**Android sdk开发记录：**

1. 参考https://blog.csdn.net/weixin\_51522235/article/details/128216091，将项目修改为sdk

2. 使用fat-arr对集成sdk进行embed打包，注意so库的cpu框架与位置，aar文件一定要放在libs文件夹下不然达不到so库，搞不定就重启请缓存重build三重大法，as一坨不出bug有鬼。

ndk {

abiFilters "armeabi-v7a", "arm64-v8a"

}

sourceSets {

main {

jniLibs.srcDirs = ['libs']

}

}

repositories {

flatDir {

dirs 'libs'

}

}

3. 打不进去的so库干脆手动塞进去算了，说的就是你身份证的！

4. 重复的资源要手动删除，转移module时要把资源也一并全都转移过去。

**逻辑更改事项：**

1. 传入通过一个接口与一个实现类实现，通过接口传入账户、密钥、卡类型、是否人脸、原活动等。实现跳转至不展现的导引类main1。

2. main1将输入数据存入userinfo单例类然后跳转到对应的活动。

3. 跳转后根据网络请求判断是否要给它过，比如日期、安卓id、包名等，还有账号和密钥。最好每个关键节点的调用都整一下。

4. 根据不同结果返回时往nfc单例的result存内容，然后每次启动到引导活动就重置单例类，最后通过单例类获取sdk结果。

5. 导引活动可以通过style的透明theme无感切换。

<style name="Transparent" parent="Theme.AppCompat.Light.NoActionBar">

<item name="android:windowBackground">@android:color/transparent</item>

<item name="android:windowIsTranslucent">true</item>

<item name="android:windowNoTitle">true</item>

</style>

<activity

android:name=".MainActivity1"

android:theme="@style/Transparent"

android:exported="false"

android:screenOrientation="portrait">

6. 把引导活动的启动和关闭动画都取消，记得把任务栏颜色调成跟别的一致。

overridePendingTransition(0, 0);

Window window = getWindow();

window.addFlags(WindowManager.LayoutParams.FLAG\_DRAWS\_SYSTEM\_BAR\_BACKGROUNDS);

window.setStatusBarColor(ContextCompat.getColor(this, R.color.purple\_700));

1. 注意不同流程返回的内容格式需一致，在每个可能的地方加上返回信息，例如查看每一个活动的每一个finish，若为结果页可以把插入result放在网络调用handler或者识别结果出来的地方，这样完成或返回按钮效果都是一样的。finish分为用户主动按钮finish和机制自动finish，自动finish前加返回，主动finish需要在之前加返回。
2. FileProvider name是app与sdk不能相同的，可以把sdk中的改掉，这样才能实现app与sdk demo同时安装。