

- [1 功能概述](#)
- [2 使用演示](#)
- [3 接口介绍](#)
 - [3.1 IPLVLCLinkMicControlBar](#)
 - [3.2 IPLVLCLinkMicLayout](#)
- [4 实现介绍](#)
 - [4.1 PLVLCLinkMicControlBar](#)
 - [4.2 PLVLCLinkMicLayout](#)
 - [4.2.1 初始化View](#)
 - [4.2.2 初始化数据](#)
- [5 子目录介绍](#)
 - [5.1 adapter目录](#)
 - [5.2 service目录](#)
 - [5.3 widget目录](#)

1 功能概述

保利威云课堂sdk支持连麦功能，举手连麦、音频连麦、视频连麦可自由选择，并支持多人连麦，丰富课堂内容。

连麦流程：讲师端发起连麦->学员举手申请连麦->讲师端允许学员连麦->学员正式加入连麦。

2 使用演示

```
//初始化
ViewStub linkmicLayoutViewStub = findViewById(R.id.plvlc_linkmic_layout);
IPLVLCLinkMicLayout linkMicLayout = (IPLVLCLinkMicLayout) linkmicLayoutViewStub.inflate();
linkMicLayout.init(liveRoomDataManager, linkMicControlBar);
linkMicLayout.hideAll();
//设置连麦布局监听器
linkMicLayout.setOnPLVLinkMicLayoutListener();
```

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

3 接口介绍

3.1 IPLVLCLinkMicControlBar

云课堂场景下，针对 连麦控制条布局 进行封装的 Interface，在[PLVLinkMicLayout]中被使用，用于将最新的连麦状态更新到控制条。定义了：

1. 外部直接调用的方法
2. 需要外部响应的事件监听器

```
public interface IPLVLCLinkMicControlBar {

    // <editor-fold defaultstate="collapsed" desc="1. 外部直接调用的方法">

    /**
     * 设置监听器
     *
     * @param onPLCLinkMicControlBarListener listener
     */
    void setOnPLCLinkMicControlBarListener(OnPLCLinkMicControlBarListener onPLCLinkMicControlBarListener);

    /**
     * 设置讲师开启或关闭连麦
     *
     * @param isTeacherOpenLinkMic true表示讲师开启连麦
     */
    void setIsTeacherOpenLinkMic(boolean isTeacherOpenLinkMic);

    /**
     * 设置开关摄像头
     *
     * @param toOpen true表示要打开摄像头
     */
    void setCameraOpenOrClose(boolean toOpen);

    /**
     * 开关麦克风
     *
     * @param toOpen true表示要打开麦克风
     */
    void setMicrophoneOpenOrClose(boolean toOpen);

    /**
     * 设置上麦成功
     */
    void setJoinLinkMicSuccess();

    /**
     * 设置离开连麦
     */
    void setLeaveLinkMic();

    /**
     * 设置是音频连麦还是视频连麦
     *
     * @param isAudio true表示是音频连麦, false表示是视频连麦
     */
    void setIsAudio(boolean isAudio);

    /**
     * 显示
     */
    void show();
}
```

```

/**
 * 隐藏
 */
void hide();
// </editor-fold>

// <editor-fold defaultstate="collapsed" desc="2. 需要外部响应的事件监听器">
interface OnPLCLinkMicControlBarListener {
    /**
     * 点击请求上麦
     */
    void onClickRingUpLinkMic();

    /**
     * 点击下麦
     */
    void onClickRingOffLinkMic();

    /**
     * 点击开关摄像头
     *
     * @param toClose true表示关闭。false表示开启
     */
    void onClickCameraOpenOrClose(boolean toClose);

    /**
     * 点击前后置摄像头
     *
     * @param toFront true表示前置，false表示后置
     */
    void onClickCameraFrontOfBack(boolean toFront);

    /**
     * 点击开关麦克风
     *
     * @param toClose true表示关闭。false表示开启
     */
    void onClickMicroPhoneOpenOrClose(boolean toClose);
}
// </editor-fold>
}

```

3.2 IPLVLCLinkMicLayout

云课堂场景下，针对 连麦布局 进行封装的 Interface。定义了：

1. 云课堂使用的连麦布局接口
2. 需要外部响应的事件监听器

```
public interface IPLVLLinkMicLayout {

    // <editor-fold defaultstate="collapsed" desc="1. 外部直接调用的方法">

    /**
     * 初始化, 设置数据
     *
     * @param liveRoomDataManager 频道数据
     */
    void init(IPLVLiveRoomDataManager liveRoomDataManager, IPLVLLinkMicControlBar linkMicControlBar);

    /**
     * 销毁
     */
    void destroy();

    /**
     * 显示
     */
    void showAll();

    /**
     * 隐藏
     */
    void hideAll();

    /**
     * 隐藏连麦列表
     */
    void hideLinkMicList();

    /**
     * 显示连麦列表
     */
    void showLinkMicList();

    /**
     * 隐藏连麦控制条
     */
    void hideControlBar();

    /**
     * 显示连麦控制条
     */
    void showControlBar();

    /**
     * 设置是否讲师打开连麦
     *
     * @param isTeacherOpenLinkMic 讲师是否打开连麦
     */
    void setIsTeacherOpenLinkMic(boolean isTeacherOpenLinkMic);

    /**
```

```

    * 设置连麦类型
    *
    * @param isAudioLinkMic true表示是音频连麦, false表示是视频连麦
    */
void setIsAudio(boolean isAudioLinkMic);

/**
 * 是否已经加入连麦
 *
 * @return true表示已经加入了连麦, false表示没有加入连麦
 */
boolean isJoinLinkMic();

/**
 * Media是否被切换到连麦列表了
 *
 * @return true表示media在连麦列表, false表示media不在连麦列表
 */
boolean isMediaShowInLinkMicList();

/**
 * 设置连麦布局监听器
 *
 * @param onPLVLinkMicLayoutListener 连麦布局监听器
 */
void setOnPLVLinkMicLayoutListener(OnPLVLinkMicLayoutListener onPLVLinkMicLayoutListener);

/**
 * 获取横屏时连麦布局的宽度
 *
 * @return 横屏宽
 */
int getLandscapeWidth();
// </editor-fold>

// <editor-fold defaultstate="collapsed" desc="2. 需要外部响应的事件监听器">

/**
 * 连麦布局监听器
 */
interface OnPLVLinkMicLayoutListener {
    //todo 未来将支持画笔事件

    // <editor-fold defaultstate="collapsed" desc="画笔事件监听">
    //      //点击擦除PPT画笔
    //      void onUpdateEraseStatus(boolean toErase);
    //
    //      //更新画笔颜色
    //      void onUpdateBrushColor(String color);
    //
    //      //更新画笔状态
    //      boolean onUpdateBrushStatus(boolean unSelected);
    //      // </editor-fold>

    // <editor-fold defaultstate="collapsed" desc="连麦事件监听">

```

```

/**
 * 加入连麦成功
 */
void onJoinChannelSuccess();

/**
 * 离开连麦
 */
void onLeaveChannel();
// </editor-fold>

// <editor-fold defaultstate="collapsed" desc="点击事件监听器">

/**
 * 切换连麦连麦中讲师的位置，因为纯视频的频道连麦时，讲师默认在主屏
 *
 * @param viewSwitcher switchView的切换器
 * @param switchView 讲师连麦Item的switch view
 */
void onChangeTeacherLocation(PLVViewSwitcher viewSwitcher, PLVSwitchViewAnchorLayout switchView);

/**
 * 点击连麦Item与media切换，只切换一次。
 * 当点击连麦列表item时，media在主屏幕，或刚好点击的item就是media，那么此时只发生一次切换，即media和连麦Item切
换。
 *
 * @param switchView 连麦Item的switch view
 */
void onClickSwitchWithMediaOnce(PLVSwitchViewAnchorLayout switchView);

/**
 * 点击连麦Item与media切换，切换两次。
 * 当点击连麦列表item时，media在连麦列表中的其他item，那么要发生两次切换：
 * 1. 先将media和此时主屏幕的那个连麦item进行切换，回到初始默认的位置。
 * 2. 再将此次点击的连麦item与主屏幕的media进行切换。
 *
 * @param switchViewHasMedia 连麦列表的上有media的那个switch view
 * @param switchViewGoMainScreen 被点击的item的switch view，对应将会被切换到主屏幕的连麦item
 */
void onClickSwitchWithMediaTwice(PLVSwitchViewAnchorLayout switchViewHasMedia, PLVSwitchViewAnchorLayout
switchViewGoMainScreen);
// </editor-fold>

}
// </editor-fold>
}

```

4 实现介绍

IPLVLCLinkMicControlBar的实现类是PLVLCLinkMicControlBar。

IPLVLCLinkMicLayout的实现类是PLVLCLinkMicLayout。

4.1 PLVLCLinkMicControlBar

PLVLCLinkMicControlBar是连麦控制条的实现。有如下几点需要注意：

1. 连麦控制条在Activity中被创建，并作为参数传递到 `PLVLCLinkMicLayout` 中，由连麦布局响应控制条的点击事件以及刷新连麦控制条的UI状态。
2. 在竖屏和横屏上分别用两套View来实现竖屏和横屏的UI。
3. 主要的逻辑集中在动画、点击事件响应、和横竖屏两套UI保持同步。

4.2 PLVLCLinkMicLayout

连麦布局是连麦功能模块的主体，主要的UI是连麦列表。且横竖屏共用同一套连麦列表对象。

4.2.1 初始化View

在初始化时，要初始化连麦列表RecyclerView和连麦列表用到的适配器PLVLinkMicListAdapter。

```
private void initView() {  
    //...  
    //init RecyclerView  
    rvLinkMicList.setLayoutManager(new LinearLayoutManager(getContext(), RecyclerView.HORIZONTAL, false));  
    rvLinkMicList.addItemDecoration(landscapeItemDecoration);  
    //init adapter  
    linkMicListAdapter = new PLVLinkMicListAdapter(...)  
    //init方向  
    curIsLandscape = PLVScreenUtils.isLandscape(getContext());  
}
```

4.2.2 初始化数据

数据的初始化在对外API方法中：

```
@Override  
public void init(IPLVLiveRoomDataManager liveRoomDataManager, IPLVLCLinkMicControlBar linkMicControlBar) {  
    linkMicPresenter = new PLVLinkMicPresenter(liveRoomDataManager, this);  
    initLinkMicControlBar(linkMicControlBar);  
}
```

由于连麦功能模块使用了mvp模式，因此首先初始化了Presenter。

此外，还初始化了连麦控制条，对连麦控制条的点击事件进行了监听。

连麦布局的主体逻辑主要集中在mvp模式的View层接口的实现中，可以参考该接口的注释和实现：


```

interface IPLVLinkMicView {
    /**
     * 响应连麦错误
     */
    void onLinkMicError(int errorCode, Throwable throwable);
    /**
     * 响应讲师开启连麦
     */
    void onTeacherOpenLinkMic();
    /**
     * 响应讲师关闭连麦
     */
    void onTeacherCloseLinkMic();
    /**
     * 响应讲师允许连麦
     */
    void onTeacherAllowJoin();
    /**
     * 响应加入连麦频道超时
     */
    void onAllowButJoinTimeout();
    /**
     * 响应在加入频道之前，View层应创建连麦适配器和初始化连麦布局
     *
     * @param linkMicUid 我的连麦Id
     * @param isAudio 是否是音频连麦
     * @param linkMicList 连麦列表。在实现中，要用该列表去渲染
     */
    void onBeforeJoinChannel(String linkMicUid, boolean isAudio, List<PLVLinkMicItemDataBean> linkMicList);
    /**
     * 响应加入连麦频道成功
     */
    void onJoinChannelSuccess();
    /**
     * 响应离开连麦频道
     */
    void onLeaveChannel();
    /**
     * 响应用户加入连麦频道
     *
     * @param uids
     */
    void onUsersJoin(List<String> uids);
    /**
     * 响应用户离开连麦频道
     *
     * @param uids
     */
    void onUsersLeave(List<String> uids);
    /**
     * 回调我当前不在连麦列表
     */
    void onNotInLinkMicList();
    /**
     * 响应用户开关视频

```

```

*
* @param uid 用户id
* @param mute true表示关闭视频, false表示开启视频
* @param pos 列表中的位置
*/
void onUserMuteVideo(final String uid, final boolean mute, int pos);
/**
* 响应用户开关音频
*
* @param uid 用户id
* @param mute true表示关闭音频, false表示开启音频
* @param pos 列表中的位置
*/
void onUserMuteAudio(final String uid, final boolean mute, int pos);
/**
* 响应本地用户麦克风音量变化
*/
void onLocalUserMicVolumeChanged();
/**
* 响应远端用户麦克风音量变化
*/
void onRemoteUserVolumeChanged(List<PLVLinkMicItemDataBean> linkMicList);
/**
* 切换第一画面
*
* @param linkMicId 新的第一画面的连麦Id
*/
void onSwitchFirstScreen(String linkMicId);
/**
* 设置第一画面的连麦ID
*
* @param linkMicId 新的第一画面的连麦Id
*/
void setFirstScreenLinkMicId(String linkMicId);
/**
* 隐藏连麦列表中的讲师item, 并根据isNeedSwitchToMain参数决定是否要把该item的view切换到主屏
*
* @param linkMicId 讲师的连麦id
* @param teacherPos 讲师item在当前连麦列表中的索引
* @param isNeedSwitchToMain 是否要切换到主屏
*/
void onAdjustTeacherLocation(String linkMicId, int teacherPos, boolean isNeedSwitchToMain);
/**
* 切换PPT View和第一画面的位置
*
* @param toMainScreen true表示切换到主屏幕, false表示切回到悬浮窗
*/
void onSwitchPPTViewLocation(boolean toMainScreen);
/**
* PPT是否被切换到连麦列表了
*
* @return true表示PPT在连麦列表, false表示PPT不在连麦列表
*/
boolean isMediaShowInLinkMicList();
/**
* 获取PPTView在连麦列表中的位置index

```

```

    *
    * @return ppt在连麦列表中的位置
    */
    int getMediaViewIndexInLinkMicList();
    /**
    * 点击连麦列表[index]位置上的画面
    *
    * @param index 连麦列表中的位置
    */
    void performClickInLinkMicListItem(int index);
    /**
    * 更新所有整个连麦列表
    */
    void updateAllLinkMicList();
}

```

5 子目录介绍

5.1 adapter目录

1. PLVLinkMicListAdapter

连麦列表RecyclerView的适配器，该类中对具体的每个连麦Item做了处理，并暴露了连麦Item的点击事件。

5.2 service目录

1. PLVLCLinkMicForegroundService

在连麦时开启前台服务来确保在后台连麦时，连麦功能正常。

5.3 widget目录

这是连麦模块的自定义View目录

1. PLVLCLinkMicRingButton

连麦话筒按钮

2. PLVLinkMicRvLandscapeItemDecoration

横屏连麦列表的ItemDecoration