

- 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view方法
 - 4.2 初始化数据方法
- 5 子目录介绍
 - 5.1 enums 目录
 - 5.2 widget 目录

1 功能概述

工具栏模块是互动学堂的基础功能之一，分为讲师视图和学生视图。该模块支持对摄像头、麦克风、画笔等工具的参数的设置。工具栏模块可以增加课堂的互动方式，促进学员与讲师之间的更加高效的沟通。

2 使用演示

在demo中的 PLVHCLiveHiClassActivity 界面中对该模块的使用介绍如下。

```
// 工具栏布局
private IPLVHCToolBarLayout plvhcToolBarLy;

//findView()
plvhcToolBarLy = findViewById(R.id.plvhc_tool_bar_ly);

// 初始化工具栏布局,initView()
plvhcToolBarLy.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);
//...
plvhcToolBarLy.initDefaultMediaStatus(!isOpenMic, !isOpenCamera, isFrontCamera);
```

3 接口介绍

在互动学堂场景下，IPLVHCToolBarLayout 针对工具栏布局进行接口定义，主要包含：

- 1、外部直接调用的方法
- 2、需要外部响应的交互事件监听器

```
public interface IPLVHCToolBarLayout {

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

    /**
     * 处理图片选择结果
     *
     * @param data 数据
     */
    void handleImgSelectResult(Intent data);

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

    /**
     * 初始化默认的媒体状态
     *
     * @param isMuteAudio 是否禁用麦克风
     * @param isMuteVideo 是否禁用摄像头
     * @param isFrontCamera 是否是前置摄像头
     */
    void initDefaultMediaStatus(boolean isMuteAudio, boolean isMuteVideo, boolean isFrontCamera);

    /**
     * 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);

    /**
     * 调整布局
     */
    void adjustLayout();

    /**
     * 接收用户举手变化
     *
     */
}
```

```

    * @param raiseHandCount 举手数量
    * @param isRaiseHand    是否举手
    */
void acceptUserRaiseHand(int raiseHandCount, boolean isRaiseHand);

/**
 * 接收我的画笔权限变化
 *
 * @param isHasPaint
 */
void acceptHasPaintToMe(boolean isHasPaint);

/**
 * 更新标注工具按钮状态
 *
 * @param showUndoButton 是否显示撤销按钮
 * @param showDeleteButton 是否显示删除按钮
 */
void changeMarkToolState(boolean showUndoButton, boolean showDeleteButton);

/**
 * 获取成员列表布局的连麦View
 *
 * @return linkMicView
 */
IPLVMultiRoleLinkMicContract.IMultiRoleLinkMicView getMemberLayoutLinkMicView();

/**
 * 获取设置布局的连麦View
 *
 * @return linkMicView
 */
IPLVMultiRoleLinkMicContract.IMultiRoleLinkMicView getSettingLayoutLinkMicView();

/**
 * 是否拦截返回事件
 *
 * @return true: 拦截, false: 不拦截
 */
boolean onBackPressed();

/**
 * 销毁, 释放资源
 */
void destroy();

/**
 * view交互事件监听器
 */
interface OnViewActionListener {
    /**
     * 发送举手事件
     *
     * @param raiseHandTime 举手时间
     */
    void onSendRaiseHandEvent(int raiseHandTime);
}

```

```

/**
 * 全屏控制
 *
 * @param isFullScreen true: 全屏, false: 退出全屏
 */
void onFullScreenControl(boolean isFullScreen);

/**
 * 初始化上课下课按钮后回调
 *
 * @param classImageView 上下课按钮
 */
void onInitClassImageView(View classImageView);

/**
 * 上课
 *
 * @param listener 监听器
 */
void onStartLesson(IPLVDataRequestListener<String> listener);

/**
 * 下课
 *
 * @param listener 监听器
 */
void onStopLesson(IPLVDataRequestListener<String> listener);

/**
 * 切换标注工具
 *
 * @param newMarkTool 新的工具
 */
void onRequestChangeDocumentMarkTool(PLVHCToolEnums.MarkTool newMarkTool);

/**
 * 切换标注工具颜色
 *
 * @param newColor 新的颜色
 */
void onRequestChangeDocumentColor(PLVHCToolEnums.Color newColor);

/**
 * 撤销画笔操作
 */
void onRequestUndo();

/**
 * 删除标注内容
 */
void onRequestDelete();
}
}

```

4 实现介绍

在 `PLVHCToolBarLayout` 工具栏布局里面，实现了 `IPLVHCToolBarLayout` 的接口。

该布局包含的元素有：聊天室、成员列表、设置、文档管理等相关界面的布局。

下面介绍该布局中涉及到的主要方法。

4.1 初始化view方法

`PLVHCToolBarLayout` 继承于 `FramLayout`，在构造器中使用 `initView()` 方法对view进行初始化处理。

```
//构造器
public PLVHCToolBarLayout(@NonNull Context context, @Nullable AttributeSet attrs, int defStyleAttr) {
    super(context, attrs, defStyleAttr);
    initView();
}

//initView()方法
private void initView() {
    LayoutInflater.from(getContext()).inflate(R.layout.plvhc_toolbar_layout, this);

    initChatroomLayout();
    initMemberLayout();
    initSettingLayout();
    initPptListLayout();
}
```

4.2 初始化数据方法

由于用户类型有两种——讲师和学生，因此在初始化数据的时候需要判断用户类型。方法在对外API中。除此之外，这里还初始化了标注工具控制条布局。

```
//PLVHCToolBarLayout.java
@Override
public void init(IPLVLiveRoomDataManager liveRoomDataManager) {
    String userType = liveRoomDataManager.getConfig().getUser().getViewerType();
    if (PLVSocketUserConstant.USER_TYPE_TEACHER.equals(userType)) {
        initTeacherLayout();
    } else {
        initStudentLayout();
    }

    chatroomLayout.init(liveRoomDataManager);
    //register after init
    chatroomLayout.getChatroomPresenter().registerView(memberLayout.getChatroomView());
    chatroomLayout.getChatroomPresenter().requestKickUsers();

    plvhcToolbarMarkToolControllerLayout.init(liveRoomDataManager);
}
```

在外部API中实现了 `IPLVHCToolBarLayout` 的接口。这里就不一一列举，仅列举部分作为参考。

```
//PLVHCToolBarLayout.java
@Override
public void onLessonPreparing(long serverTime, long lessonStartTime) {
    if (plvhcToolbarClassIv != null) {
        plvhcToolbarClassIv.setSelected(false);
        isClassButtonEnable = true;
    }
    if (plvhcToolbarStudentHandsUpLy != null) {
        plvhcToolbarStudentHandsUpLy.setVisibility(View.INVISIBLE);
    }
    chatroomLayout.onLessonPreparing(serverTime, lessonStartTime);
}
//.....
```

5 子目录介绍

5.1 enums 目录

这里面枚举了标注工具用到的一些常量，包括标注工具显示、标注工具类型以及标注工具颜色。

	具体枚举数据
标注工具显示	隐藏、显示标注工具列表、显示标注工具颜色
标注工具类型	画板移动工具、画板选区工具、铅笔(自由画线)工具、箭头工具、文本工具、橡皮擦工具、清屏
标注工具颜色	红、蓝、绿、黄、黑、白

5.2 widget 目录

这是标注工具自定义的控制布局。包括对标注工具的初始化、显示、隐藏以及点击事件响应等。