1. 我会实现推特登陆SDK来讲解如何使用三方SDK智能布署工具
2. 首先是贴出推特相关文档和原生示例

开发文档地址

<https://dev.twitter.com/twitterkit/overview>

如何使用Android的SDK

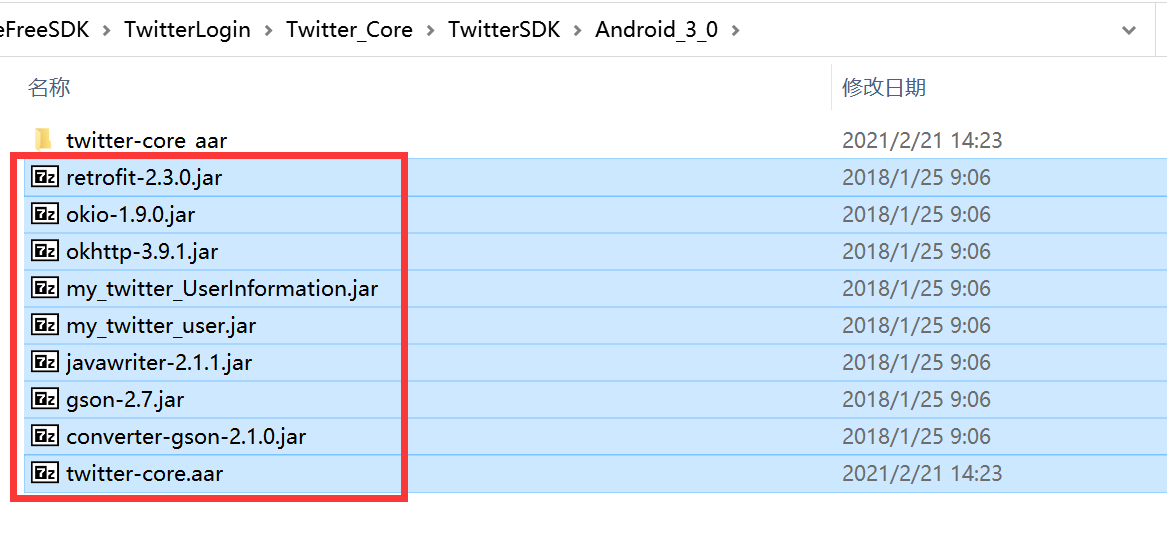
<https://dev.twitter.com/twitterkit/android/installation>

示例下载地址:

<https://github.com/twitter>

<https://github.com/twitter/twitter-kit-android>

1. 下载推特的Android SDK



主要是twitter-core.aar

其他是依赖包

1. 再下载三方SDK智能布署工具

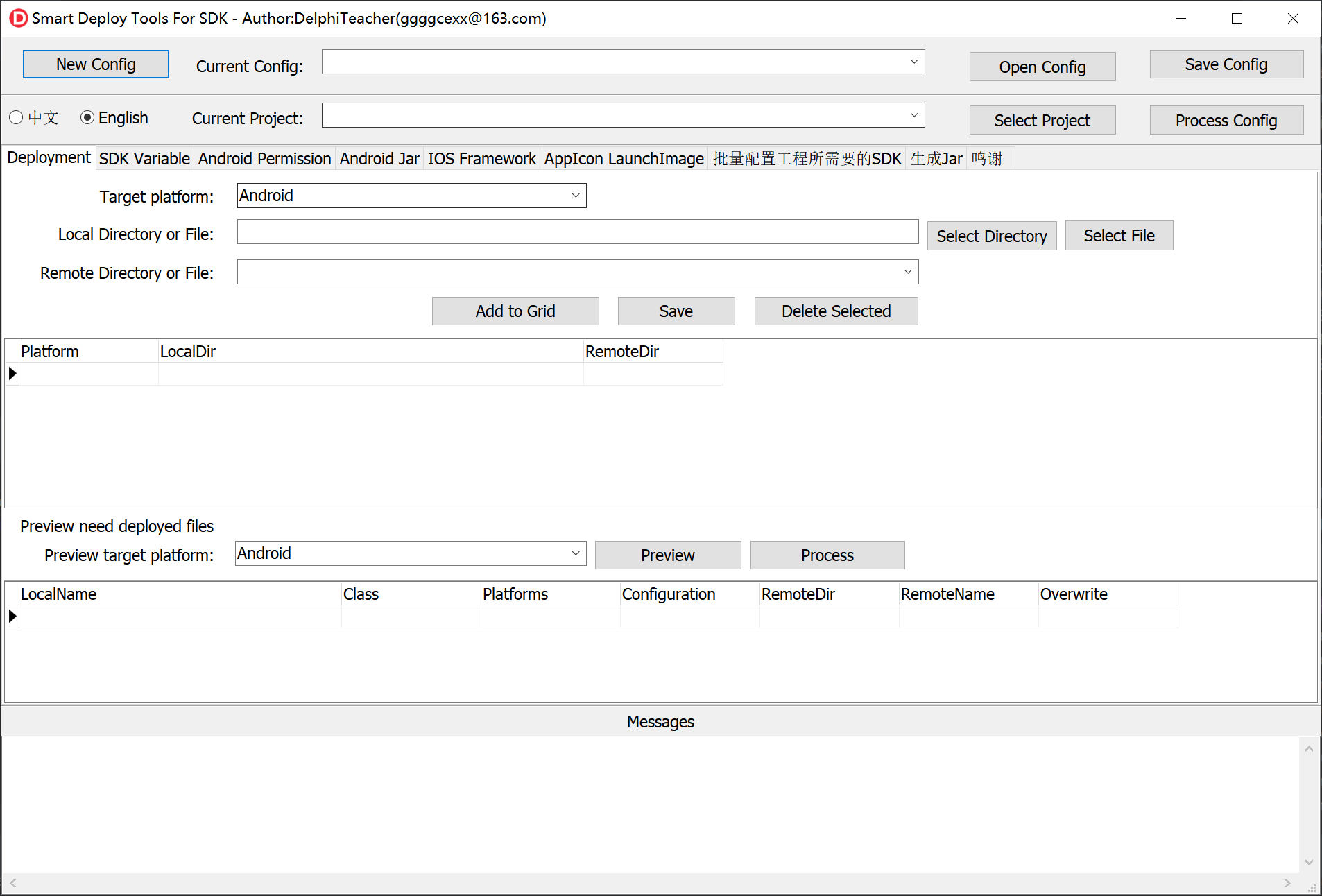
编译好的：

<http://www.orangeui.cn/download/sdk/OrangeSDK_SmartDeploy.zip>

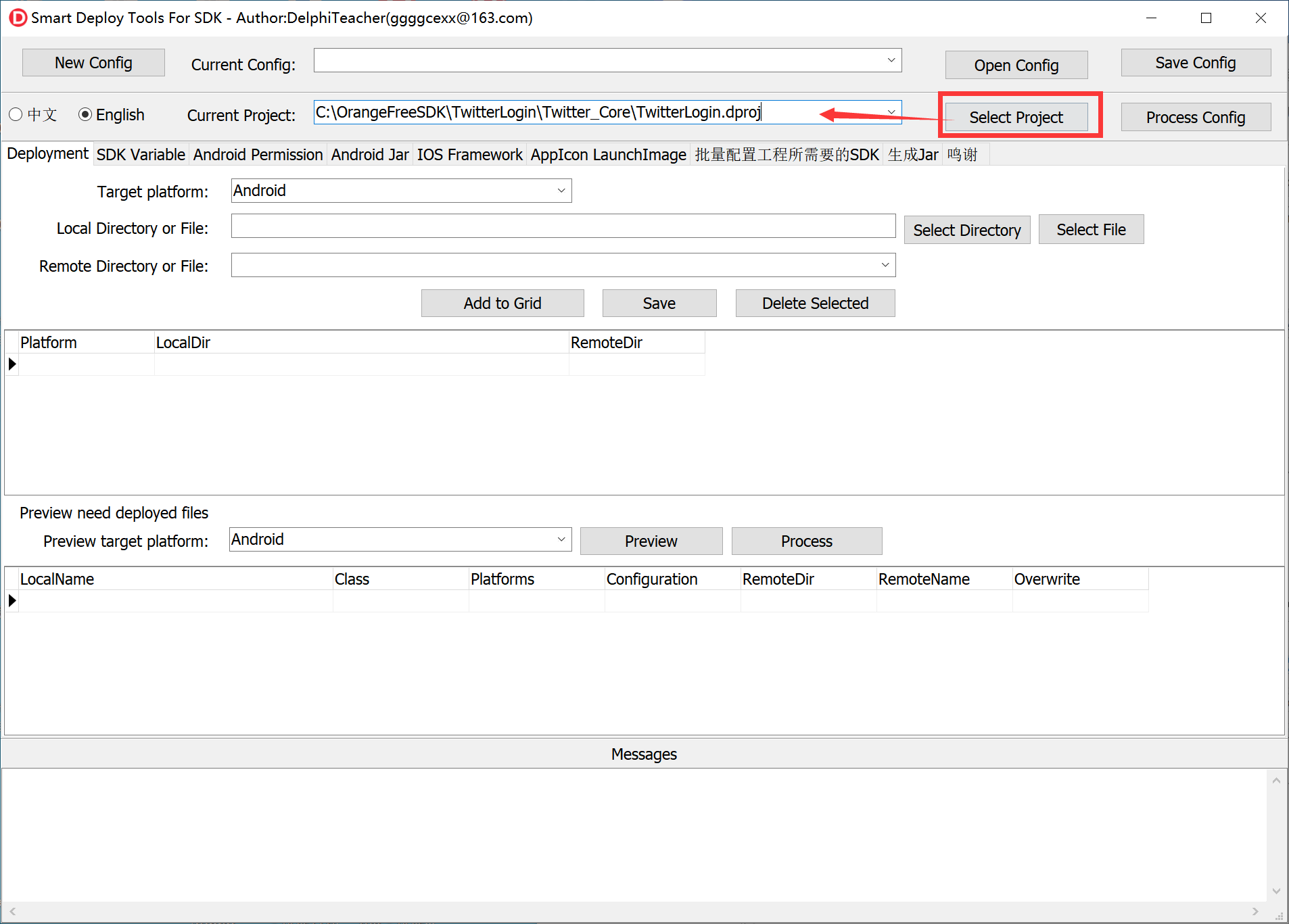
或者直接下载源码地址自己编译：

<https://github.com/DelphiTeacher/OrangeFreeSDK>

三方SDK智能布署工具的界面如下：

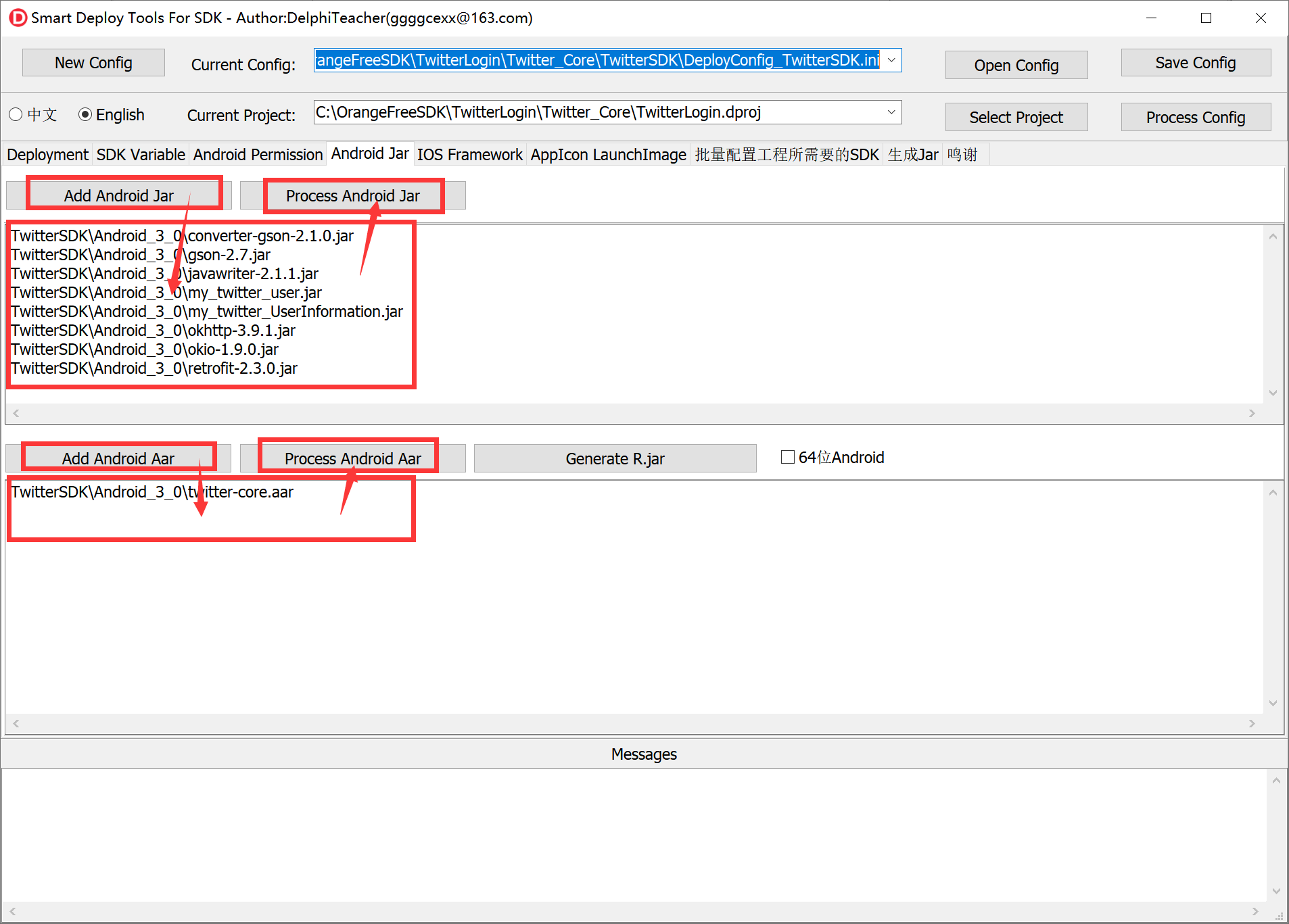


1. 先点击“Select Project”按钮来选择要配置的工程

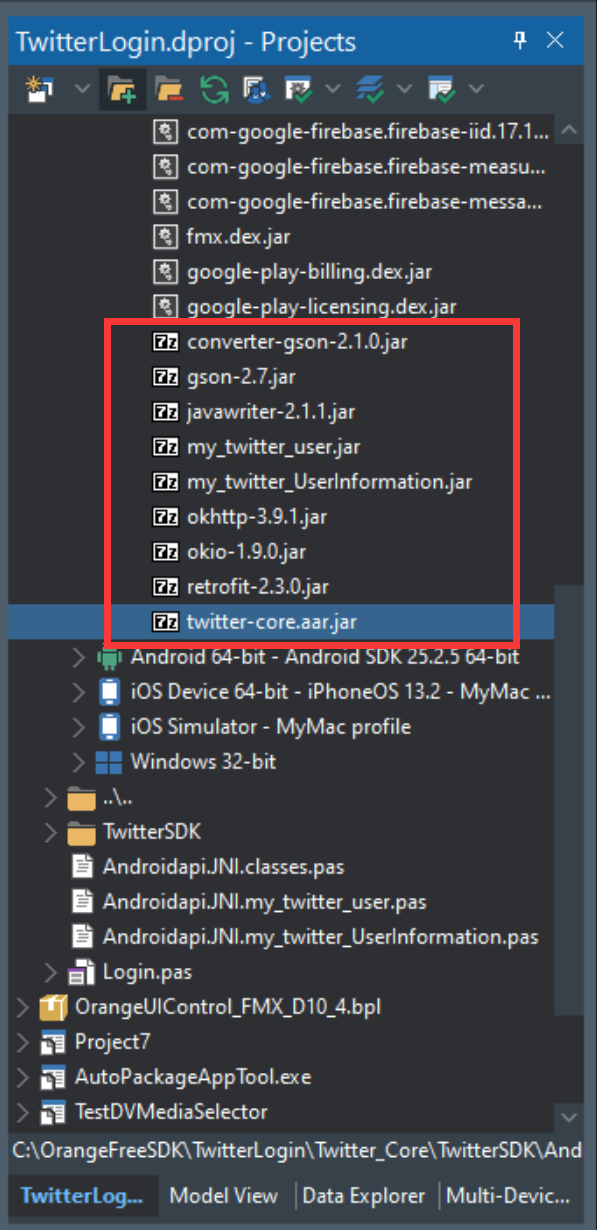


1. 再点击”Add Android Jar”按钮，添加所有的jar，再点击”Process Android Jar”

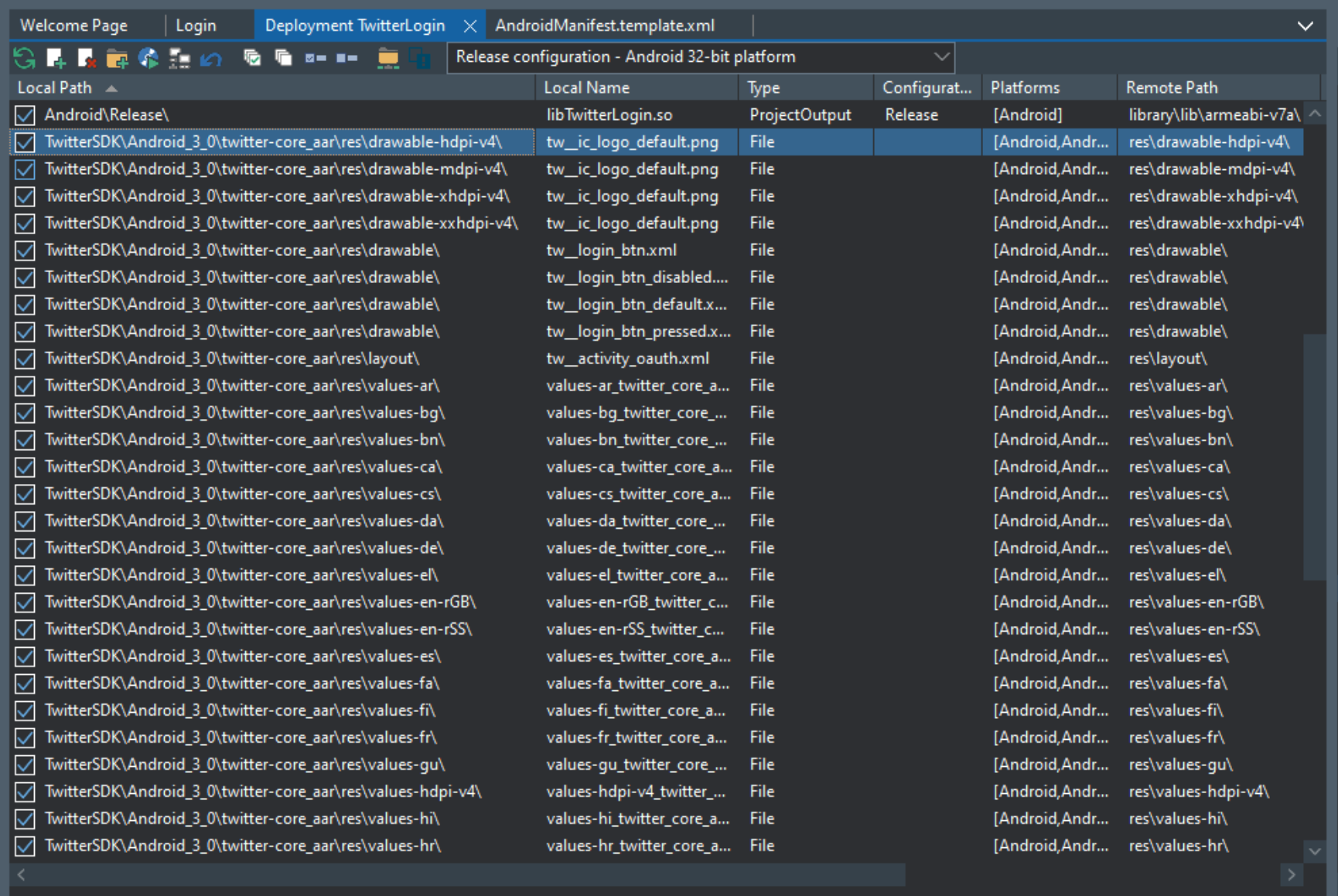
点击”Add Android Aar”按钮，添加所要的aar，再点击“Process Android Aar”



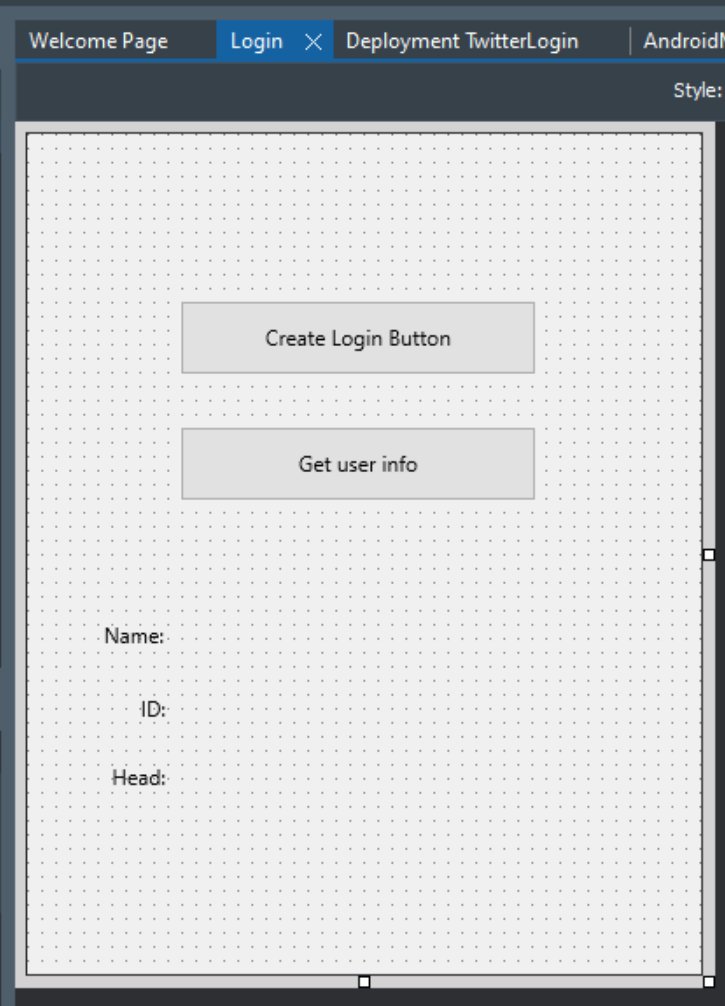
Jar和aar都已经添加到工程中去了



Aar包也解压出来了，其中的资源也布署到工程中去了



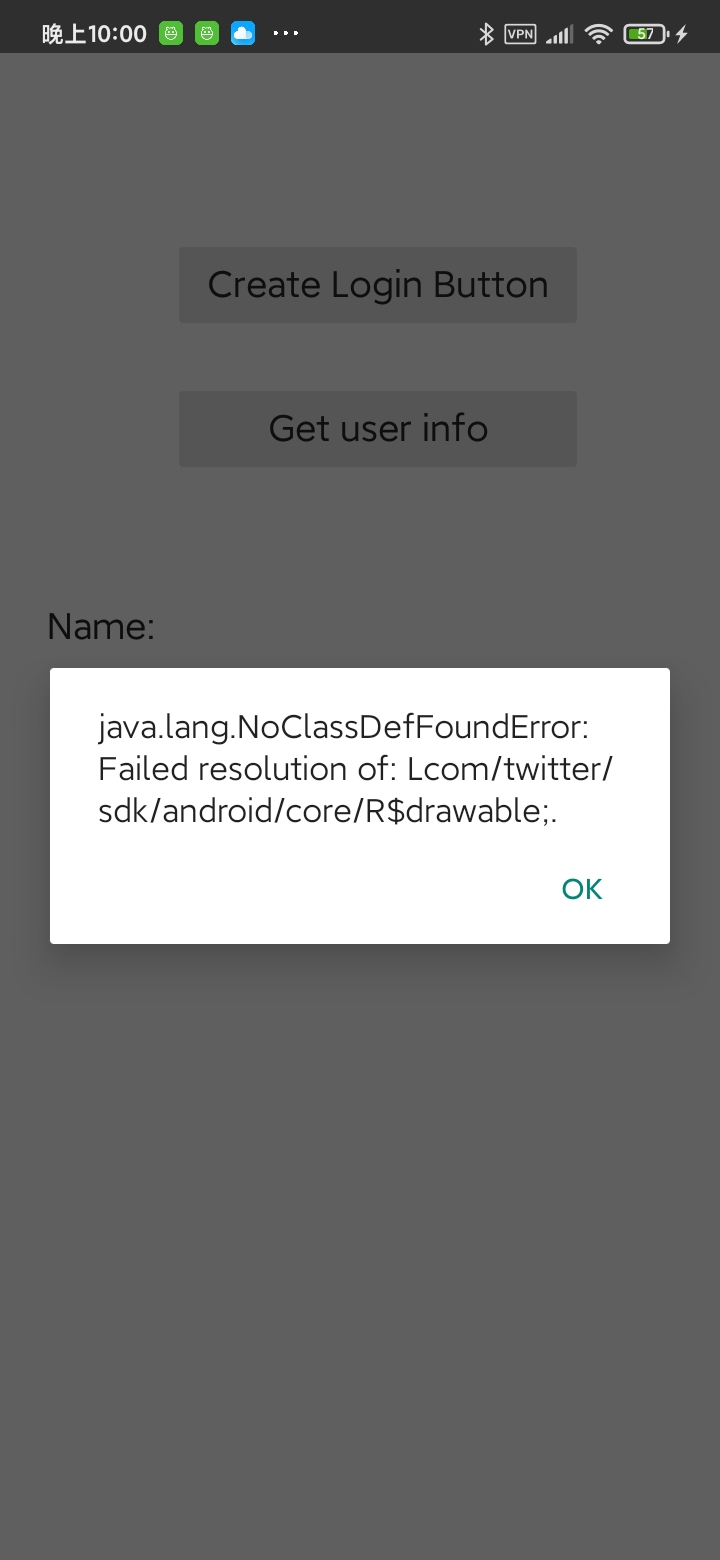
1. 接下来将jar翻译成Delphi的调用单元，在窗体中进行调用，先创建推特按钮，显示在窗体上



Button“Create Login Button”

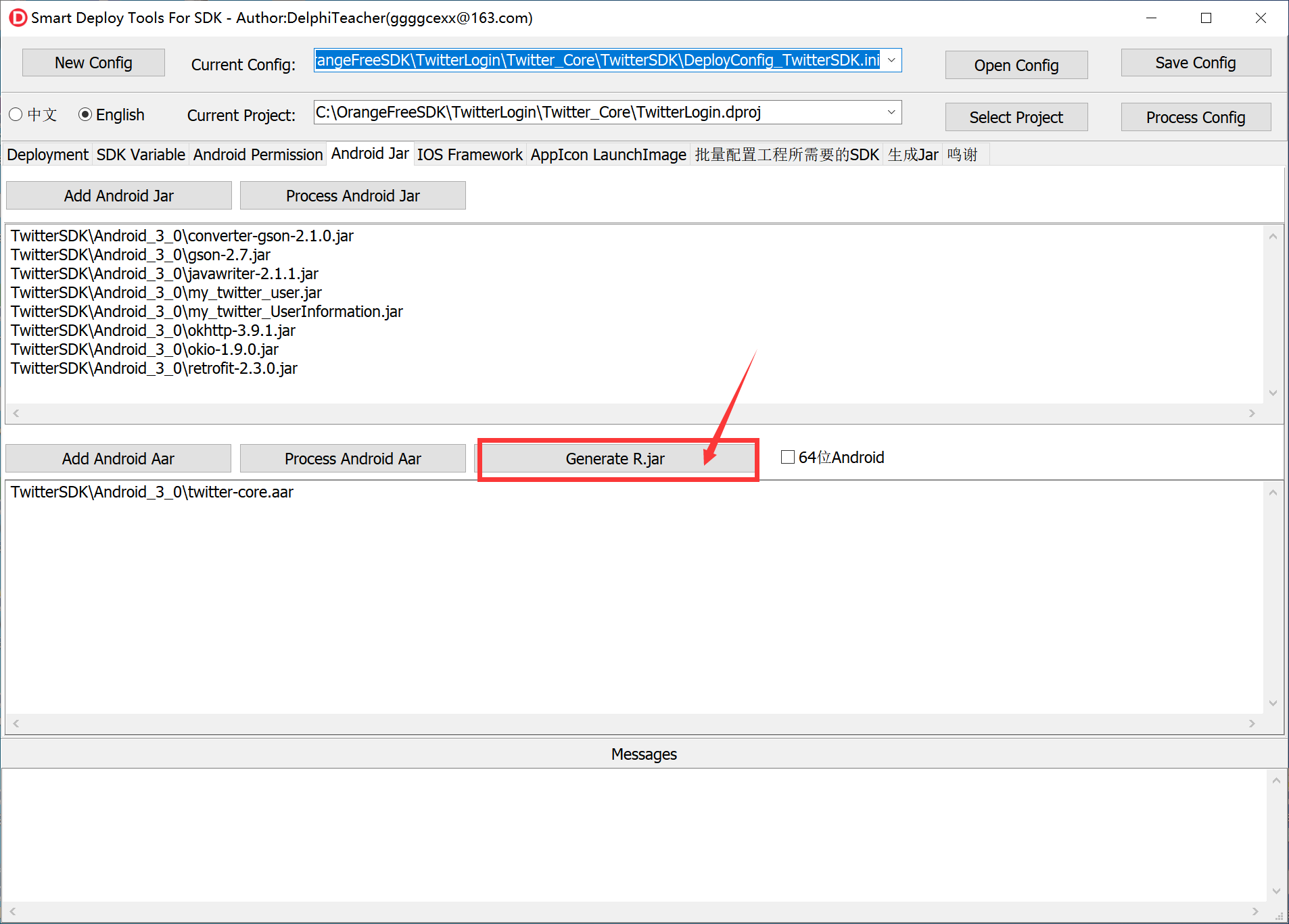
Button “Get user info”

1. 接下来运行到真机上查看效果,点击“Create Login Button”，会弹出一个异常，



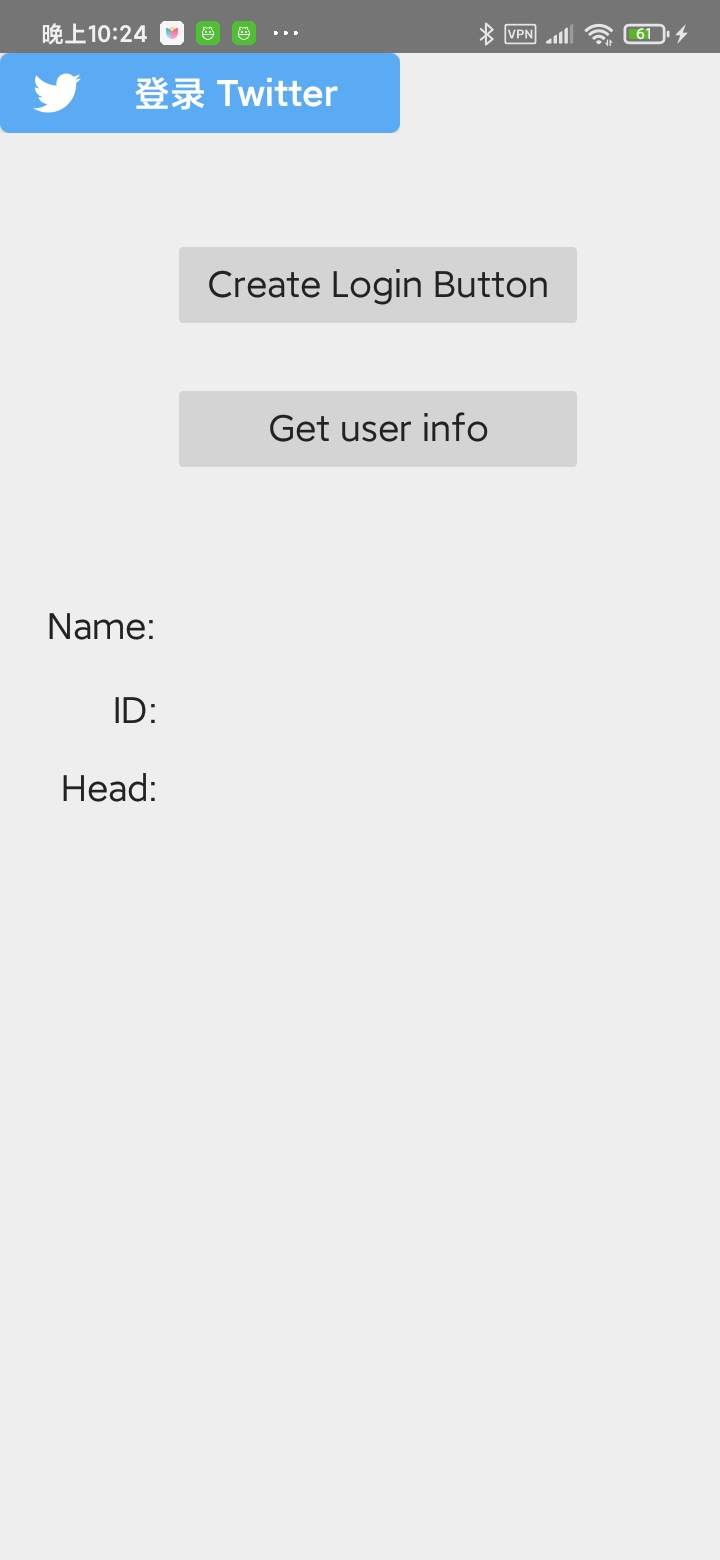
因为aar包需要访问自己包中的资源，因此我们需要生成该aar包的R.java，并打包成jar添加到工程中去。

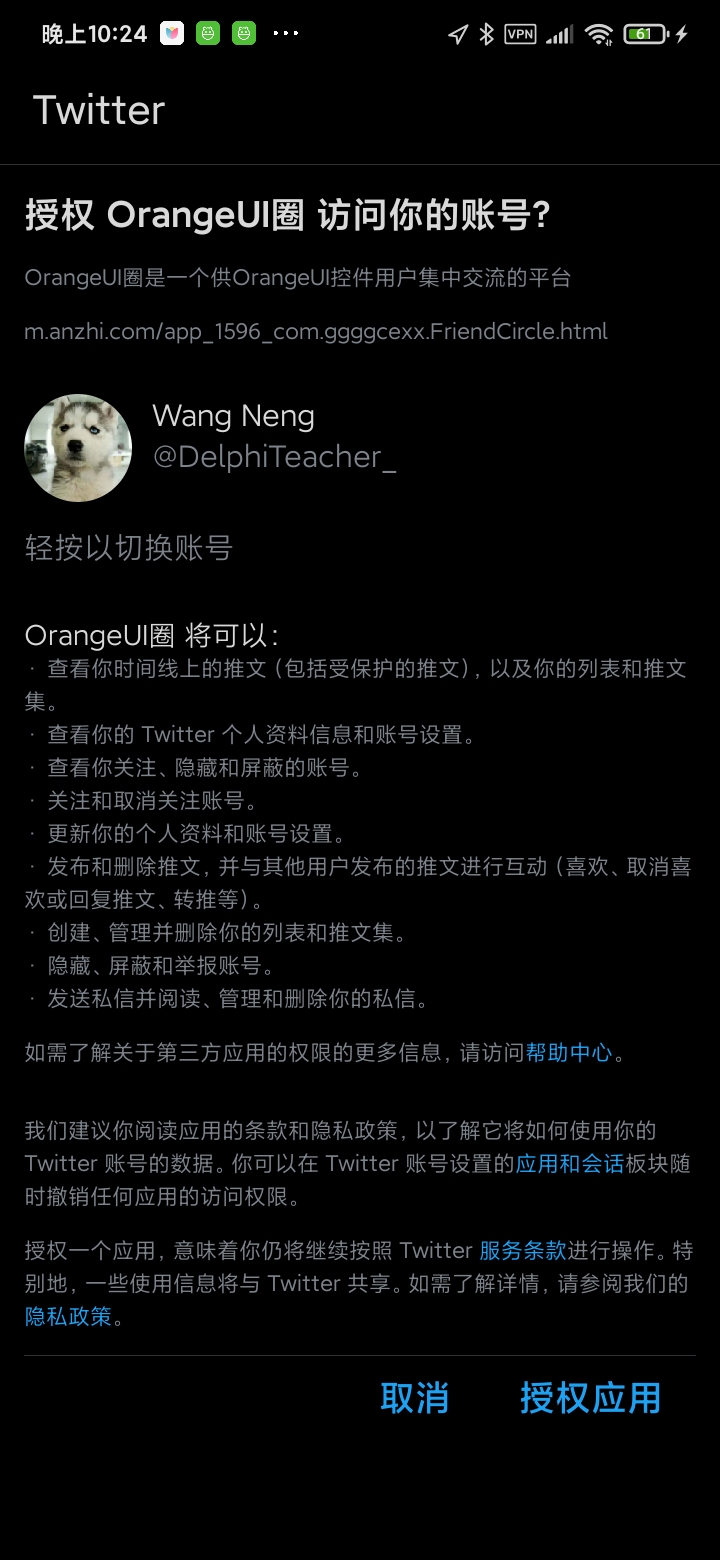
您需要点击“Generate R.jar”按钮

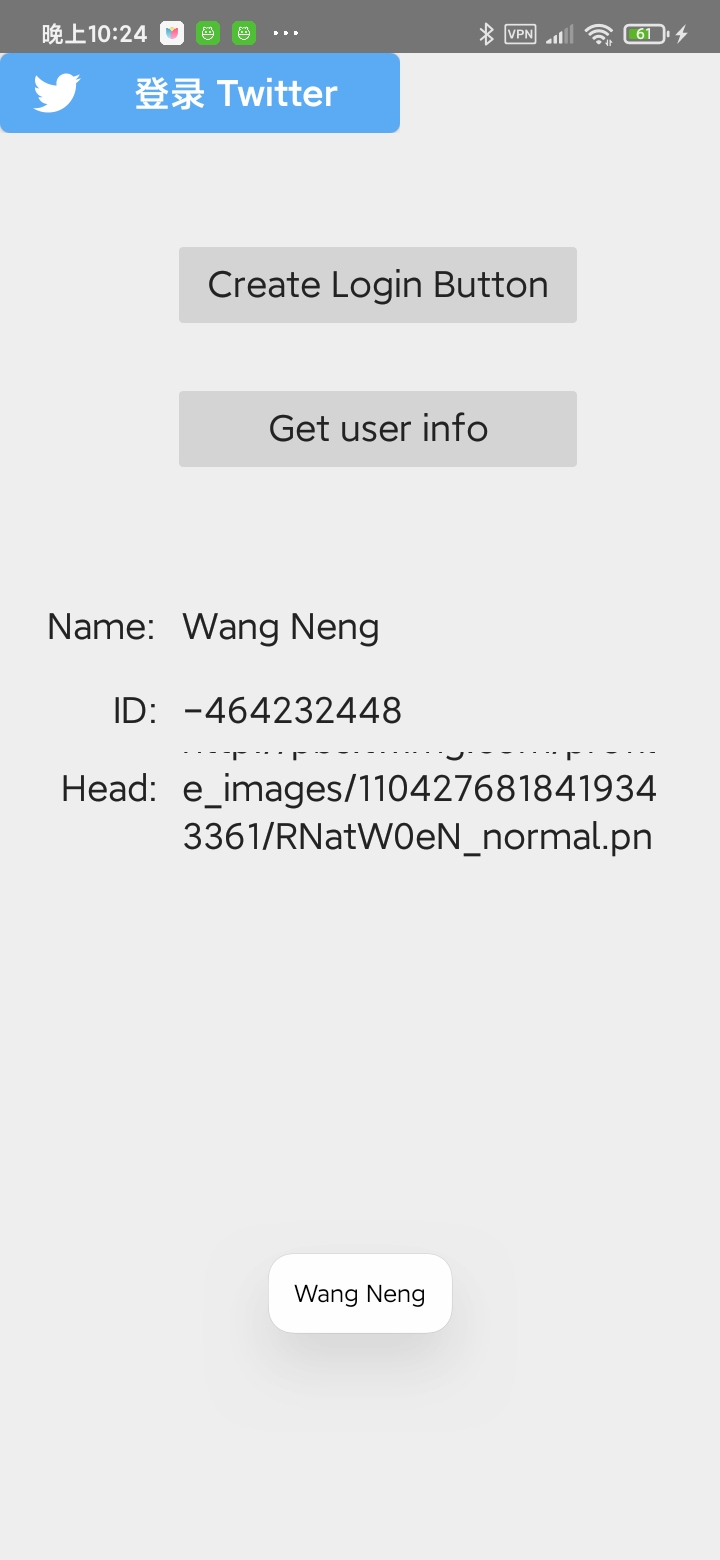


它会生成R.jar并添加到工程中去

1. 再编译并运行到真机查看效果



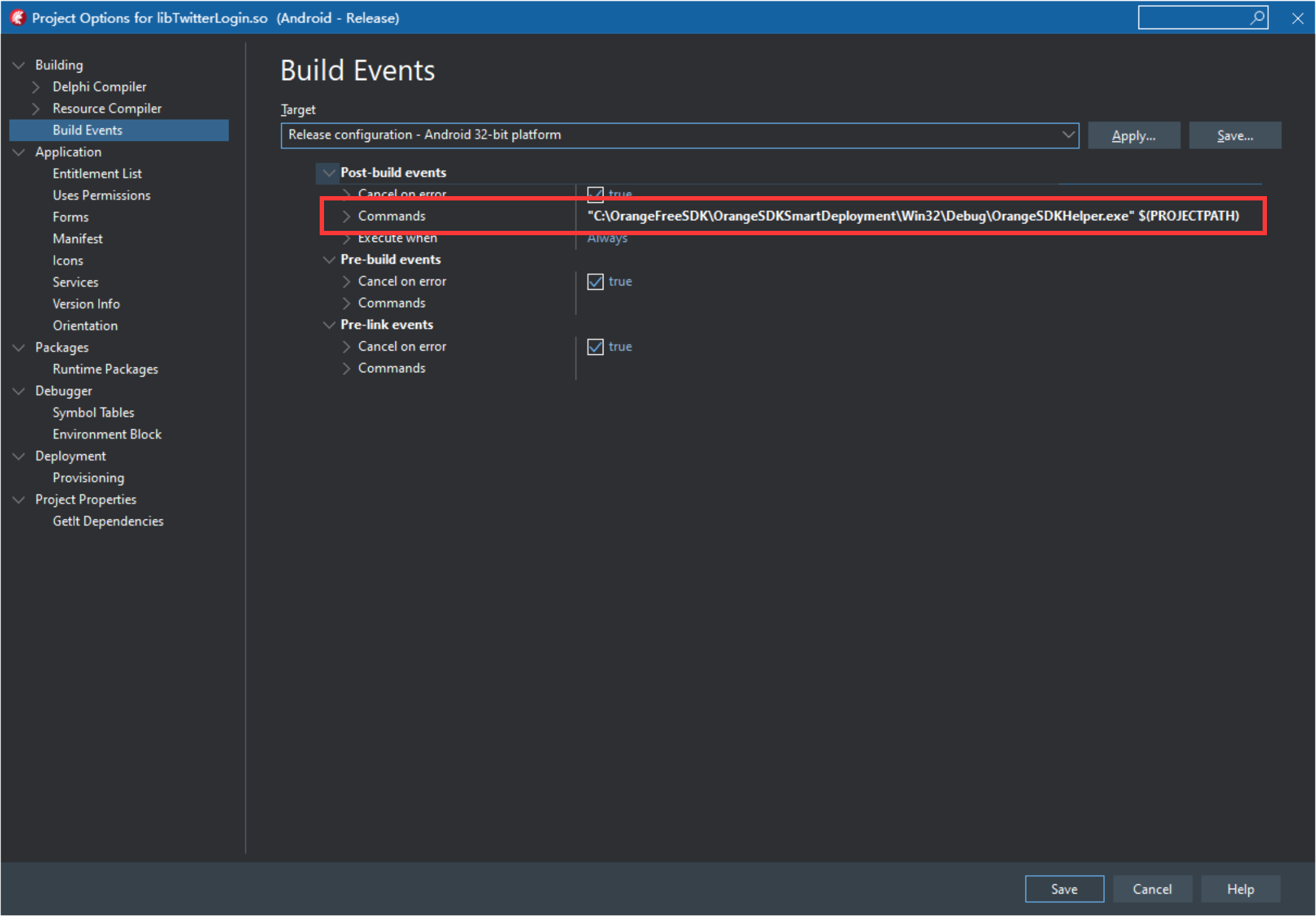




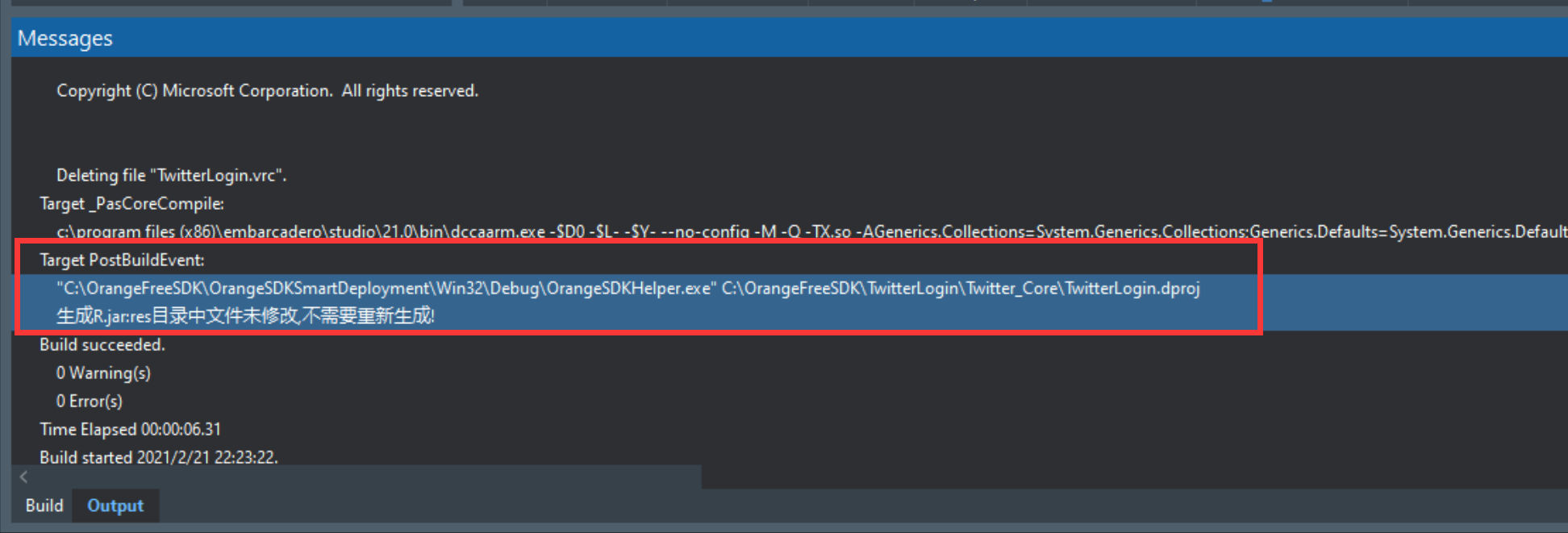
1. 如果工程的资源文件布署被改动过之后，需要重新点击“Generate R.jar”按钮，如果你嫌这个麻烦，那么可以使用OrangeSDKHelper.exe来自动处理

在工程中配置Post-build events为：

"C:\OrangeFreeSDK\OrangeSDKSmartDeployment\Win32\Debug\OrangeSDKHelper.exe" $(PROJECTPATH)



每次编译后，你都能在Output页中查看相关的信息



# Getting Started[¶](https://dev.twitter.com/twitterkit/android/installation" \l "getting-started" \o "Permalink to this headline)

新建Android工程，

包名为com.ggggcexx.orangeui

**配置Gradle**

dependencies {

compile 'com.twitter.sdk.android:twitter-core:3.1.1'

compile 'com.twitter.sdk.android:tweet-ui:3.1.1'

compile 'com.twitter.sdk.android:tweet-composer:3.1.1'

compile 'com.twitter.sdk.android:twitter-mopub:3.1.1'

}

**初始方式一:**

在values.xml中添加:

<resources>

<string android:name="com.twitter.sdk.android.CONSUMER\_KEY">XXXXXXXXXXX</string>

<string android:name="com.twitter.sdk.android.CONSUMER\_SECRET">XXXXXXXXXXX</string>

</resources>

在Application创建时添加:

public class CustomApplication {

public void onCreate() {

Twitter.initialize(this);

}

}

**初始方式二:**

public void onCreate() {

TwitterConfig config = new TwitterConfig.Builder(this)

.logger(new DefaultLogger(Log.DEBUG))

.twitterAuthConfig(new TwitterAuthConfig("CONSUMER\_KEY", "CONSUMER\_SECRET"))

.debug(true)

.build();

Twitter.initialize(config);

}

用推特登陆: Log In with Twitter：

https://dev.twitter.com/twitterkit/android/log-in-with-twitter

**登陆方式一: Log In Button**

import com.twitter.sdk.android.core.Callback;

import com.twitter.sdk.android.core.Result;

import com.twitter.sdk.android.core.TwitterException;

import com.twitter.sdk.android.core.TwitterSession;

import com.twitter.sdk.android.core.identity.TwitterLoginButton;

loginButton = (TwitterLoginButton) findViewById(R.id.login\_button);

loginButton.setCallback(new Callback<TwitterSession>() {

@Override

public void success(Result<TwitterSession> result) {

// Do something with result, which provides a TwitterSession for making API calls

}

@Override

public void failure(TwitterException exception) {

// Do something on failure

}

});

**获取到登陆后的Token:**

TwitterSession session = TwitterCore.getInstance().getSessionManager().getActiveSession();

TwitterAuthToken authToken = session.getAuthToken();

String token = authToken.token;

String secret = authToken.secret;

**请求获取用户的邮箱地址:Request User Email Address**

Before using this feature, ensure that “Request email addresses from users” is checked for your Twitter app.

TwitterAuthClient authClient = new TwitterAuthClient();

authClient.requestEmail(session, new Callback<String>() {

@Override

public void success(Result<String> result) {

// Do something with the result, which provides the email address

}

@Override

public void failure(TwitterException exception) {

// Do something on failure

}

});