

- 1 功能概述
- 2 核心类介绍
 - 2.1 PLVLSChatroomViewModel
 - 2.1.1 生命周期
 - 2.1.2 获取聊天记录
 - 2.1.3 发送消息与消息回复
 - 2.1.4 消息数组
 - 2.1.5 监听与回调

1 功能概述

聊天室模块包括发言、历史记录查看、消息回复、踢人、禁言等功能。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 核心类介绍

2.1 PLVLSChatroomViewModel

`PLVLSChatroomViewModel` 是手机开播场景 Scene 层聊天室的核心类，由于一个应用只允许存在一个聊天室，所以 `PLVLSChatroomViewModel` 采用的是单例的模式，负责创建、持有、销毁 Common 层聊天室核心类 `PLVChatroomPresenter` 的实例对象以及与 Common 层聊天室模块进行通信，核心类

`PLVLSChatroomViewModel` 提供功能如下：

1. 对 View 层提供发送消息的接口；
2. 管理 Common 层返回的消息模型；
3. 在 View 层需要刷新 UI、更新列表数据时，通过回调通知 View 层。

2.1.1 生命周期

`PLVLSChatroomViewModel` 是单例类，进入直播间启动聊天室的时候调用 `- setup` 方法，离开直播间时调用 `- clear` 方法，代码如下：

```
// 使用新的直播间数据启动聊天室管理器
[[PLVLSChatroomViewModel sharedViewModel] setup];
// 退出前调用，用于资源释放、状态位清零
[[PLVLSChatroomViewModel sharedViewModel] clear];
```

代码注释提到的“直播间数据”指的是 `PLVRoomDataManager` 单例持有的 `roomData`。

2.1.2 获取聊天记录

Common 层聊天室核心类 `PLVChatroomPresenter` 在初始化完毕之后，会自动获取第一页的聊天记录，默认一页的消息条数为 20，获取更多聊天记录代码如下：

```
[[PLVLChatroomViewModel sharedViewModel] loadHistory];
```

2.1.3 发送消息与消息回复

`PLVLChatroomViewModel` 类提供手机开播场景所需的发送消息的API，具体接口定义如下：

```
/// 发送文本消息
/// @param content 消息文本
/// @param replyChatModel 回复消息模型（非回复消息该字段为nil）
/// @return YES表示数据将有更新，可等待收到回调后刷新列表；NO表示socket未登录或房间关闭，可进行toast提示
- (BOOL)sendSpeakMessage:(NSString *)content replyChatModel:(PLVChatModel * _Nullable)replyChatModel;

/// 发送图片消息
/// @param image 图片
/// @return YES表示数据将有更新，可等待收到回调后刷新列表；NO表示socket未登录或房间关闭，可进行toast提示
- (BOOL)sendImageMessage:(UIImage *)image;
```

2.1.4 消息数组

`PLVLChatroomViewModel` 类提供以下消息列表数组，作为聊天室 View 层的数据源：

```
/// 全部消息数组
@property (nonatomic, strong, readonly) NSMutableArray <PLVChatModel *> *chatArray;
```

2.1.5 监听与回调

`PLVLChatroomViewModel` 提供了属性 `delegate` 用于设置回调监听，属性 `delegate` 遵循协议 `PLVLChatroomViewModelProtocol`，用于在 View 层需要刷新 UI、更新列表数据时，通过回调通知 View 层：

```
@protocol PLVLSChatroomViewModelProtocol <NSObject>
```

```
@optional
```

```
/// 返回本地发送的公聊消息（包含禁言的情况）
```

```
/// 用于刷新列表、滚动列表到底部
```

```
- (void)chatroomViewModel_didSendMessage;
```

```
/// 返回socket接收到的公聊消息
```

```
/// 用于刷新列表、显示新消息提示
```

```
- (void)chatroomViewModel_didReceiveMessages;
```

```
/// socket通知有消息被删除（1条或多条）
```

```
/// 用于刷新列表
```

```
- (void)chatroomViewModel_didMessageDeleted;
```

```
/// 获取历史聊天记录成功时触发
```

```
/// 用于刷新列表，停止【下拉加载更多】控件的动画
```

```
/// @param noMore 是否还有更多历史消息，YES表示已加载完，此时可隐藏【下拉加载更多】控件
```

```
/// @param first 是否是初次加载历史消息，初次加载需滚动列表到底部
```

```
- (void)chatroomViewModel_loadHistorySuccess:(BOOL)noMore firstTime:(BOOL)first;
```

```
/// 获取历史聊天消息失败时触发
```

```
/// 用于停止【下拉加载更多】控件的动画
```

```
- (void)chatroomViewModel_loadHistoryFailure;
```

```
@end
```