## 分享SDK使用说明文档

### 一、分享流程

分享SDK的工作流程其实很简单，即：

pic

### 二、设计思路

鉴于微博分享回调必须依附于Activity，腾讯QQ、QZone也是一样。因此分享中，用一个Activity作为分享渠道的转发，通过onActivityResult（）回调结果。

### 三、分享配置

1、混淆相关，在proguard-rules里面加上：

**-ignorewarnings  
-keep** class com.tencent.mm.sdk.\*\* {  
 \*;  
}

2、AndroidManifest.xml相关配置：

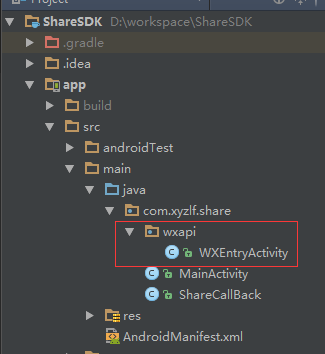
<meta-data  
 android:name="tencent\_qq\_appid"  
 android:value="@string/tencent\_qq\_appid"/>  
<meta-data  
 android:name="weixin\_key"  
 android:value="@string/weixin\_key"/>  
<meta-data  
 android:name="weixin\_redirecturi"  
 android:value="@string/weixin\_redirecturi" />  
<meta-data  
 android:name="sina\_weibo\_redirecturi"  
 android:value="@string/sina\_weibo\_redirecturi" />  
<meta-data  
 android:name="sina\_weibo\_key"  
 android:value="@string/sina\_weibo\_key" />

<!-- 微信分享注册 -->  
<activity  
 android:name=".wxapi.WXEntryActivity"  
 android:configChanges="keyboardHidden|orientation|screenSize"  
 android:exported="true"  
 android:screenOrientation="portrait"  
 android:theme="@android:style/Theme.Translucent.NoTitleBar" />  
  
<!-- QQ SDK 需要註冊Activity -->  
<activity  
 android:name="com.tencent.connect.common.AssistActivity"  
 android:configChanges="orientation|keyboardHidden"  
 android:screenOrientation="behind"  
 android:theme="@android:style/Theme.Translucent.NoTitleBar" />  
<activity  
 android:name="com.tencent.tauth.AuthActivity"  
 android:launchMode="singleTask"  
 android:noHistory="true" >  
 <intent-filter>  
 <action android:name="android.intent.action.VIEW" />  
 <category android:name="android.intent.category.DEFAULT" />  
 <category android:name="android.intent.category.BROWSABLE" />  
 <data android:scheme="tencent222222" />  
 <!-- 100380359 100381104 222222 -->  
 </intent-filter>  
</activity>  
<!-- QQ SDK 需要註冊 Activity -->

微信分享配置需特别注意：

必须生成一个叫做：WXEntryActivity的Activity，完整包名必须是：{App packageName}.wxapi.WXEntryActivity，如下：

<!-- 微信分享注册 -->  
<activity  
 android:name=".wxapi.WXEntryActivity"  
 android:configChanges="keyboardHidden|orientation|screenSize"  
 android:exported="true"  
 android:screenOrientation="portrait"  
 android:theme="@android:style/Theme.Translucent.NoTitleBar" />



### 四、使用方式

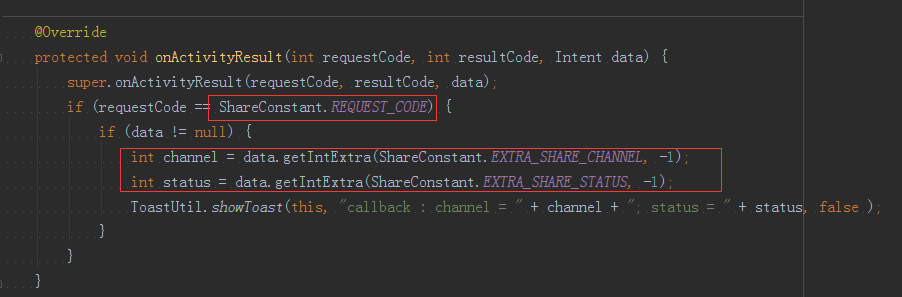
**1、调出分享界面**



*/\*\*  
 \* 使用统一数据结构  
 \*/*public void showShareDialog() {  
 ShareEntity testBean = new ShareEntity("我是标题", "我是内容，描述内容。");  
 testBean.setUrl("https://www.baidu.com"); //分享链接  
 testBean.setImgUrl("https://www.baidu.com/img/bd\_logo1.png");  
 ShareUtil.*showShareDialog*(this, testBean, ShareConstant.*REQUEST\_CODE*);  
}

*/\*\*  
 \* 使用数组数据结构  
 \*/*public void showShareDialog() {  
 ShareEntity testBean = new ShareEntity("我是标题", "我是内容，描述内容。");  
 testBean.setUrl("https://www.baidu.com"); //分享链接  
 testBean.setImgUrl("https://www.baidu.com/img/bd\_logo1.png");  
  
 SparseArray<ShareEntity> sparseArray = new SparseArray<>();  
 sparseArray.put(ShareConstant.*SHARE\_CHANNEL\_QQ*, testBean);  
 sparseArray.put(ShareConstant.*SHARE\_CHANNEL\_QZONE*, testBean);  
 sparseArray.put(ShareConstant.*SHARE\_CHANNEL\_WEIXIN\_CIRCLE*, testBean);  
 sparseArray.put(ShareConstant.*SHARE\_CHANNEL\_WEIXIN\_FRIEND*, testBean);  
 sparseArray.put(ShareConstant.*SHARE\_CHANNEL\_SINA\_WEIBO*, testBean);  
 sparseArray.put(ShareConstant.*SHARE\_CHANNEL\_SYSTEM*, testBean);  
   
 ShareUtil.*showShareDialog*(this, testBean, ShareConstant.*REQUEST\_CODE*);  
}

处理分享回调接口，在调用的Activity的onActivityResult中处理：



**2、使用分享功能**

**不需要分享界面，只是单纯的使用分享功能：**

*/\*\*  
 \* 使用具体的分享功能  
 \*  
 \* QQ举例  
 \*/*public void startShare(int channel) {  
  
 channel = ShareConstant.*SHARE\_CHANNEL\_QQ*;  
  
 ShareEntity testBean = new ShareEntity("我是标题", "我是内容，描述内容。");  
 testBean.setUrl("https://www.baidu.com"); //分享链接  
 testBean.setImgUrl("https://www.baidu.com/img/bd\_logo1.png");  
 ShareUtil.*startShare*(this, channel, testBean, ShareConstant.*REQUEST\_CODE*);  
}

### 五、渠道动态配置

在使用方式的基础上，传递渠道值，可动态展示分享渠道，如下：

*/\*\*  
 \* 渠道动态可配  
 \*  
 \* 只展示微信、微信朋友圈  
 \*/*public void startShare() {  
  
 ShareEntity testBean = new ShareEntity("我是标题", "我是内容，描述内容。");  
 testBean.setUrl("https://www.baidu.com"); //分享链接  
 testBean.setImgUrl("https://www.baidu.com/img/bd\_logo1.png");  
   
 int channel = ShareConstant.*SHARE\_CHANNEL\_WEIXIN\_CIRCLE* | ShareConstant.*SHARE\_CHANNEL\_WEIXIN\_FRIEND*;  
   
 ShareUtil.*showShareDialog*(this, channel, testBean, ShareConstant.*REQUEST\_CODE*);  
}

### 六、分享大图

（1）分享本地图片

ShareEntity testBean = new ShareEntity("我是标题", "我是内容，描述内容。");  
1、本地drawable图片，使用方式：testBean.setDrawableId(R.drawable.icon);

2、本地手机SD卡图片，使用方式：testBean.setImgUrl("/sdpath/filepath/icon.png");

3、本地代码截屏Bitmap，使用方式：

Bitmap bitmap = null; // 你自己截屏的bitmap

String filePath = ShareUtil.*saveBitmapToSDCard*(this, bitmap);  
testBean.setImgUrl(filePath);

（2）微信分享大图

添加大图属性：

ShareEntity testBean = new ShareEntity("", "");  
testBean.setShareBigImg(true);

其他设置如上介绍。