目录

[1. 引言 1](#_Toc44489851)

[1.1 设计目的 1](#_Toc44489852)

[1.2 设计内容 1](#_Toc44489853)

[1.3 开发环境 1](#_Toc44489854)

[2. 系统分析与设计 2](#_Toc44489855)

[2.1 需求分析 2](#_Toc44489856)

[2.2 软件架构设计 2](#_Toc44489857)

[3. 各功能模块的详细设计 2](#_Toc44489858)

[3.1 基于websocket的即时通讯模块 2](#_Toc44489859)

[3.1.1 实现步骤 2](#_Toc44489860)

[3.1.2 关键技术点说明 2](#_Toc44489861)

[3.1.3 运行效果图 3](#_Toc44489862)

[3.2 采用restful接口的前后端交互模块 3](#_Toc44489863)

[3.2.1 实现步骤 3](#_Toc44489864)

[3.2.2 关键技术点说明 3](#_Toc44489865)

[3.2.3 运行效果图 4](#_Toc44489866)

# 引言

## 设计目的

Android可以通过调用Springboot的restful接口来获取服务端提供的数据显示到手机页面上，故以此来制作一个可以即时通讯的程序

## 设计内容

基于websocket的即时通讯模块以及采用restful接口的前后端交互模块

## 开发环境

Android Studio 、Idea

# 系统分析与设计

## 需求分析

（1）需要用idea搭建一个基于Springboot的服务端程序，给客服端提供可调用数据的接口以及websocket服务

（2）需要用Android Studio来做客户端界面以及调用服务端的接口及websocket服务

## 软件架构设计



# 各功能模块的详细设计

## 基于websocket的即时通讯模块

### 实现步骤

服务端采用Springboot搭建，整合了websocket

### 关键技术点说明

Springboot整合webocket后可以通过ws://localhost:8080 /ws/{id}给每个用户建立连接，id=1的用户发送携带id=的2请求体数据的post请求时，id=2的用户会立即受到该信息同时做出反馈。这就是websocket的功能，我就基于此功能实现了聊天室。

### 运行效果图



图3.1.3 利用websocket在线测试工具测试的结果

## 采用restful接口的前后端交互模块

### 实现步骤

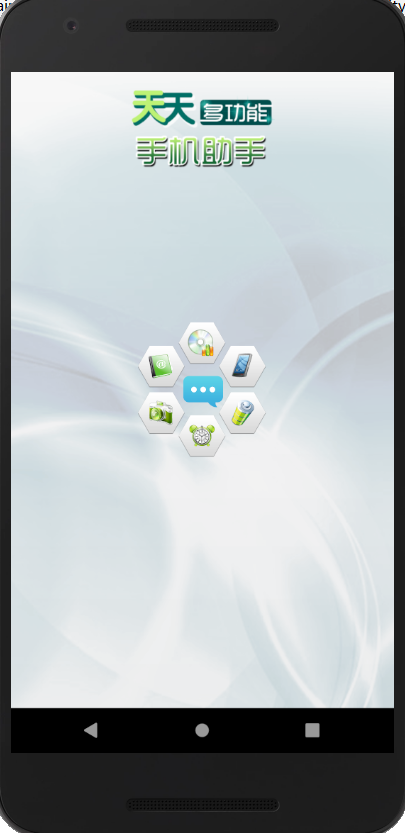
服务端采用的是rest风格的url，安卓前端就可以通过这些url来获取想要的数据，但安卓自带的类库不能实现，需要通过OkHttp这个请求库来完成，所以需要先引入这个包，再将里面的类封装成工具类方便调用；安卓访问服务端还有个问题就是数据的格式，restful一般携带的是json数据，所以还需要引入Gson这个包来统一数据的格式。

### 关键技术点说明

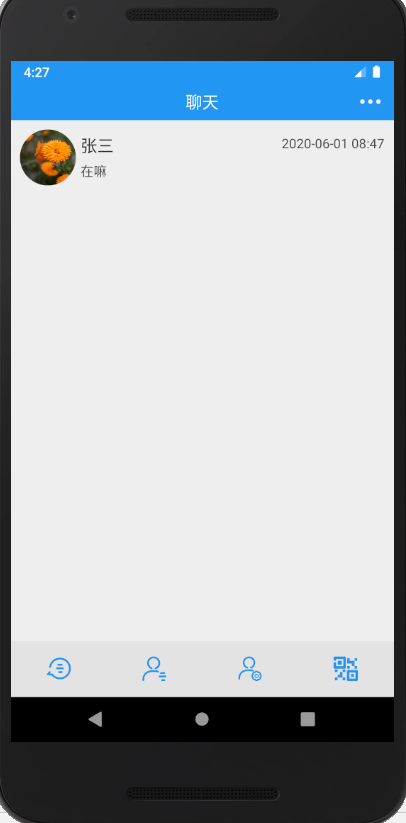
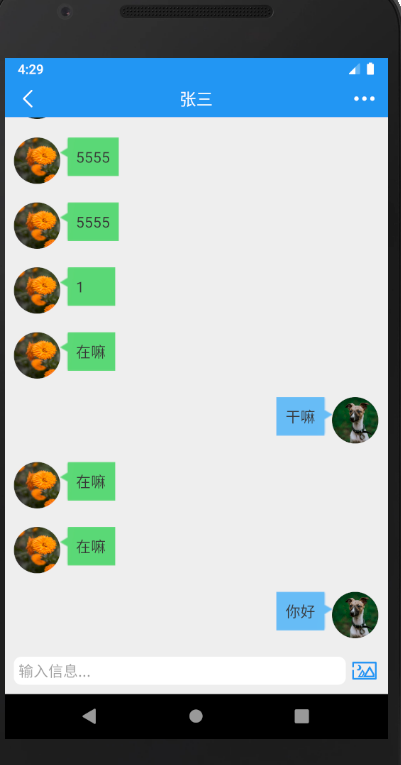
OkHttp来实现前端向后端请求数据，Gson来解决数据的格式问题

### 运行效果图

1．点击天天助手中间的按钮会跳到聊天室的登录页面，输入账号密码就会向服务端发送请求传递数据，服务端再从数据库中匹配数据

2．进入聊天页面后，程序会先从数据库中获取已发送的消息，传递给安卓客户端；用户发送消息时，会将消息通过OKHttp通过json形式传递给后端，后端再将数据存入数据库中。

3．另一用户登录后会看到未读消息，实际因为消息在数据库中状态字段为0，点击进入聊天页面，会将未读消息修改为已读，再从数据库中查询历史消息。

