



惠州学院

HUIZHOU
UNIVERSITY

毕业论文(设计)

中文题目: 基于微信小程序的同城拼车系统

英文题目: Local Carpooling System
Based on WeChat Applet

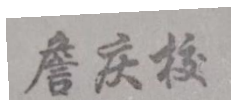
姓 名	<u>詹庆校</u>
学 号	<u>1714080903118</u>
专业班级	<u>17 网络工程 1 班</u>
指导老师	<u>陈绪行 老师</u>
提交日期	<u>2021 年 2 月 27 日</u>

教务处制

学术诚信声明

本人所呈交的毕业论文，是在指导教师的指导下独立完成。研究工作所取得的成果、数据、图片资料均真实可靠。除文中已注明引用的内容外，不包含任何其他人或集体已经发表或撰写过的作品或成果。对本论文的研究做出重要贡献的个人和集体，均已在文中以明确的方式标明。本人完全意识到本声明的法律结果由本人承担。

本人签名：

A rectangular stamp containing the handwritten signature '詹庆松' in black ink.

日期：2021.2.27

基于微信小程序的同城拼车系统

作者姓名：詹庆校 专业班级：17 网络工程 1 班 指导老师：陈绪行老师

摘要

基于微信小程序的同城拼车系统是一个后端采用 SpringBoot+Redis+MySQL 技术的系统，包括了微信授权、行程发布、聊天室交流等功能。该系统除了保留传统意义上的拼车模式，省去平台的复杂手续和费用，实现了用户之间的透明交易，还借助了小程序的平台解决了传统 APP 存在的弊端，轻松使用，简化了发布行程需求的步骤，具有轻量使用的优秀特性。

本文主要的研究方向为基于微信小程序的同城拼车系统，本人将对系统进行需求分析和可行性研究，确定需求并进行相关模块设计，代码构建以及软件测试。

关键词： 微信小程序 拼车系统 SpringBoot

Local Carpooling SystemBased on WeChat Applet

Author: Zhan QingXiao Major: network engineering Tutor: Chen XuHang

Abstract

The city Carpooling System based on WeChat Applet is a back-end system using SpringBoot + Redis + MySQL technology, including wechat authorization, itinerary release, chat room communication and other functions. The system not only retains the traditional carpooling mode, saves the complex procedures and costs of the platform, realizes the transparent transaction between users, but also solves the disadvantages of traditional app with the help of small program platform, which is easy to use, simplifies the steps of publishing travel demand, and has the excellent characteristics of light use.

The main research direction of this paper is the city carpooling system based on wechat applet. I will analyze the requirements and feasibility of the system, determine the requirements and carry out relevant module design, code construction and software testing.

Key words: WeChat Applet Carpooling System SpringBoot

目 录

1 绪论.....	1
1.1 研究背景和意义.....	1
1.2 研究现状.....	1
1.3 研究的主要内容以及论文的组织结构.....	3
1.3.1 研究的主要内容	3
1.3.2 论文的组织结构.....	4
2 相关技术分析.....	5
3 系统分析与设计.....	9
3.1 需求分析.....	9
3.1.1 系统需求.....	9
3.1.2 功能需求.....	9
3.1.3 性能需求.....	10
3.1.4 安全需求.....	10
3.1.5 开发环境需求.....	10
3.2 系统总体结构设计.....	11
3.3 数据库设计.....	12
4 系统的详细设计与实现.....	17
4.1 用户发布行程接口的实现.....	17
4.2 聊天室进入与退出接口的实现.....	18
4.3 微信模块.....	18
4.3.1 微信信息绑定接口的实现.....	18
4.3.2 微信消息接受接口的实现.....	19
4.3.3 微信消息发送接口的实现.....	20
5 系统测试.....	22
6 总结与展望.....	29
6.1 总结.....	29
6.2 展望.....	29
参考文献.....	31
致 谢.....	33
附 录.....	34

1 绪论

1.1 研究背景和意义

自二十一世纪起，我国的生产力和信息化的高速发展，社会经济往信息化和科技化的趋势发展，在提升了生活方便性的同时，也对环境造成不小的压力，其中，高校学生网约车出行也越来越频繁。

有时候，车主出行时车上还有好几个车位，而乘客有时候出行的时候只有一两个人，如果乘客和车主拥有相同或者目的地相近的行程，本来可以一辆车就完成，而由于信息不对等导致变成两辆车出行，增加了汽车尾气的排放，同时还易对交通造成拥挤。这一现象在大学里面尤其常见的是学校放假的时候，大家的目的地都是车站，多人拼一辆车明显会比一个人打一辆车便宜。

目前大部分高校学生出行方式大多采用滴滴快车的形式，即使出行的人数只有一两个，但是还是叫了一辆车，花费的费用也经过系统的提成加收容易导致花费偏高。单人或双人只叫一辆车的话会造成多辆汽车出行，高校学生是一个庞大的群体，如果每个人都这样的话，积少成多，聚沙成塔，交通拥挤，环境影响，资源浪费的问题就会接踵而来，而从小事做起，实行共享经济，拼车出行，不仅解决上述问题，还能在日常使用中省下一笔笔的费用，无疑是个好办法。尤其是17年的共享经济的大爆发，加速了我们的平时生活往绿色环保的方面发展的脚步。

因此搜集和整合需求信息，让资源得到高效和最大化的利用，减小多辆汽车出行所排放尾气的对环境造成的影响，同时做到对交通压力的缓解，就是这个拼车系统的意义所在。

1.2 研究现状

在国外,最早出现拼车行为的是在新加坡。从上世纪七十年代就开始使用通行证制度,规定如果私家车乘坐人数少于4人,就需要办理通行证才能在道路上行驶,否则就会被罚款。这实际上鼓励了民众拼车出行方式的兴起。1984年,为了缓解洛杉矶市因为奥运会的交通压力,洛杉矶政府出台了“拼车”方案。即向高速公路提供一条专用的车道,只要乘坐的人数达到要求就可以免费使用,如果人数不达到要求违规使用该车道就会被处以高额罚款,为此出现了拼车可以省钱的概念。^[8]

2015年,共享概念开始在国内兴起,资本和巨头开始对共享概念进行商业布局,催收了共享电宝,共享电动车的产生。2016年为共享概念发展的元年,2018年是走向成熟的一年。我国国民的环保、健康意识也在经历过“雾霾风波”之后也逐渐增强;而一、二线城市交通现状变得越来越差,且在限行的政策等限制下,拼车概念这类环保、灵活、高效、健康的出行方式便受到了关注,拼车出行成为受到人们普遍追捧的一种绿色出行方式。^[3]

显然,相较于传统的出行模式,一人一车或者两人一车的出行来说,共享出行的模式可以缓解交通压力,环保和健康的出行也能减小对环境的压力。假如消费者都可以对自己的资产实现共享,比如汽车、房屋等,那么社会对这些产品的额外需求就可能会下降。无形中就可以减小车贷房贷的压力。消费者有需求,市场就有应对,在公共交通上,顺路搭乘(即“顺风车”“评车”)的发展必是长久能存在的模式,并能实实在在地改变我们生活的模式。^[1]

国内拼车领域有名的app是滴滴,滴滴的优势在于快车服务,随叫随到,并且只要拥有私家车,人人都是司机。但滴滴本身存在不少问题:在拼车服务方面,注册流程繁琐,作为车主需要上传除去上传驾驶证、身份证、车辆信息之外,还需要几天的人工时间审核,手续繁杂,个人信息全被平台收集,且提供拼车服务的多是那些专业跑滴滴的,从而将拼车演变成之前的网约车模式,共享模式演变成商业模式。

小程序在娱乐,实用性等方面已成为日常使用方式,作为移动便携的互联网入口,被广泛地运用在众多行业领域,小程序是一种不需要下载安装即可使用的应用,“触手可及”“用完就走”是它的特点,随时可用但又无需安装卸载^[1]。小程序的改变传统app的使用方式,减少了手机app的数量,释放了手机的内

存使用空间，且无需担心系统兼容等问题。显然，用小程序的形式，简化发布行程的流程，回归原始的拼车模式，无疑是解决上述问题的好方法。而费用结算方面，网约车还未出现之前，出租车的形式就是打表，到站后结算，所以没必要走平台，完全可以线下结算，不仅免去繁杂的注册和使用等流程，同时省去平台的手续费。

1.3 研究的主要内容以及论文的组织结构

1.3.1 研究的主要内容

本项目研究目标为基于微信小程序的同城拼车系统。乘客和车主可以通过微信小程序，在上面发布自己的出行需求，通过获取微信地图的数据选择好出发和目的地，并填好联系方式，出行时间，出行人数或者车上剩下空位，还可以备注出行要求，诸如车主要求乘客没有吸烟习惯，乘客备注是否携带行李等等。

本小程序主要分为三大模块：

1) 行程信息发布功能：乘客与车主可以利用地图发布自己的行程信息，在对出行的信息进行填写后，选择是乘客寻找车辆，还是车辆寻找乘客，便可发布出行的需求信息；

2) 交流功能，乘客与车主可以对自己感兴趣的行程进行交流

3) 个人中心，乘客与车主可以在个人中心对自己的行程发布记录和自己感兴趣的相同活相近的行程的收藏内容进行查看比对。

依据软件工程的流程规范对小程序系统进行开发，流程如下：

1) 系统需求分析，确定基于微信小程序的拼车系统后台以及客户端需要具备的功能（2020.12.17-2021.01.01）

2) 基于微信小程序的拼车系统后台以及客户端的概要设计，并确定初步方案（2021.01.02-2021.01.16）

3) 基于微信小程序的拼车系统后台以及客户端的总体设计和详细设计（2021.01.12-2021.02.12）

4) 基于微信小程序的拼车系统后台以及客户端的数据库设计

(2021.02.13-2021.02.23)

5) 基于微信小程序的拼车系统后台以及客户端进行系统测试

(2021.02.24-2021.03.1)

开发时间计划如图 1-1 所示:

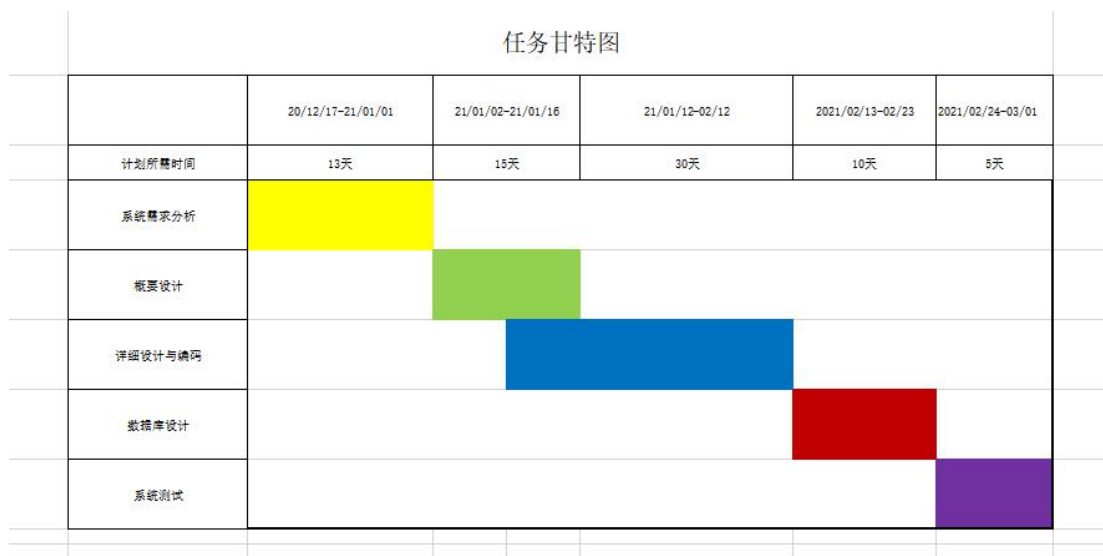


图 1-1 任务甘特图

1.3.2 论文的组织结构

1) 绪论。介绍基于小程序的拼车系统的设计开发背景,结合目前的拼车系统技术的相关技术实现的系统,确定本论文的研究内容和总体组织结构

2) 相关技术分析。比较目前的主流和常用的技术框架,选择适宜的技术

3) 需求分析。针对目标用户,具体分析系统的功能需求

4) 系统总体设计。根据系统的功能需求,首先进行系统调查和分析,然后将系统的总体需求进行划分并确定系统模块

5) 系统详细设计。针对上一流程划分好的模块,逐一对每个模块进行详细分析和流程图设计,进一步对系统进行逻辑设计(如果可以的话写出数据字典后还能画出数据流图)、物理设计;

6) 系统测试。按照设定的系统需求,测试软件的功能稳定性

7) 总结与展望。对基于小程序的同城拼车系统后台以及客户端进行项目总结,针对可能存在的缺陷或者不足提出相关解决思路或者参考方法。

2 相关技术分析

2.1 服务器技术分析

微信小程序是一种基于移动端微信平台开发的程序,最为突出的特点就是数据的传输过程安全且高效,它采用的是与 C/S 架构相同的交互方式,加上小程序所采用的 MINA 框架具有实时响应的特点,所以在完成数据的绑定和同步响应时速度特别快,使得基于 Apache HTTP Server 服务器中间件的小程序具有速度和安全的保障,有利于进行系统聊天室的开发。^[12]

系统架构

基于微信小程序的拼车系统采用的是小程序模式,通过服务器与微信服务器进行数据交互,因此用户只需要登录微信通过小程序就可以访问服务器。

软件架构如图所示:

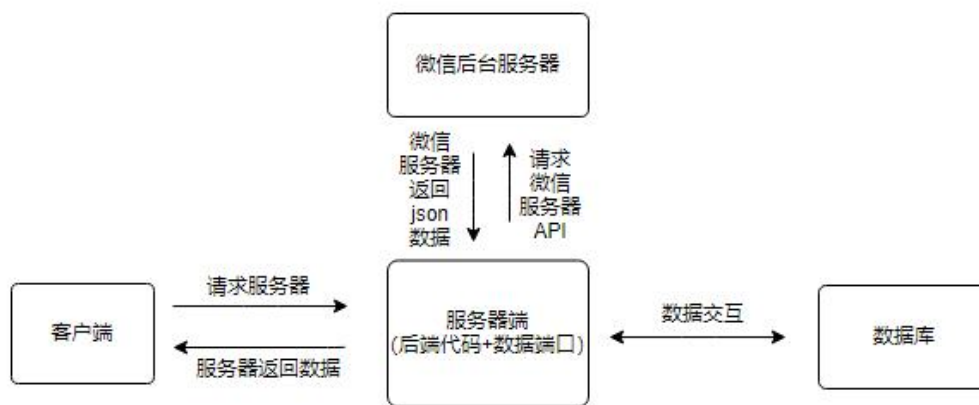


图 2-1 软件架构图

2.2 后端框架分析

由于稳定的性能、良好的开放性及严格的安全性，还有其优秀的跨平台性，J2EE 受到广泛地使用，J2EE 目前成熟且经常使用的有 SSH 和 SSM。

SSH 通常是指 Struts、Spring、Hibernate，它们是 J2EE 目前较为成熟的 Web 开发框架组合。它们分工分别是：Struts 是作为系统的控制器，Spring 负责管理各层组件，核心是控制反转(ioc)和面向切面(AOP)的容器框架，Hibernate 负责持久化层的数据交互，以面向对象思想封装 JDBC 来进行数据库的访问。

SSM 与 SSH 不同的是，它是由 Spring、SpringMVC、Mybatis 这三个开源的框架整合的，SpringMVC 代替了 Struts 作为系统的控制器，Mybatis 代替 Hibernate 来负责系统的 DAO 层，相较于 SSH，它是采用了 MVC (Model、View、Controller) 思想的一种技术，比基于对象的 SSH 来说，SSM 的使用更多。

然而两个框架由于其开发笨重、配置繁多、部署流程复杂、第三方技术集成难度大，这些框架在开发过程中需要配置各种配置文件，大大影响开发的效率。在 SpringBoot+Mybatis+Redis 中，SpringBoot 是用来简化 Spring 应用开发的轻量级开源框架，从本质上看，SpringBoot 是一些代码库的集合，通过简化配置来简化 spring 在整个的应用的搭建和开发过程，方便了开发者的开发步骤。SpringBoot 不但继承 Spring 框架原有的优秀特性，还通过集成大量框架的方式很好的解决依赖包之间版本冲突，不稳定引用性等问题。^[4]

综上所述，基于小程序的拼车系统后台将采用 SpringBoot，即采用 (SpringMVC+MyBatis) 技术进行快速开发。

2.3 数据库分析

关系型数据库，指采用关系模型组织数据的数据库。通俗化语言就是关系数据库是由若干具有表示实体之间联系关系功能的二维表组成。^[20] 数据存储是由行和列组成的表中，每次按行查询检索信息；SQL Server、Oracle 和 MySQL 是目前主流的关系数据库。在使用上区别不大，根据系统要求，采用体积小，反应快的 MySQL。

MySQL 是提供支持多种语言连接的接口的开源数据库，在多数中小型项目开发上备受欢迎。它的核心程序采用完全的多线程编程，拥有轻量级进程的特性，

可以灵活使用，效率高。内存分配系统是基于线程的运行，快速而且稳定的，持续性使用且不必担心其稳定性。

2.3.1 Redis 技术

因为该系统在调用地图 api 的时候，微信服务器响应需要大概两秒钟，并且在实现聊天室功能需要大量 sql 语句去与数据库交互，当两个用户在聊天的时候，有可能因为数据库的反应太慢导致读到“脏”数据，导致聊天界面信息错乱，故需要采用一种缓存技术来提高微信在调用 api 时候的速度和数据库信息交互时候的一致性和完整性。

Redis 是基于 key-value 的存储系统，通过多种键值数据类型来适应不同场景下的存储需求，Redis 主要具有以下优点：

1) 响应快速

Redis 响应非常快，写入执行速度可以达到大约 110 000 /s，读取速度 81 000 /s，其速度远超数据库。特别是当存入常用数据时，系统性能会成倍提高。

2) 支持的数据类型丰富，字符，队列，哈希等二进制的数据类型操作 Redis 都支持；

3) redis 支持事务，对数据的更改操作都是原子性的，要么全部执行，要么全部不执行；

因为 Redis 具有上述这些优点，可以利用缓存大大减轻数据库的压力，且由于其持久化的特性可以降低数据丢失的概率，保证了聊天内容数据在系统中的一致性和完整性。

2.5 前端框架

前端采用微信小程序云开发，微信小程序的目录结构主要可以分为四种文件类型，分别是：.wxml 文件、.wxss 文件、.js 文件和.json 文件。前两类文件分别是类似于 html 的页面结构文件、CSS 的样式文件；.json 文件是关于配置的

文件，而 .js 就是传统的 JavaScript 文件，只是微信小程序相较于传统 js 文件不同的是小程序的 js 文件会将一个 Page 对象实例化并作为函数的入口。^[16]

本系统所调用的地图 api 就存放在 js 文件中。

微信小程序前端采用了 jar 包的形式来封装通用函数，对外提供模块化接口设计的工具函数。根据 js 的闭包原理，我们只需要通过调用一个函数就可以返回所需要的值。微信前端小程序还有个优秀的特性就是前后端完全分离，数据接口由后端提供，前端通过调用端口实现系统功能，完成数据交互。

由于微信小部件本身提供了许多已经打包好的功能，利用微信小程序提供的 API，可以调用地图实现系统获取起点和终点的功能。微信自带的地图 api 本质主要由两个函数实现：wx.openlocation() 可以定位当前位置对应的经纬度和地址消息，而 wx.getlocation() 可以获取地理位置接口，而在选择终点时的移动地图选点是由 wx.chooseLocation() 来实现，原理是通过移动选点，获取该点位置的经纬度和地址消息，然后拼接成 url 并通过调用库函数 wx.navigateTo() 去寻找 url 所对应的图片消息，实例化成 Page 对象返回形成前端地图界面。

综上所述，微信小程序前端开发具有以下优点：

1) 快捷方便，即用即走。

不需要再下载什么 APP，既费流量，又占空间内存。

2) 速度快、不占内存

小程序前端代码都是存在云端的微信服务器上的，得益于微信，小程序可以直接打开，速度也比较快。并且还不占用手机内存。

3) 功能丰富，场景丰富

可以调用比 H5 更多的手机系统功能来开发，例如 GPS 定位、拍视频、重力感应等。

3 系统分析与设计

本章对基于小程序的同城拼车系统后台及客户端进行系统总体设计，首先对系统进行完整需求分析，再根据需求分析进行详细设计。

3.1 需求分析

3.1.1 系统需求

模拟学校学生外出的情况，设计相关的需求。在基于小程序的拼车系统中，主要角色为车主和乘客，为了简化拼车的流程，将车主和乘客集成一个角色，作为程序的使用者，可以以一个身份发布起点到终点的需求。

1) 在后台服务器中，使用者通过微信授权登录，可以调用微信地图接口选择起点和终点发布行程信息

2) 使用者可以通过聊天室，与行程需求发布者对拼车的需求进行交流。

3) 当订单完成时，使用者可以对订单信息和订单对应的聊天室进行关闭

4) 数据库需要存储记录行程信息，包括行程位置信息，时间，个人信息等

基于小程序的拼车系统后台及其客户端为有拼车和出行提供一个便捷服务的平台，省去下载和注册 app 的繁琐程序，即用即走。旨在提高有出行需求的使用便捷性以及整合资源信息，减少出行车辆，保护环境。

3.1.2 功能需求

根据上述的基于小程序的拼车系统及其客户端所要实现的功能需求，可以将后台分为行程发布模块和信息交流模块以及微信模块等。

后台功能主要由微信客户端的小程序界面与用户进行交汇,当前台接受到请求,后台服务器 就负责将数据信息处理后存储于数据库中,同时提供微信客户端,前台界面,websocket 和地图接口。每个接口对应不同的功能,根据不同请求提供相应的业务流程,整合处理数据存放于后台数据库,再返回 json, xml 等格式的数据,并提供相对应的 api 服务。

综上所述,得益于微信小程序提供很多封装好的函数的特点,使用简单,调用函数容易,故采用前端小程序。

3.1.3 性能需求

基于微信小程序的拼车系统要实现的是由 web 后端和微信小程序相辅相成的平台,除了满足系统要求,整个软件在性能方面需要满足一下要求:

- 1) 小程序前端界面友好简洁,操作简单
- 2) 可维护性: 根据需求选用适合的开发框架,在软件的开发和维护方面变得不难
- 3) 稳定性: 系统在使用过程中运行稳定,不易出 bug

3.1.4 安全需求

任何软件都需要对安全性有一些要求,保护使用者的信息以及保障小程序运行环境的稳定性,在安全方面,整个软件需要满足以下要求:

- 1) 机密性: 系统在未经过授权登录的情况下禁止访问数据库
- 2) 完整性: 任何登录用户对用户表只有读权限
- 3) 持久性: 即使收到恶意攻击,小程序依然能运作
- 4) 认证: 确保登录的用户是可信的(可从微信服务器获取真实用户信息)

3.1.5 开发环境需求

基于微信小程序的拼车系统后台采用 Springboot+Redis+MySQL 的框架和技术。

后台系统的开发环境需求如下:

- 1) 操作系统: Windows10
- 2) 开发工具: IDEA
- 3) 服务器: 本地运行

3.2 系统总体结构设计

本系统共有三模块，对应的功能是行程模块、通讯模块和微信模块。

行程模块：使用者可以对行程信息进行管理修改进行数据操作后进行行程的发布。

通讯模块：使用者调用微信登录接口获取凭证（code），将获取到的凭证通过后台服务器换取 session，建立通讯。

微信模块：后台服务器通过微信开发接口，调用地图 api 和获取登陆者的信息，为使用者提供小程序形式的系统功能。

如图 3-1 所示，系统模块划分图如下：

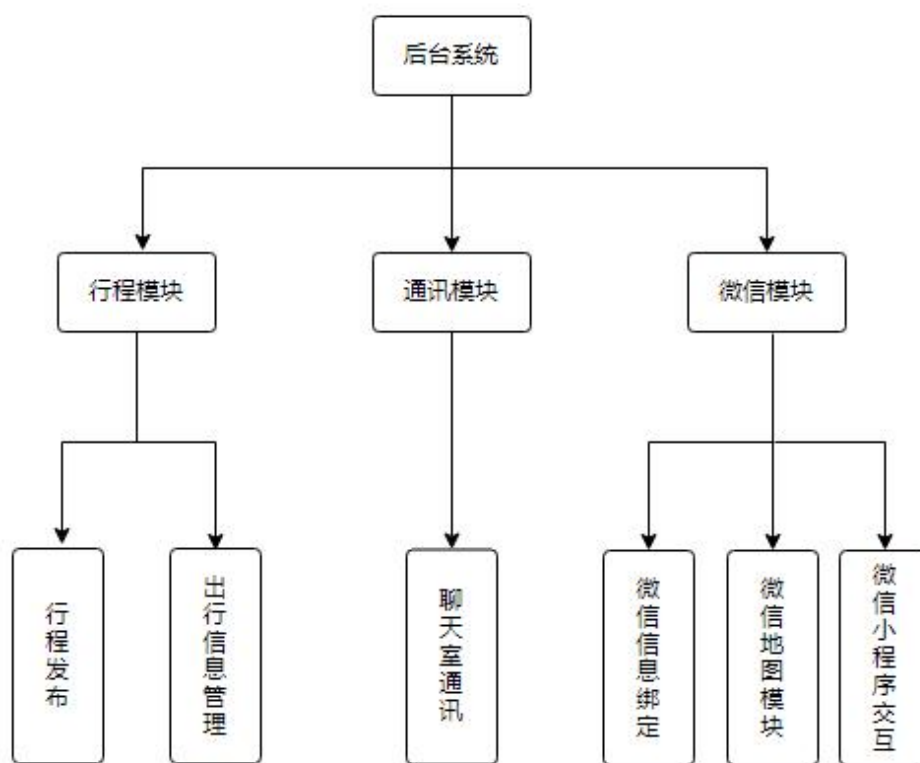


图 3-1 系统模块划分图

后端管理设计包括数据库设计以及为实现数据交互后端所提供给前端客户端的 api 接口。

小程序采用的解析技术类似 React Native 和 Weex (React Native 和 Weex 是两种开发高性能原生应用的框架) 相比于公众号的形式, 小程序具有更好的原生体验, 微信前端的界面主要有三部分, 主要处理控制微信业务逻辑部分的模块, 数据绑定、页面渲染和组件构建的部分, 还有主要响应用户和交互的监听线程函数。为搭建一个稳定安全高效的小程序, 整个小程序的框架主要分为 View (视图层) 和 APP Service (逻辑层), 在这之间有数据传输和事件系统。^[12]

如图 3-2 微信小程序框架:

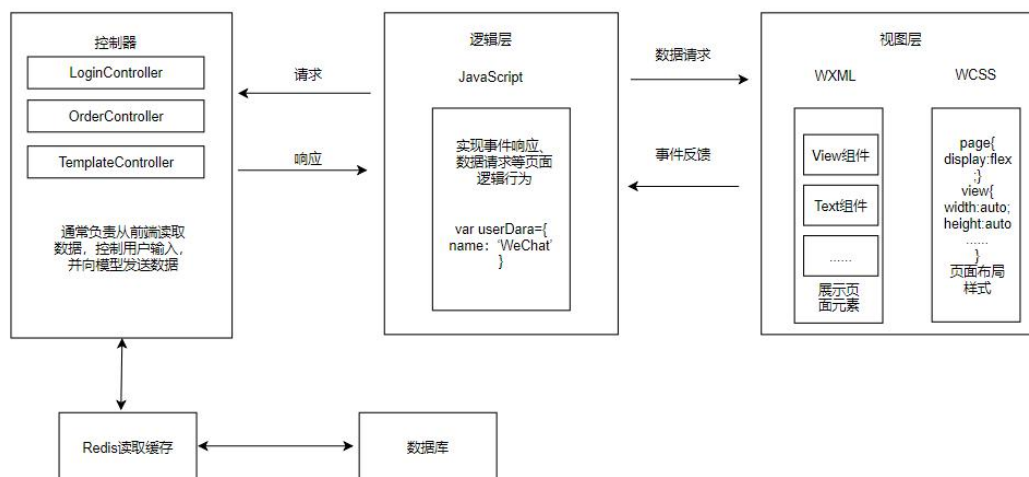


图 3-2 微信小程序框架

3.3 数据库设计

根据系统需求确定系统所需的数据实体和流程图对数据库进行设计和构建。可以主要分成用户、聊天室和订单三个实体之间的联系, 故可以围绕它们设计表格为用户表存放用户信息, 聊天室表存放聊天信息, 地址表存放地址信息 (起点和终点), 拼车信息表存放订单信息, 设计一个用户映射表是方便找到聊天室的

创建人，设计的逻辑是谁创建的订单就是谁创建了聊天室，当有多个聊天室被创建时，方便系统将订单和聊天室联系起来。

3.3.1 数据库 E-R 图

该系统涉及到多个实体，包含用户，聊天室，用数据库实体关系模型展示关联，如图 3-3 所示：

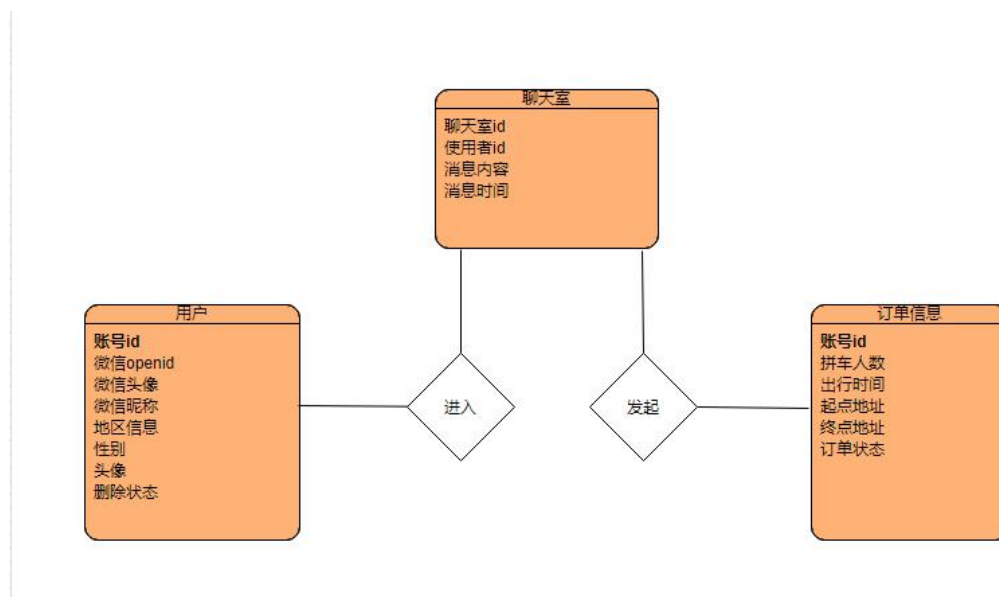


图 3-3 E-R 图

数据库逻辑结构设计：

用户表（user）

编号	是否主键	字段名称	数据类型	大小	允许为空	说明
1	是	id	int	128	否	Id
2	否	gmt_create	datetime	0	是	创建时间
3	否	gmt_modified	datetime	0	是	修改时间
4	否	last_login_time	datetime	0	是	最后一次登录时间
5	否	openid	varchar	64	否	微信 openid

6	否	nick_name	varchar	64	是	微信昵称
7	否	gender	tinyint	3	是	性别
8	否	avatar_url	varchar	256	是	用户头像
9	否	country	varchar	256	是	地区信息
10	否	province	varchar	32	是	
11	否	city	varchar	32	是	
12	否	language	varchar	8	是	区分中英版微信
13	否	school	varchar	32	是	学校
14	否	phone	int	16	是	电话
15	否	deleted	tinyint	2	否	删除状态（0，1）

地址表（address）（10,6 意思是长度为 10，小数点后 6 位）

编号	是否主键	字段名称	数据类型	大小	允许为空	说明
1	是	Id	varchar	128	否	Id
2	否	gmt_create	datetime	0	是	创建时间
3	否	gmt_modified	datetime	0	是	修改时间
4	否	name	varchar	32	是	搜索地点
5	否	address	varchar	64	是	地址
6	否	latitude	double	10,6	是	纬度
7	否	longitude	double	10,6	是	经度

8	否	type	tinyint	2	是	'0'起点，'1'终点
9	否	hot_point	int	128	是	使用次数

聊天室消息表（Roommessage）（聊天室 id 是当多个订单被创建时，可以区分消息内容属于哪个聊天室）

编号	是否主键	字段名称	数据类型	大小	允许为空	说明
1	否	user_id	int	128	否	使用者 id
2	否	time	date	0	否	消息时间
3	否	message	varchar	1024	否	消息内容
4	否	order_id	int	128	否	聊天室 id

拼车信息表（Pin_order）

编号	是否主键	字段名称	数据类型	大小	允许为空	说明
1	是	Id	int	128	否	Id
2	否	gmt_create	datetime	0	是	创建时间
3	否	gmt_modified	datetime	0	是	修改时间
4	否	nick_name	varchar	64	是	创建者微信昵称
5	否	Startaddress_id	int	128	是	起始地的 id （地址表的主键）
6	否	Endaddress_id	int	128	是	目的地的 id （地址表的主键）
7	否	target_time	datetime	0	是	出行时间

8	否	target_num	int	32	是	出行人数
9	否	current_num	int	32	否	当前聊天人数

用户与聊天室映射表 (User_order_re)

编号	是否主键	字段名称	数据类型	大小	允许为空	说明
1	是	Id	varchar	128	否	Id
2	否	gmt_create	datetime	0	是	创建时间
3	否	gmt_modified	datetime	0	是	修改时间
4	否	Order_id	int	128	是	聊天室 id
5	否	User_id	int	128	是	创建者 id
6	否	leader	tinyint	4	否	队长队员 (1, 0)

4 系统的详细设计与实现

4.1 用户发布行程接口的实现

获取微信客户端请求的数据，在数据库中进行查询，验证使用者身份后，将行程发布的相关地点信息存储在数据库中。如果保存成功，则返回发布成功，若失败（登录身过期或失效），则返回失败。

用户发布行程信息订单图如图 4-1 所示：

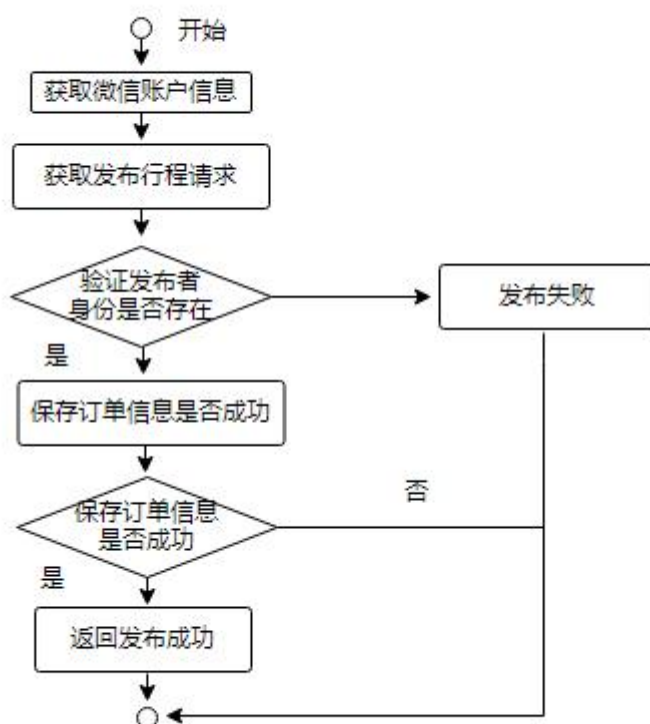


图 4-1 发布行程活动图

4.2 聊天室进入与退出接口的实现

获取微信客户端请求的信息，后台服务器根据 id，在数据库中进行查询用户信息表判断请求发起者的是否为聊天室的发起者，如果不是，就将当前正在聊天人数加一，如果消息发送成功，将使用者在聊天室所发送的聊天内容写入后台数据库存储起来，如果发送消息失败错误，则提示返回错误信息。设计表格存储聊天室消息是为了防止用户掉线或者网络不好导致聊天内容丢失，

聊天室交流图如图 4-2 所示：

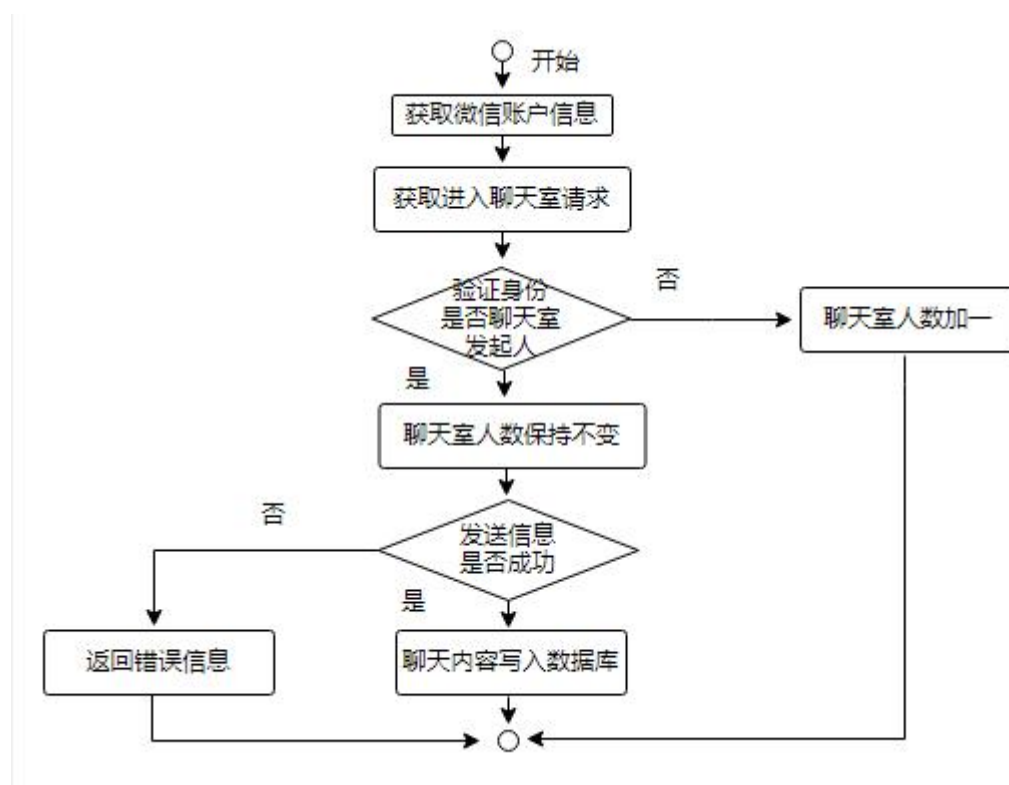


图 4-2 聊天室交流活动图

4.3 微信模块

4.3.1 微信信息绑定接口的实现

后台服务器在接收到读者发送的请求时，调用数据库的查询函数将请求中的账户信息与原有信息进行比对，如果数据库没有该身份信息，则向微信服务器发起获取使用者微信账户信息（微信 openid、微信昵称、地区和微信头像）的请求，将获得的数据保存到数据库。

微信信息绑定接口活动图如图 4-3 所示：

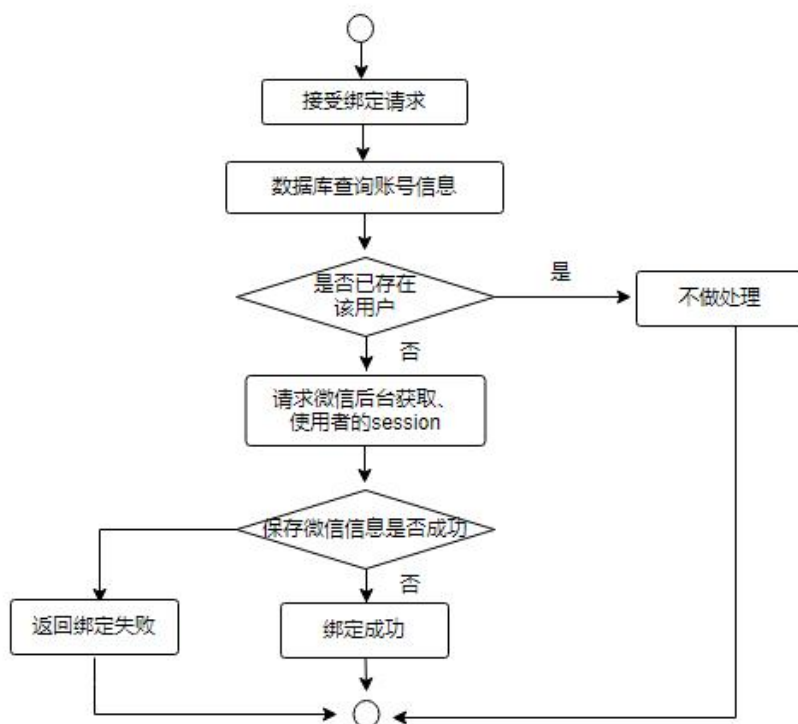


图 4-3 微信信息绑定接口活动图

4.3.2 微信消息接受接口的实现

微信服务器会根据对应接口接收到的消息转化为 json 格式的数据包，然后拼接到开发者填写的 URL 上发送给后台服务器。后台服务器可以判断传递的消息类型，选择相应的接口处理消息，并转换成 json 格式然后再返回给微信后台服务器。

微信消息接收活动图如图 4-4 所示：

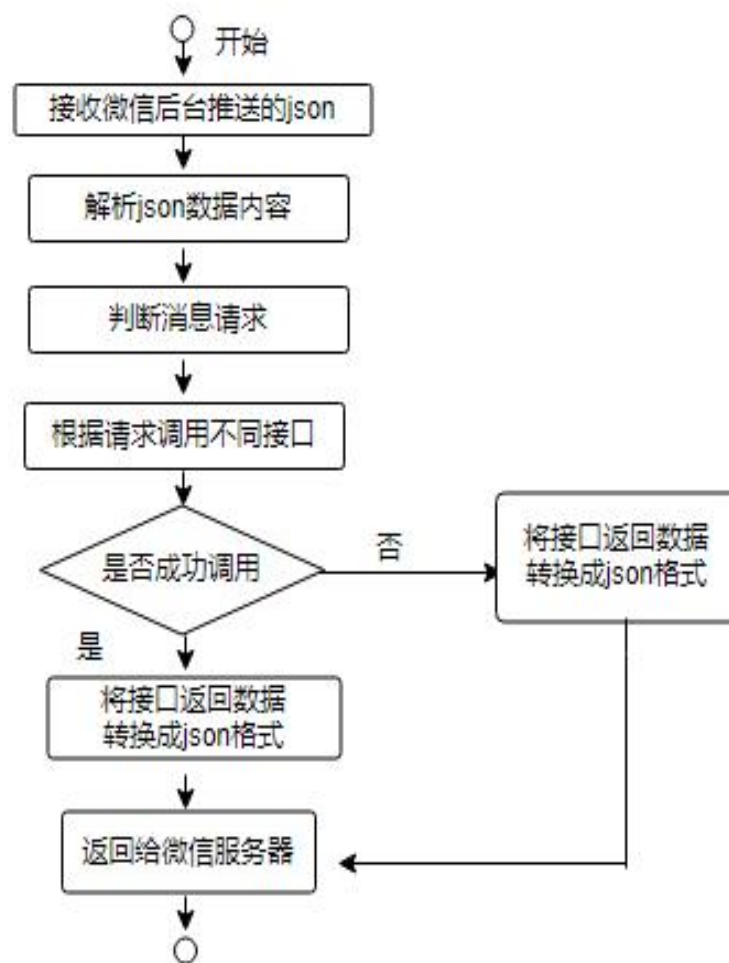


图 4-4 微信消息接受活动图

4.3.3 微信消息发送接口的实现

微信消息发送活动图如图 4-5 所示：

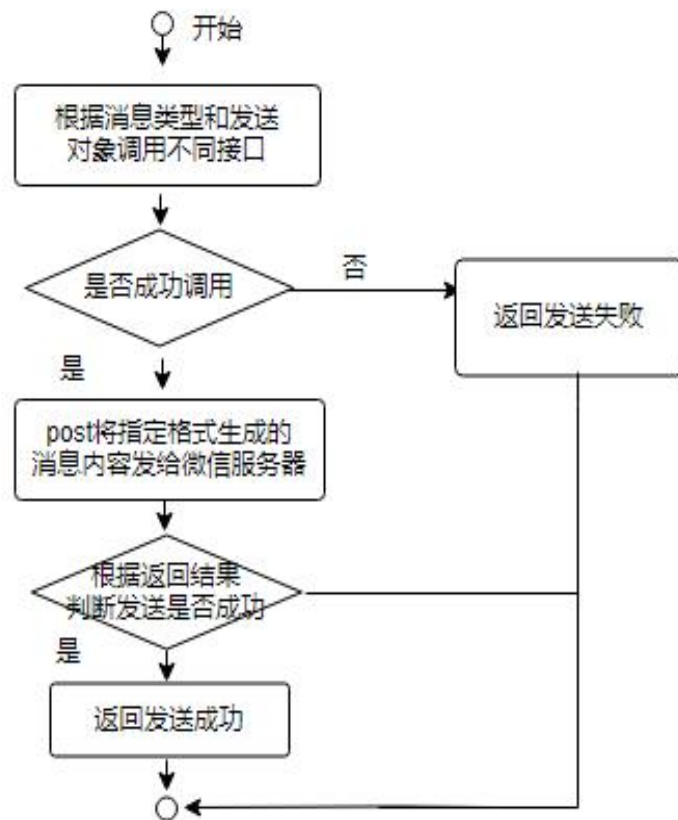


图 4-5 微信消息发送活动图

当使用者在微信小程序交互界面向聊天室发送消息时，微信会引用自身的模板处理，形成 json 格式处理返回给后端 springboot，后端查询数据库利用 mybatis 的 xml 配置查询，返回的是数组 json 格式，然后响应给前端，前端处理后形成聊天室界面。

5 系统测试

5.1 功能测试

5.1.1 用户登录测试

使用小程序需要先进行微信登录，用户登录设计逻辑如图所示，登录过程需要调用官方提供的 api 接口 `wx.login(object)`，然后获取登录凭证（code），将凭证发送给后台服务器换取微信的用户唯一标识（openid）与登录的密钥（session_key），再生成自己固定的 session，不会因为网络传输导致数据丢失，如图 5-1 和 5-2 所示，微信授权登录便可获取到信息。^[6]



图 5-1 微信授权登录效果图

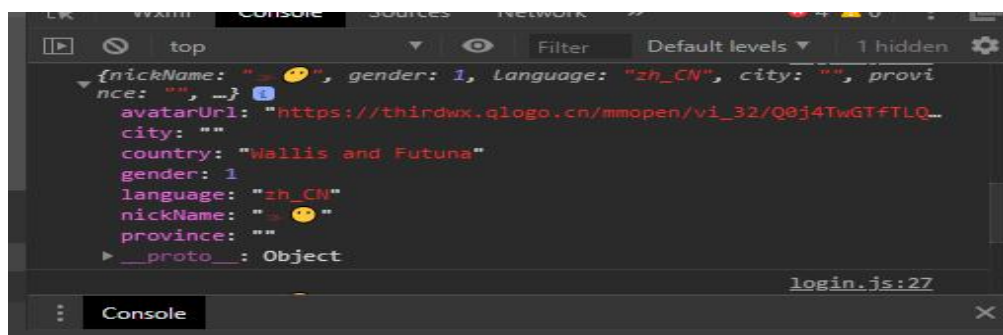


图 5-2 微信授权登录获取到的信息

5.1.2 出行功能测试:

选择出发地点, 如图 5-3 所示:



图 5-3 选择出发地点功能测试

选择出行终点，如图 5-4 所示：



图 5-4 选择终点功能测试

选择出行时间和人数，如图 5-5 所示：



图 5-5 选择出行时间和人数

5.1.3 聊天室功能测试:

当有新的人加入时，聊天室显示当前人数。

如图 5-6 所示:



图 5-6 显示聊天室当前活动人数

发布行程成功后可以选择该行程的聊天室进行交流，如图 5-7 所示:

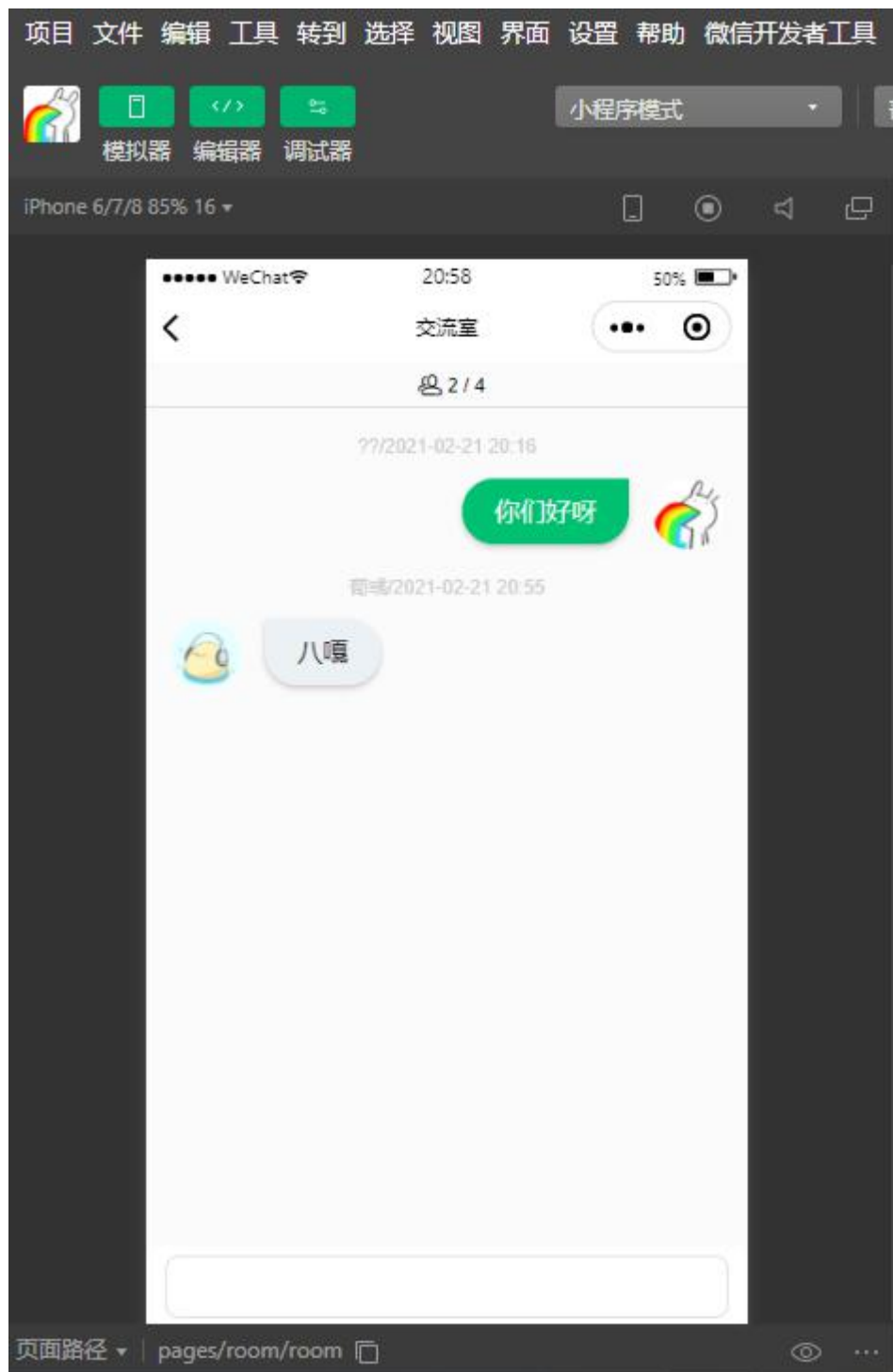


图 5-7 聊天室交流功能测试

6 总结与展望

6.1 总结

基于微信小程序的拼车系统主要包含两个子功能，分别是行程发布和聊天交流两个实用的功能组建成了这一系统。

微信小程序的成熟和普及避免了用户需再次安装其他应用来满足共享出行的需求，借助微信平台避免了移动端对不同手机的适配和兼容问题。微信小程序的便携性和组件的简洁性，给用户更好的使用体验。通过 springboot 设计实现的后台管理系统方便了系统的运行维护，微信开发者工具编写的前端，层次分明，易于代码检查和调试。

本项目的后台服务器将一些主要操作封装成接口，与后台系统、数据库还有微信服务器进行交互，使用 springboot 框架内置的 springmvc 框架，将项目整个分成 Model 层、Action 层、Service 层和 Dao 层，使得程序耦合性降低，整体功能性进行分层。

本文详细介绍了拼车系统的后台及其客户端的相关技术和需求分析，系统的功能模块设计和实现。虽然之前没有使用框架和微信小程序的开发经验，但在经过一些相关书籍和视频的学习了解，并在实习期间有去学习一些前端的基础知识，所以逐渐习惯了微信开发的设计思想和微信 api 接口的开发文档。

6.2 展望

虽然整个系统经过了详细的理论分析，但是因为本身对于知识的掌握还有些缺乏和不充分，前期的需求分析做的不够完善，导致前面的需求中有很多是不必要的需求，系统在实际运行中，仍存在一些遗漏的系统 bug，或者因为编码的格式导致的系统错误，加之时间有些赶，功能的完善性并不是很够。因此希望在未来能对系统的以下各个方面进行改进：

- (1) 对系统各个功能模块进行优化，使系统运行更加稳定
- (2) 设计一套推荐算法，比如基于地点或距离推荐合适度高的订单，

(3) 对数据库和小程序界面再进行优化，提高查询效率和页面简洁性

(4) 添加几个有趣的游戏或者交友小功能，让平时不使用出行的人也可以无聊时解解闷

参考文献

- [1] 赵光辉. 基于互联网的共享交通发展态势研究[J]. 综合运输, 2016, 38(06):21-27+31.
- [2] 张小龙. 关于小程序[J]. 商业观察, 2018(02):73-75.
- [3] 李琨浩. 基于共享经济视角下城市共享单车发展对策研究[J]. 城市, 2017(03):66-69.
- [4] 邓笑. 基于 Spring Boot 的校园轻博客系统的设计与实现[D]. 湖北:华中科技大学, 2018.
- [5] 陈伟. 微信小程序开发实战[M]. 人民邮电出版社:, 201709.
- [6] 倪旺. 基于微信小程序的车位预约设计[D]. 河北大学, 2020.
- [7] 萨师煊, 王珊等. 数据库系统概论(第三版)[M]. 北京:高等教育出版社, 2005. 45-187
- [8] 王茂福. 拼车的发展及其效应[J]. 中国软科学, 2010(11):54-61.
- [9] SUSAN WIEDENBECK, VENNILA RAMALINGAM. Novice comprehension of small programs written in the procedural and object-oriented styles[J]. International Journal of Human - Computer Studies, 1999, 51(1).
- [10] 沈炜, 徐慧, 汤倩. Mysql 数据库编程技术与实例[M]. 北京:人民邮电出版社, 2005. 114-226
- [11] 雷磊. 微信小程序开发入门与实践[M]. 北京:清华大学出版社, 2017
- [12] 李哲, 周灵. 微信小程序的架构与开发浅析[J]. 福建电脑, 2019, 35(12):66-69.
- [13] 徐星明. 微信小程序第三方平台的设计与实现[D]. 山东大学, 2020.
- [14] Anonymous. Frost & Sullivan: Car Pooling and Car Sharing: Simple Solution to Solve Complex Issues[J]. M2 Presswire, 2008.
- [15] 陆禹文. 基于微信小程序的移动学习平台的设计、开发及应用[D]. 兰州大学, 2020.
- [16] 柏超宇, 顾怡, 杨丽雯, 张建波. 智慧校园微信小程序云服务开发与构建[J]. 电子技术与软件工程, 2018(19):72-73.
- [17] 王婷婷, 谢晓茹, 于佳. 微信小程序对移动互联网生态圈的影响[J]. 电脑知识与技术, 2018, 14(32):294-296.
- [18] 储广昕. 分析软件开发中数据库设计理论的实践[J]. 信息通信, 2015, 149(5):127-128.

- [19] 潘哲, 邓芙蓉, 韩明正, 刘彩营. 基于 WebSocket 协议的特征匹配聊天系统[J]. 科技经济导刊, 2020, 28(15):9-10.

致 谢

在论文的结尾，我想对我大学遇到的所有老师和同学们表达由衷的感谢！是你们陪伴了我的大学时光，有鼓励，有陪伴，让我成长了许多，也拥有了如此多彩的大学，如此美丽的青春。

2020 突如其来的疫情影响了 20 和 21 两届学生，大三上半学期正从学校开开心心回家过寒假的时候，就被按了暂停键，对于我们来说，在惠州学院的记忆只拥有两年半，就直接变成准大四生了，然后就要面对实习和毕业论文两大难题，虽然疫情期间出来找实习特别难，伴随着焦虑心理，但有学院和老师们的鼓励，慢慢也就平稳渡过了。记得刚入大学时，特别腼腆，都不敢跟陌生人说话，那会加部门还是被师兄师姐“强拉去”的，很幸运加入了很有爱的部门，大家在一起说说笑笑，很快就把我的性格打开了，在这期间学习到很多与人相处的技巧，我不算那种读书特别厉害的人，所以我不能说我的大学过的多么充实，但是我能说我大学没有虚度，我一直认为大学不是来做书呆子的地方，大学是来发掘兴趣爱好，打磨性格的地方。

最后，我要特别感谢陈绪行老师，老师给我论文的指导态度一直是循循善诱，谆谆教导，不给我压力，鼓励我早睡早起，是一位好老师。感谢大学遇到的所有人，感谢学习上给过我照顾的老师 and 同学，感谢生活上给我帮助的朋友和家人，希望在未来的日子里，大家一切安好。

附 录