

# 小说Android-SDK对接文档

## 1、媒体接入步骤

通过媒介或市场人员向我方获取appKey和secretKey，并且提供媒体方的奖励货币单位和汇率（用作给用户阅读等奖励）以及每日奖励上限，媒体方需提供给我方回调接口地址（用作完成阅读的奖励通知）

## 2、对接sdk步骤

### 1. 在build.gradle中添加

```
allprojects {  
    repositories {  
        maven { url  
            "http://47.97.184.214:8081/nexus/content/groups/public" }  
    }  
}
```

### 2. 在app/build.gradle中 添加

```
```java  
android {  
    defaultConfig {  
        ndk { abiFilters "armeabi" }  
    }  
}  
  
dependencies{  
    implementation ("com.iwangzha:novel:2.3.0"){  
        transitive = true  
    }  
}
```

### 3. 在gradle.properties添加

```
Android.useDeprecatedNdk=true
```

### 4. 在AndroidManifest.xml中添加

```
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"  
>  
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
```

```

<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" />
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"
/>
<uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission
android:name="android.permission.REQUEST_INSTALL_PACKAGES" />
<uses-permission android:name="android.permission.READ_SETTINGS" />
<uses-permission android:name="android.permission.GET_TASKS" />
<uses-permission android:name="android.permission.WAKE_LOCK" />

```

在application标签中添加

```
android:networkSecurityConfig="@xml/network_security_config"
```

创建 res/xml/network\_security\_config.xml

```

<?xml version="1.0" encoding="utf-8"?>
<network-security-config>
  <base-config cleartextTrafficPermitted="true">
    <trust-anchors>
      <certificates src="system" />
    </trust-anchors>
  </base-config>
</network-security-config>

```

5. 将我方提供的proguard.cfg文件内容添加到混淆文件中
6. 初始化SDK

```

public class DemoApplication extends Application {
    public void onCreate() {
        super.onCreate();
        //打开日志
        NovelSdk.setDebug(true);
        NovelSdk.init(Application context, String appkey, String secretKey)
    }
}

```

7. 初始化接口说明

```

/**
 * sdk初始化入口
 *
 * @param context 必须是application context
 * @param appkey 合作方Appkey
 * @param secretKey appkey对应的密钥
 */
public static void init(Application context, String appkey, String
secretkey)

```

## 8. 使用SDK:

```

NovelManager.init(context, consumerId, callback)

/**
 * sdk初始化入口
 *
 * @param context 必须是Activity context
 * @param consumerId 用户Id
 * @param NovelTxCallback 回调
 */

public static void init(final Activity context, String consumerId,
NovelTxCallback callback)

public class NovelTxCallback {
    //UserId 为空时, 去登陆按钮点击事件回调
    public void goLogin(){}
}

```

## 3、参数

参数	类型	是否必须	说明
activity	Activity	是	
appKey	String	是	合作方AppKey
msecretKey	String	是	appKey对应的密钥(secretKey)
consumerId	String	是	合作方自己用户的唯一识别ID

## 4、回调接口（如果不需要奖励，不用提供此接口）

当用户完成阅读章节，会调用媒体app的回调接口，接口请求方式为 [POST] ，请求参数为：

参数	类型	是否必须	说明
consumerId	String	是	媒体方的用户唯一标示
rewardAmount	String	是	给用户的奖励数额