo 里面提供了如何获取 charge 的实例方法，供用户参考。

六、发起支付

```
/**
 * 发起支付
 * @param activity
 * @param charge 与服务器交互后的充值要素
 * @param callback 充值结果回调接口
 */

PaymaxSDK.pay(Activity activity, String charge, PaymaxCallback callback);
```

注：当前环境为真实支付环境

七、获取支付状态

从 `PaymaxCallback` 的 `onPayFinished()` 方法中获得支付结果。支付成功后，用户服务器也会收到 Paymax 服务器发送的异步通知。最终支付成功请根据服务端异步通知为准。

```
@Override
public void onPayFinished(PayResult result) {
```

```
String msg = "Unknow";
switch (result.getCode()) {
    case Paymax.CODE_SUCCESS:
        //支付成功
        msg = "Complete, Success!";
        break;

    case Paymax.CODE_ERROR_CHARGE_JSON:
        //非空格式
        msg = "charge string isn't a json string error.";
        break;

    case Paymax.CODE_FAIL_CANCEL:
        //用户取消
        msg = "cancel pay.";
        break;

    case Paymax.CODE_ERROR_CHARGE_PARAMETER:
        //字段不全
        msg = "some charge paramters error.";
        break;

    case Paymax.CODE_ERROR_WX_NOT_INSTALL:
        //微信未安装
        msg = "wx not install.";
        break;

    case Paymax.CODE_ERROR_WX_NOT_SUPPORT_PAY:
        //微信版本不支持
        msg = "ex not support pay.";
        break;

    case Paymax.CODE_ERROR_WX_UNKNOW:
        //微信未知错误
        msg = "wechat failed.";
        break;

    case Paymax.CODE_ERROR_ALI_DEAL:
        //支付宝正在处理中
        msg = "alipay dealing.";
        break;

    case Paymax.CODE_ERROR_CONNECT:
        //支付宝网络连接错误
        msg = "alipay network connection failed.";
        break;

    case PaymaxSDK.CODE_ERROR_CHANNEL:
```



```
//渠道错误
msg = "channel error.";
break;

case PaymaxSDK.CODE_ERROR_LAK_USER_NO_NULL:
    //拉卡拉商户用户号为空
    msg = "lklpay user no is null.";
    break;

}
```

注意事项

Android 不允许在 UI 线程中进行网络请求，所以请求 charge 对象的时候请使用 thread+handler 或者使用 AsyncTask 。example 里面的示例程序使用的就是 AsyncTask 方式请求 charge 对象。

混淆设置

用户进行 apk 混淆打包的时候，为了不影响 Paymax SDK 以及渠道 SDK 的使用，请在 proguard-rules 中添加一下混淆规则。

```
-dontwarn com.alipay.**
-keep class com.alipay.** {*; }

-dontwarn com.tencent.**
-keep class com.tencent.** {*; }

-dontwarn com.swwx.paymax.**
-keep class com.swwx.paymax.** {*; }

-dontwarn com.lkl.**
-keep class com.lkl.** {*; }
```