

- 1 功能概述
- 2 使用演示
- 3 接口介绍
- 4 实现介绍
  - 4.1 初始化视图
    - 4.1.1 布局容器初始化
    - 4.1.2 布局具体内容初始化
  - 4.2 初始化数据
- 5 子目录介绍
  - 5.1 item
  - 5.2 widget

## 1 功能概述

互动学堂支持连麦功能，支持学生进入课堂自动连麦、学生举手申请连麦，也支持讲师直接邀请学生连麦，并支持多人连麦，丰富课堂内容。

学生举手连麦流程：学生举手申请连麦 -> 讲师端允许学生连麦 -> 学生正式加入连麦。

讲师邀请上麦流程：讲师选择一名学生邀请上麦 -> 学生点击确定立即上麦，或等待5秒后自动上麦

## 2 使用演示

互动学堂连麦功能以布局方式引入，并在初始化时设置相应参数，监听回调即可。

```
plvhcLinkmicLy = findViewById(R.id.plvhc_linkmic_ly);
// 初始化连麦布局
plvhcLinkmicLy.init(liveRoomDataManager);
// 初始化连麦的媒体配置
boolean isOpenMic = getIntent().getBooleanExtra(EXTRA_IS_OPEN_MIC, true);
boolean isOpenCamera = getIntent().getBooleanExtra(EXTRA_IS_OPEN_CAMERA, true);
boolean isFrontCamera = getIntent().getBooleanExtra(EXTRA_IS_FRONT_CAMERA, true);
plvhcLinkmicLy.muteAudio(!isOpenMic);
plvhcLinkmicLy.muteVideo(!isOpenCamera);
plvhcLinkmicLy.switchCamera(isFrontCamera);
// 设置布局回调 - 连麦
plvhcLinkmicLy.setOnViewActionListener(new IPLVHCLinkMicLayout.OnViewActionListener() {
    // 连麦布局回调方法 ...
});
```

详细的调用代码以及和其他模块的通信，请参考 `PLVHCLiveHiClassActivity` 类中的代码调用。

### 3 接口介绍

互动学堂连麦布局的接口定义为 `IPLVHCLinkMicLayout`，接口定义了可供外部直接调用的功能方法，以及需要外部响应的事件监听器。

1. 提供给外部的可直接调用功能方法

```
/**
 * 初始化
 *
 * @param liveRoomDataManager 直播间数据管理器
 */
void init(IPLVLiveRoomDataManager liveRoomDataManager);

/**
 * 设置view交互事件监听器
 *
 * @param listener 监听器
 */
void setOnViewActionListener(OnViewActionListener listener);

/**
 * 静音音频
 *
 * @param mute true表示静音, false表示打开
 */
void muteAudio(boolean mute);

/**
 * 禁用视频
 *
 * @param mute true表示禁用视频, false表示打开视频
 */
void muteVideo(boolean mute);

/**
 * 切换前后置摄像头方向
 */
void switchCamera(boolean front);

/**
 * 上课
 */
void startLesson(IPLVDataRequestListener<String> listener);

/**
 * 下课
 */
void stopLesson(IPLVDataRequestListener<String> listener);

/**
 * 发送举手事件
 *
 * @param raiseHandTime 举手时间
 */
void sendRaiseHandEvent(int raiseHandTime);

/**
 * 设置布局的可见性
 *
 * @param visibility One of {@link android.view.View#VISIBLE}, {@link android.view.View#INVISIBLE}, or {@link android.view.View#GONE}.
 */
```

```
*/  
void setVisibility(int visibility);  
  
/**  
 * 是否是课节上课状态  
 *  
 * @return true: 是课节上课状态, false: 不是课节上课状态  
 */  
boolean isLessonStarted();  
  
/**  
 * 获取连麦Presenter  
 *  
 * @return linkMicPresenter  
 */  
IPLVMultiRoleLinkMicContract.IMultiRoleLinkMicPresenter getLinkMicPresenter();  
  
/**  
 * 销毁, 释放资源  
 */  
void destroy();
```

## 2. 需要外部响应的事件监听器

```
/**
 * view交互事件监听器
 */
interface OnViewActionListener {
    /**
     * 布局大小变化
     */
    void onLayoutSizeChanged();

    /**
     * 用户举手变化
     *
     * @param raiseHandCount 举手数量
     * @param isRaiseHand 是否举手
     */
    void onUserRaiseHand(int raiseHandCount, boolean isRaiseHand);

    /**
     * 我的画笔权限变化
     *
     * @param isHasPaint 是否有画笔权限
     */
    void onHasPaintToMe(boolean isHasPaint);

    /**
     * 设置连麦readerView
     */
    void onSetupLinkMicRenderView(SurfaceView surfaceView, String linkMicId);

    /**
     * 学生获取奖杯
     *
     * @param userName 用户名称
     */
    void onGetCup(String userName);

    /**
     * 网络质量回调
     *
     * @param networkQuality {@link com.plv.linkmic.PLVLinkMicConstant.NetQuality}
     */
    void onNetworkQuality(int networkQuality);

    /**
     * 上行流量网络状态
     *
     * @param networkStatusVO
     */
    void onUpstreamNetworkStatus(PLVNetworkStatusVO networkStatusVO);

    /**
     * 远端连麦用户网络状态
     *
     * @param networkStatusVO
     */
}
```

```

void onRemoteNetworkStatus(PLVNetworkStatusVO networkStatusVO);

/**
 * 课节准备中
 */
void onLessonPreparing(long serverTime, long lessonStartTime);

/**
 * 课节开始
 */
void onLessonStarted();

/**
 * 课节结束
 */
void onLessonEnd(long inClassTime, boolean isTeacherType, boolean hasNextClass);
}

```

## 4 实现介绍

`IPLVHCLinkMicLayout` 的实现类是 `PLVHCLinkMicLayout`，讲师和学生共用同一个布局。

### 4.1 初始化视图

初始化视图分为两个阶段完成：第一阶段是布局容器初始化，第二阶段是布局具体内容初始化

#### 4.1.1 布局容器初始化

布局容器初始化在 `PLVHCLinkMicLayout` 构造时即开始执行，包括连麦视图插桩的添加、启动前台服务和保持屏幕常量等。

```

private void initView() {
    LayoutInflater.from(getContext()).inflate(R.layout.plvhc_linkmic_layout, this, true);

    classStopNoNextDialog = new PLVHCClassStopNoNextDialog(getContext());
    classStopHasNextDialog = new PLVHCClassStopHasNextDialog(getContext());

    // 启动前台服务，防止在后台被杀
    PLVForegroundService.startForegroundService(PLVHCLiveHiClassActivity.class, "POLYV互动学堂",
    R.drawable.plvhc_ic_launcher);
    // 防止自动息屏、锁屏
    setKeepScreenOn(true);
}

```

#### 4.1.2 布局具体内容初始化

布局具体内容初始化会在获取到频道限制的连麦人数后执行。

互动学堂支持 1v6 和 1v16 的连麦，这两种连麦方式的连麦视图排列方式不同，因此需要在确定频道连麦人数后才初始化具体的布局内容。

布局具体内容初始化包括连麦列表视图创建、视图尺寸调整、连麦视图回调监听设置，以及执行需要在初始化后执行的延时任务。

```
private void initLinkMicItemLayout(boolean isLittleLayout) {
    if (isLittleLayout) {
        ViewStub linkMicLyStub = findViewById(R.id.plvhc_linkmic_little_ly_stub);
        linkMicItemLayout = (IPLVHCLinkMicItemLayout) linkMicLyStub.inflate();
        initLayoutSize(true);
    } else {
        ViewStub linkMicLyStub = findViewById(R.id.plvhc_linkmic_large_ly_stub);
        linkMicItemLayout = (IPLVHCLinkMicItemLayout) linkMicLyStub.inflate();
        initLayoutSize(false);
    }
    if (linkMicItemLayout != null) {
        linkMicItemLayout.setOnViewActionListener(new IPLVHCLinkMicItemLayout.OnViewActionListener() {
            @Override
            public void onClickItemView(final int position, final PLVLinkMicItemDataBean linkMicItemDataBean) {
                // 连麦视图点击回调 ...
            }
        });
        linkMicItemLayout.setOnRenderViewCallback(new PLVHCLinkMicItemView.OnRenderViewCallback() {
            // 连麦渲染视图的回调监听 ...
        });
        // 延时任务执行
        for (Runnable task : initiatedTasks) {
            task.run();
        }
        initiatedTasks.clear();
    }
}
```

## 4.2 初始化数据

初始化数据包括数据管理器 `IPLVLiveRoomDataManager` 的接收保存、课节详情数据的监听、Presenter `IMultiRoleLinkMicPresenter` 的注册和初始化。

```
@Override
public void init(IPLVLiveRoomDataManager liveRoomDataManager) {
    this.liveRoomDataManager = liveRoomDataManager;
    observeLessonDataBean();

    linkMicPresenter = new PLVMultiRoleLinkMicPresenter(liveRoomDataManager);
    linkMicPresenter.registerView(linkMicView);
    linkMicPresenter.init();
}
```

连麦以 mvp 模式实现，具体交互逻辑在 `linkMicView` 内实现，详情可以参考代码块 **连麦 - MVP模式的view层实现**。

## 5 子目录介绍

### 5.1 item

`item` 目录存放连麦列表布局视图和连麦视图。

`IPLVHCLinkMicItemLayout` 为连麦列表布局视图对外提供接口的定义。

`PLVHCabsLinkMicItemLayout` 为连麦列表布局视图的父类，具体提供了多种连麦列表布局的通用方法实现。

`PLVHCLinkMicItemLittleLayout` 为 1v6 连麦模式下的布局视图，`PLVHCLinkMicItemLargeLayout` 为 1v16 连麦模式下的布局视图。

`PLVHCLinkMicItemView` 为单个连麦渲染视图。

### 5.2 widget

`widget` 目录存放连麦功能特有的视图控件，包括学员被邀请上麦的倒计时弹窗。