

# 龙珠悠悠 Android SDK 接入说明文档

## 1. YoYo-SDK 简介

YoYo-SDK 是 龙珠悠悠直播 提供给第三方开发的开发工具包。方便开发 者可以用精简的代码直接为自己的应用接入龙珠悠悠的直播资源和直播功能。

## 2. 支持环境

支持 android 4.0.3 及以上版本，支持 armv5/armv7a/armv64/x86 运行

## 3. 获取 Appkey

YoYo-SDK 接入前需要联系官方分配一个 Appkey 以便继续后续接入。

## 4. 使用 Android Studio 集成

- ◆ 将 yoyo\_sdk-xxx.aar 文件放入项目的 libs 目录下，
- ◆ 将 pl\_droidsonroids\_gif 文件夹中的 so 文件根据自己项目需求拷贝到 libs 目录下面
- ◆ 如果需要使用我们的播放器请将 KSYVideoView.class 和 ksyMediaPlayer 目录下的文件分别拷贝到项目的代码目录和 libs 目录下面
- ◆ 然后在项目的 build.gradle 文件中配置中添加：

```
repositories {  
    flatDir {  
        dirs 'libs'  
    }  
}  
  
dependencies {  
    compile (name:"live_sdk-release", ext:"aar")  
    compile 'org.java-websocket:Java-WebSocket:1.3.0'  
    ...  
}
```

## 5. 配置 Appkey

在 manifest 文件中填写 Appkey:

```
<manifest.....>  
<application.....>  
<meta-data android:name=" YOYO_APP_KEY" android:value="your_appkey"></meta-data>
```

```
</application>
</manifest>
```

在代码中配置 appkey:  
在程序入口调用如下方法:

```
YoyoApi.init(Application app, String appKey, YoyoServerInterface<Boolean> iface);
```

请查看 [初始化](#) 模块, 该模块有详细介绍

## 6. 权限说明

在 manifest 文件中添加以下权限

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.WAKE_LOCK" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
```

## 7. manifest 剩余配置

添加如下 activity 配置

```
<activity android:name="com.xcyo.yoyo.ui.activity.room.MediaRoomActivity"
android:screenOrientation="portrait"/>
<activity android:name="com.xcyo.yoyo.ui.activity.report.ReportActivity"
android:screenOrientation="portrait"/>
<activity android:name="com.xcyo.yoyo.ui.activity.report.other.ReportOtherActivity"
android:screenOrientation="portrait"/>
<activity android:name="com.xcyo.yoyo.ui.activity.report.suc.ReportSucActivity"
android:screenOrientation="portrait"/>
<activity android:name="com.xcyo.yoyo.ui.activity.roomEnd.EndRoomActivity"
android:screenOrientation="portrait"/>
```

## 8. 初始化

启动 SDK 之前, 必须初始化 SDK, 开发者只需要调用 sdk 提供的初始化接口即可:

```
YoyoApi.init(Application app, String appKey, YoyoServerInterface<Boolean> iface);
```

- ◆ app 应用 Application 对象;
- ◆ appKey 可传空, 如果为空, 会读取 manifest 文件中配置的 appkey;

◆ iface 如果需要知道初始化成功与否，可传一个回调函数，不需要则传空；  
注意：只有 **YoyoApi.init 接口成功**后才可以进行后续的操作

## 9. 测试环境配置

打印日志，默认关闭，可通过调用如下方法开启：

```
YoyoApi.setDebugMode(boolean debugMode);
```

打开测试环境，默认是正式环境，测试环境服务器地址：<http://ready.sdk.xcyo.com/sdk>,可调用如下方法设置：

```
YoyoApi.setTestEnv(boolean testEnv );
```

## 10. 设置事件回调

```
setEventCallBack(YoyoEventInterface iface)
```

此方法用于设置房间内相应的点击事件回调。

◆ iface 事件回调接口，回调事件包括房间内登录（YoyoEventLogin）、充值（YoyoEventExchange）、分享（YoyoEventShare）。onEvent()方法可以得到当前的FragmentActivity，方便布局。

YoyoEventLogin，进行登录处理，包括布局与逻辑；

YoyoEventExchange，进行房间内的充值处理，包括布局与逻辑；

YoyoEventShare，进行房间内的分享处理，包括布局与逻辑。

范例：

```
YoyoApi.setEventCallBack(new YoyoEventInterface (){  
    @Override  
    public void onEvent(YoyoEvent event, FragmentActivity frag){  
        switch(event){  
            case YoyoEventLogin:  
                break;  
            case YoyoEventExchange:  
                break;  
            case YoyoEventShare:  
                break;  
        }  
    }  
});
```

## 11. 用户登录

在需要登录的地方调用以下接口：

```
loginYoyo(String openId, String thirdPartyToken, YoyoServerInterface<Boolean> iface);
```

- ◆ openId 用户唯一标识；
- ◆ thirdPartyToken 第三方访问令牌；
- ◆ Iface 回调接口，返回是否登录成功。

范例：

```
YoyoApi.loginYoyo("364031514","123456",new YoyoServerInterface<Boolean>() {  
    @Override  
    public boolean onOk(Boolean response) {  
        return true;  
    }  
  
    @Override  
    public boolean onError(YoyoErrorCode err, String msg) {  
        return true;  
    }  
});
```

注意事项：openId 与 thirdPartyToken 均不可为空。

## 12. 更新用户信息

可通过调用 `updateUserInfo(ArrayMap<String,String>map,YoyoServerInterface<Boolean> iface)` 来修改用户信息

- ◆ map 要修改的用户信息的 Map 集合
- ◆ Iface 回调接口，返回修改是否成功

范例：

```
YoyoApi.getShareData(map,new YoyoServerInterface<Boolean>() {  
    @Override  
    public boolean onOk(Boolean response) {  
        return true;  
    }  
});
```

```

    }

    @Override
    public boolean onError(YoyoErrorCode err, String msg) {
        return true;
    }
});

```

注意事项：map 不可为空；map 可用字段有 alias(用户昵称)、avatar(用户头像)、gender(用户性别)、birthday(用户生日[时间戳])、province(省份)、city(城市)、height(身高[cm])、weight(体重[kg])、bust(胸围)、waist(腰围)、hips(臀围)。

### 13. 用户退出

退出登录可以调用 `logoutYoyo(YoyoServerInterface<Boolean> iface);`

- ◆ iface 回调接口，返回是否退出成功。

范例：

```

YoyoApi.loginYoyo(new YoyoServerInterface<Boolean>() {
    @Override
    public boolean onOk(Boolean response) {
        return true;
    }

    @Override
    public boolean onError(YoyoErrorCode err, String msg) {
        return true;
    }
});

```

### 14. 获取房间列表

获取房间列表调用 `getHostList(YoyoServerInterface<YoyoHomeIndexRecord> iface);`

- ◆ iface 回调接口，请求成功时，返回房间列表信息。

范例：

```
YoyoApi.loginYoyo(new YoyoServerInterface<YoyoHomeIndexRecord>() {  
    @Override  
    public boolean onOk(YoyoHomeIndexRecord response) {  
        return true;  
    }  
  
    @Override  
    public boolean onError(YoyoErrorCode err, String msg) {  
        return true;  
    }  
});
```

## 15. 进入房间

进入某个房间可以通过调用 `YoyoApi.enterRoom(String roomId)`

- ◆ `roomId` 房间列表信息中对应的 `uid`

范例：

```
YoyoApi.enterRoom("12345");
```

注意事项：

- 1.在 `Init` 接口调用成功后使用；
- 2.`roomId` 不可为空。

## 16. 设置播放器

我们提供 `m3u8` 和 `rtmp` 两种视频源；

默认播放器是系统播放器，播放源是 `m3u8`；

另外还有我们自己的播放器，播放源是 `rtmp`，使用我们自己的播放器，安装包会相应的增加 5-6M 左右；推荐使用我们自己的播放器，播放速度快且流畅，系统播放器加载会比较慢，不推荐使用。

接入方也可根据自己需求定制自己的播放器，需继承 `YoyoBaseVideoView` 类并实现相应的方法，然后调用如下方法：

```
YoyoApi.setMediaPlayerClazz(Class<? extends YoyoBaseVideoView> cls)
```

- ◆ `cls` 接入方实现的 `YoyoBaseView` 类

## YoyoBaseView 类

```
public abstract class YoyoBaseVideoView {  
    public static String VIDEO_TYPE_RTMP = "rtmp";  
    public static String VIDEO_TYPE_M3U8 = "m3u8";  
    protected YoyoVideoListener mListener;  
  
    .....  
  
    public View getVideoView(){  
        return mVideoView;  
    }  
  
    //播放器支持 rtmp, 返回 VIDEO_TYPE_RTMP;  
    //反之, 返回 VIDEO_TYPE_M3U8  
    public String getVideoType(){  
        return VIDEO_TYPE_RTMP;  
    }  
  
    .....  
  
    //新建一个播放器 view 实例, 用来替换原始的播放器  
    protected abstract View initView(Context context);  
    public abstract void start(String url); //开始播放  
    public abstract void stop(); //停止播放  
    public abstract void destory(); //销毁的时候释放资源  
  
    .....  
}
```

## YoyoVideoListener 类

```
//YoyoBaseView 类持有一个 Listener 的实例,  
//监听播放器的大小变化, 播放出错, 播放完成等事件,  
//需要在设置播放器自有监听的事件里面调用该接口,  
//具体调用方法见 Demo 中 KSYVideoView.class  
  
public interface YoyoVideoListener {  
    public abstract void onSizeChanged(int width,int height);  
    public abstract void onError();  
    public abstract void onCompletion();  
}
```

## 17. 悠币充值

充值悠币前需要三方应用请求己方服务器生成交易订单号，然后再调用以下接口：

```
YoyoApi.exchange(int money, String orderId, String thirdToken, YoyoServerInterface<Boolean> iface)
```

- ◆ money 需要充值的金额，以人民币为单位;
- ◆ orderId 三方应用生成的订单号,
- ◆ thirdToken 第三方访问令牌
- ◆ iface 回调

范例：

```
YoyoApi.exchange(2215, "您在 YoYo 平台的交易单号", new YoyoServerInterface<Boolean>()
{
    @Override
    public boolean onOk(Boolean response) {
        Toast.makeText(MainActivity.this, “兑换成功”,
            Toast.LENGTH_SHORT).show();
        return false;
    }
    @Override
    public boolean onError(YoyoErrorCode err, String msg) {
        Toast.makeText(MainActivity.this, “兑换出错”,
            Toast.LENGTH_SHORT).show();
        return false;
    }
});
```

## 18. 分享

分享功能可视己方应用情况打开或者关闭，默认关闭，

可通过调用 `YoyoApi.setShareAble(Boolean shareAble)` 来打开

- ◆ shareAble true 为打开分享功能，false 为关闭分享功能

打开分享功能后，进入房间会有分享按钮，可通过调用 `getShareData(YoyoServerInterface iface)` 来获取分享需要的数据，



- ◆ iface 回调接口，返回对应房间的分享信息。

范例：

```
YoyoApi.getShareData(new YoyoServerInterface<YoyoShareRecord>() {  
    @Override  
    public boolean onOk(YoyoShareRecord response) {  
        return true;  
    }  
  
    @Override  
    public boolean onError(YoyoErrorCode err, String msg) {  
        return true;  
    }  
});
```

## 19. 错误码

错误码列表：

YoyoErrorCodeEmptyAppKey,	//appkey 为空
YoyoErrorCodeSDKNotInit,	//sdk 未初始化
YoyoErrorCodeInvalidToken,	//token 过期
YoyoErrorCodeInvalidUser,	//非法用户
YoyoErrorCodeInvalidRoomId,	//无效的房间号
YoyoErrorCodeErrorParams,	//参数错误
YoyoErrorCodeNeedLogin,	//需要登录
YoyoErrorCodeErrorNet,	//网络错误
YoyoErrorCodeSystemError,	//系统错误

## 20. 实体类

**YoyoSharedRecord:**

```
public class YoyoSharedRecord extends BaseRecord {  
    private String imgUrl;//宣传照地址  
    private String url;//直播间地址  
    private String alias;//主播昵称
```

```
public String getImgUrl() {  
    return imgUrl;  
}  
  
...  
}
```

### **YoyoHomeIndexRecord:**

```
public class YoyoHomeIndexRecord extends BaseRecord {  
    private List<YoyoIndexRecord> list;  
  
    public List<YoyoIndexRecord> getList() {  
        return list;  
    }  
  
    public void setList(List<YoyoIndexRecord> list) {  
        this.list = list;  
    }  
}  
  
public class YoyoIndexRecord extends BaseRecord {  
    private String uid;//用户 id  
    private String niceId;//靓号  
    private String cover;//房间封面  
    private String alias;//主播昵称  
    private String singerLevel;//主播等级  
    private String memberNum;//观众人数  
    private String title;//房间标题  
    private String isLive;//直播状态（0：未开播 1：开播）  
  
    public String getUid() {  
        return uid;  
    }  
}
```

```
public void setUid(String uid) {  
    this.uid = uid;  
}  
...  
}
```