

- [1 功能概述](#)
- [2 使用演示](#)
- [3 实现介绍](#)
 - [3.1 PLVLCChatFragment](#)
 - [3.2 PLVLCChatLandscapeLayout](#)
 - [3.3 PLVLCQuizFragment](#)

1 功能概述

聊天室模块包括发言、提问、点赞、历史记录、欢迎语等功能。聊天室模块在UI、交互、功能上，相较于其他模块，都会更复杂更庞大。因此设计、搭建、维护一个聊天室，也是一件较为费时费力的事情。我们推荐直接使用保利威封装好的聊天室模块，该部分代码完全开源，支持直接使用，以及二次开发。

2 使用演示

聊天室模块的UI布局在云课堂场景分为三个部分组成，在页面菜单包括聊天tab、提问tab两个部分，横屏的播放器区域也包括聊天layout部分。以下实例为页面菜单中添加聊天tab，代码如下：

```
//聊天室mvp-presenter
private IPLVChatroomContract.IChatroomPresenter chatroomPresenter;

private void addChatTab(PolyvLiveClassDetailV0.DataBean.ChannelMenusBean channelMenusBean) {
    //页面菜单添加聊天tab的标题
    pageMenuTabTitleList.add(channelMenusBean.getName());
    //聊天fragment
    chatFragment = new PLVLCChatFragment();
    //初始化聊天列表
    chatFragment.init(chatCommonMessageList);
    //配置聊天页面所需的数据
    chatFragment.setIsLiveType(liveRoomDataManager.getConfig().isLive());
    //配置聊天页面的UI交互监听器
    chatFragment.setOnViewActionListener(new PLVLCChatFragment.OnViewActionListener() {
        @Override
        public void onShowBulletinAction() {
            //...
        }
    });
    //聊天室mvp-presenter注册聊天室mvp-view
    chatroomPresenter.registerView(chatFragment.getChatroomView());
    //页面菜单添加聊天tab
    pageMenuTabFragmentList.add(chatFragment);
}
```

3 实现介绍

3.1 PLVLCChatFragment

云课堂场景下的互动聊天tab页，包含聊天信息列表、发送信息输入框、表情布局、点赞布局、欢迎语、公告等元素。该类和聊天室mvp-presenter的通信，是通过该类中定义的聊天室mvp-view，当调用聊天室mvp-presneter的registerView方法后，聊天室mvp-presneter即可通知聊天室mvp-view的更新，同时聊天室mvp-view也能调用调用聊天室mvp-presenter的方法。定义聊天室mvp-view的示例代码：

```

//聊天室mvp-presenter
private IPLVChatroomContract.IChatroomPresenter chatroomPresenter;

//聊天室mvp-view
private IPLVChatroomContract.IChatroomView chatroomView = new PLVAbsChatroomView() {
    @Override
    public void setPresenter(@NonNull IPLVChatroomContract.IChatroomPresenter presenter) {
        super.setPresenter(presenter);
        chatroomPresenter = presenter;
    }

    @Override
    public void onSpeakEvent(@NonNull PLVSpeakEvent speakEvent) {
        super.onSpeakEvent(speakEvent);
    }

    @Override
    public void onImgEvent(@NonNull PLVChatImgEvent chatImgEvent) {
        super.onImgEvent(chatImgEvent);
    }

    @Override
    public void onLikesEvent(@NonNull PLVLikesEvent likesEvent) {
        super.onLikesEvent(likesEvent);
    }

    @Override
    public void onLoginEvent(@NonNull PLVLoginEvent loginEvent) {
        super.onLoginEvent(loginEvent);
    }

    @Override
    public void onLogoutEvent(@NonNull PLVLogoutEvent logoutEvent) {
        super.onLogoutEvent(logoutEvent);
    }

    @Override
    public void onBulletinEvent(@NonNull PolyvBulletinVO bulletinVO) {
        super.onBulletinEvent(bulletinVO);
    }

    @Override
    public void onRemoveBulletinEvent() {
        super.onRemoveBulletinEvent();
    }

    @Override
    public void onCloseRoomEvent(@NonNull final PLVCloseRoomEvent closeRoomEvent) {
        super.onCloseRoomEvent(closeRoomEvent);
    }

    @Override
    public void onRemoveMessageEvent(@Nullable String id, boolean isRemoveAll) {
        super.onRemoveMessageEvent(id, isRemoveAll);
    }
}

```

```

    @Override
    public void onCustomGiftEvent(@NonNull PolyvCustomEvent.UserBean userBean, @NonNull PLVCustomGiftBean
customGiftBean) {
        super.onCustomGiftEvent(userBean, customGiftBean);
    }

    @Override
    public void onLocalSpeakMessage(@Nullable PolyvLocalMessage localMessage) {
        super.onLocalSpeakMessage(localMessage);
    }

    @Override
    public void onLocalImageMessage(@Nullable PolyvSendLocalImgEvent localImgEvent) {
        super.onLocalImageMessage(localImgEvent);
    }

    @Override
    public void onSpeakImgDataList(List<PLVBaseViewData> chatMessageDataList) {
        super.onSpeakImgDataList(chatMessageDataList);
    }

    @Override
    public void onHistoryDataList(List<PLVBaseViewData<PLVBaseEvent>> chatMessageDataList, int
requestSuccessTime, boolean isNoMoreHistory, int viewIndex) {
        super.onHistoryDataList(chatMessageDataList, requestSuccessTime, isNoMoreHistory, viewIndex);
    }

    @Override
    public void onHistoryRequestFailed(String errorMsg, Throwable t, int viewIndex) {
        super.onHistoryRequestFailed(errorMsg, t, viewIndex);
    }
}
};

```

3.2 PLVLCChatLandscapeLayout

云课堂场景下的横屏聊天布局，该类和 `PLVLCChatFragment` 一样，都能显示聊天列表信息、加载历史记录，是在横屏时显示的一个聊天区域。该类中也定义了聊天室mvp-view，负责与聊天室mvp-presenter的通信。

3.3 PLVLCQuizFragment

云课堂场景下的咨询提问tab页，主要是用于向讲师提问题的聊天页面，这个页面发送的信息只有讲师能够看到。该类中也定义了聊天室mvp-view，负责与聊天室mvp-presenter的通信。