

**毕业设计（论文）**

**毕业设计（论文）题目：**

|  |
| --- |
| 自习帮-校园自习室预约系统设计与开发 |

|  |  |
| --- | --- |
| 学院/分校： | 理工学院 |
| 年级、专业： | 2019秋软件工程 |
| 教育层次： | 本科 |
| 学生姓名： | 韦治宇 |
| 学号： | 20192310830193 |
| 指导教师： |  |
| 完成日期： | 2022年5月25日 |

**论文独创性声明**

本人郑重声明：所呈交的毕业设计（论文），是本人在老师指导下，独立进行研究所取得的成果。论文中除了已经注明引用的内容外，不包含任何他人享有的著作权内容。其他个人和集体对本研究工作的启发和所做出的贡献，均已在论文中以明确的方式标明。如本文被查证有抄袭或剽窃行为，本人愿意承担由此引发的法律后果，并依据学校的规章制度接受相应处理。

**签 名**： **日 期**：

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**论文版权使用授权声明**

本人完全了解上海开放大学关于收集、保存、使用毕业论文的规定，同意如下各项内容：按照学校要求提交学位论文的印刷本和电子版本；学校有权保存学位论文的印刷本和电子版，并采用影印、缩印、扫描、数字化或其它手段保存论文；学校有权提供目录检索以及提供本论文全文或者部分的阅览服务，以及出版学位论文；学校有权按有关规定向国家有关部门或者机构送交论文的复印件和电子版；在不以赢利为目的的前提下，学校可 以适当复制论文的部分或全部内容用于学术活动。

**签 名**： **日 期**：

**目 录**

[摘 要 I](#_Toc22511)

[Abstract II](#_Toc11542)

[一、 绪论 1](#_Toc28299)

[（一）课题背景和意义 1](#_Toc15325)

[（二）共享自习室中的预约管理体系的重要性 1](#_Toc24965)

[1. 提高管理效率、降低人力成本 1](#_Toc16294)

[2. 实时更新自习室座位情况 2](#_Toc4136)

[（三） 课题研究内容 2](#_Toc5471)

[（四）论文组织结构 2](#_Toc15162)

[二、 系统可行性研究 4](#_Toc7193)

[（一） 经济可行性 4](#_Toc26971)

[（二） 技术可行性 4](#_Toc15462)

[（三） 进度可行性 5](#_Toc13187)

[（四） 系统可行性分析总结 6](#_Toc23391)

[三、 系统需求分析 7](#_Toc25184)

[（一）功能需求 7](#_Toc7952)

[1. 用户登录与注册： 7](#_Toc19352)

[2. 用户查看/查找历史创建自习室 7](#_Toc22308)

[3. 创建/加入/收藏自习室 7](#_Toc5702)

[4. 用户与管理员/用户的沟通 8](#_Toc13311)

[5.个人中心 8](#_Toc2086)

[（二）系统性能需求分析 8](#_Toc1967)

[四、 系统概要设计 10](#_Toc18759)

[(一) 系统结构设计 10](#_Toc26457)

[(二) 系统功能模块设计 10](#_Toc14916)

[五、系统数据库的分析与设计 12](#_Toc30986)

[（一） 数据关系模型及分析 12](#_Toc6485)

[（二） 数据库表结构及分析 12](#_Toc22803)

[1. 学生用户表 (stuents) 12](#_Toc21673)

[2. 未创建自习室表（rooms） 13](#_Toc7011)

[3. 已创建自习室表（hasrooms） 13](#_Toc7413)

[4. 已预约自习室表（hasRoomLists） 14](#_Toc20592)

[5. 待审核申请表（reviewRoomList） 14](#_Toc8476)

[6. 已收藏教室表（collectRoomList） 15](#_Toc29362)

[7. 段位榜单表（praiseStuLists） 15](#_Toc25680)

[8. 聊天信息表（praiseStuLists） 15](#_Toc2244)

[（三） 系统功能结构设计 16](#_Toc27179)

[1. 系统功能设计全景 16](#_Toc27036)

[（四）系统主要页面设计及说明 17](#_Toc4231)

[1. 界面设计风格 17](#_Toc15230)

[2. 登录页面 17](#_Toc27936)

[3. 注册页面 19](#_Toc19351)

[4. 首页页面 20](#_Toc30024)

[5. 自习室详细页面 21](#_Toc2917)

[6. 个人中心页面 23](#_Toc10289)

[六、系统测试 30](#_Toc10832)

[（一）系统测试方式 30](#_Toc30879)

[（二） 系统测试 30](#_Toc13602)

[1. 黑盒测试 30](#_Toc22775)

[2. 白盒测试 32](#_Toc12874)

[（三） 测试总结 33](#_Toc9486)

[七、结束语 34](#_Toc10495)

[参考文献 35](#_Toc16162)

[致 谢 36](#_Toc12999)

**摘 要**

伴随着科学技术的突飞猛进，社会也在快速发展，信息和知识也在增长，知识的更新周期变得越来越短，人们创新的速度加快，学习型社会正在快速形成，随着社会的进步，对人们知识素质的要求越来越高，人力资源与教学资源的重要性提高，学习就成为了每个人、每个组织、以及社会的殷切需要。

人们对自学场所的需求越来越迫切，在以往传统的可自习场所中，例如：图书馆、家里、咖啡厅等场所暴露出来的很多问题，如：座位稀缺、置物占座、管理松散、环境吵闹、安全问题突出，疫情管控效果差等影响自习质量的各种问题，预约制度的自习室能为人们提供的更加安静、乾净、有序、舒适的学习环境，这种制度的自习室快速兴起并吸引求学人们的注意力。

在这个后疫情时代，预约制的共享自习室提供了更为安全与方便的学习环境，无需到现场查看座位的情况即可在线预约座位，防止了找座位时跑空的情况发生，减少了人员的聚集，便于管理，更大大提升了自习室的座位使用率。

本论文为自习室提供了一种实用的预约制的解决方案，系统采用了基于NodeJS的前后端分离B/S结构进行开发，前端综合使用了Vue3.0、Ant-Design-Vue，等开发框架，后端综合使用了NodeJs和koa2等。开发工具使用了WebStorm2021,数据库为MongoDB，

实现了线上的自习室座位管理。

**关键词：**自习室；线上预约；Node；B/S；

# Abstract

With the rapid development of science and technology, the society is in the rapid development of information and knowledge is growing, the updating cycle of knowledge is becoming more and more short, the innovation of the speed, the learning society is rapidly to form, with the progress of the society, more and more high to the requirement of people knowledge quality, human resources and the increasing importance of teaching resources, Learning becomes an urgent need for everyone, every organization, and society.

People have an increasingly urgent demand for places for self-study. In the traditional places for self-study, such as libraries, homes, cafes and other places, many problems are exposed, such as: Scarce, rear seat, seat management loose, environmental noise, safety problems, poor disease control results various problems affecting the quality of self-study, study room reservation system can provide people with more quiet, clean, orderly and comfortable study environment, study the system of study area rapid rise and attract people's attention.

In this post-epidemic era, the reservation sharing study room provides a safer and more convenient learning environment. You can reserve seats online without going to the site to check the seat situation, which prevents running out of seats when looking for seats, reduces the gathering of personnel, facilitates management, and greatly improves the seat utilization rate of the study room.

This paper provides a practical reservation system solution for the study room. The system uses the FRONT and back end separation B/S structure based on NodeJS for development. Vue3.0, Ant-Design-Vue and other development frameworks are integrated in the front end, and NodeJS and KOA2 are integrated in the back end. The development tool uses WebStorm2021, the database is MongoDB,

The online seat management of self-study room is realized.

**Key words:** study room; Online booking; Node;B/S;

###### 

# 绪论

**（一）课题背景和意义**

自习场所是我们生活和学习中必然接触到的地方，例如：图书馆，咖啡厅，家里等等，这些地方无不暴露出影响个人学习质量的问题，如图书馆里开馆时间存在限制，无关人员随意进出，制造响动，容易没有座位等。咖啡厅声音异常嘈杂，无人管理等。家里没有学习氛围完全找不到感觉，容易开小差等。

那么，选择去传统的公共共享自习室就能解决所有的问题吗？传统的公共共享自习室也存在着很多不可忽视的问题，座位跑空、无人管理、学习分为差等等，最具代表性的问题是以各种手段来占座的问题，如果没有很好的管理手段，会出现很多物品在人不在的浪费座位的现象。

所以，为了改善部分问题，提高教学资源整合的工作效率，强化管理，开发了自习帮-校园自习室预约系统。

## **（二）**共享自习室中的预约管理体系的重要性

如今互联网技术飞速发展，我们可以利用此技术，改善传统式自习室的上座制度，自习帮-校园自习室预约系统是一套高效的管理自习室座位的解决方案，其重要性主要体现在以下两个方面：

### 提高管理效率、降低人力成本

不少传统的公共共享自习室并没有配备自习室座位预约系统，这就给自习室管理员添增了许多的不应存在的管理难度。例如自习室座位的线上预约，未来也可以开放核销与放行、计算费用、客流经营数据统计等，这样功能都让人工手动登记会耗费巨量的人力资源，拉低了员工工作效率，增加了许多管理难度。但如果由预约系统来登记座位与记录各种预约信息则可大幅节省人力成本。

### 实时更新自习室座位情况

只需登陆网站即可实时了解每个自习室的使用情况和座位情况，无需自己前往自习室查看有没有座位，从而彻底消灭了座位跑空从而浪费时间的情况，既节省了用户的时间，又方便了自习室管理。

## 课题研究内容

本课题将讨论基于NodeJS语言的Koa2框架座位开发后台，使用Vue3.0框架与Ant-Design-Vue设计并搭建前台界面，开发一个B/S架构，主要为学校共享自习室使用的预约系统，为使用自习室的学生或其他用户提供：在线查询自习室座位情况、在线预约等一系列服务。实现对校园公共共享自习室的信息化座位预约。

**主要实现的模块如下：**

1. 用户登录与注册：分为了登录、和注册两个子模块；
2. 用户查看/查找历史创建自习室：可以通过自习室地点、楼层、以及创建自习室的时间进行筛选查找自习室/历史自习室的自习室信息；
3. 创建/加入/收藏自习室：分为创建、加入、以及收藏三个子模块；
4. 用户与用户的沟通：用户与创建了自习室的管理员或者与其他用户进行聊天沟通；
5. 个人中心：个人中心分为自习提醒、已预约自习室、待审核自习室、自习室收藏夹、个人资料展示及修改、自习室段位排行榜、对话/申请以及退出账号八个功能；

## **（四）**论文组织结构

**本系统的论文的内容组织结构包括以下七大部分：**

第一部分：绪论部分。主要介绍了本系统的开发背景以及意义、课题研究的内容、论文的组织结构等基本信息。

第二部分：系统可行性分析。主要论述了对本系统的经济、技术、进度的可行性分析。

第三部分： 需求分析。主要论述了对本系统的功能需求分析以及系统性能需求分析。

第四部分： 系统概要设计。主要论述了本系统的功能模块的设计和系统的结构设计。

第五部分： 系统数据库的分析与设计。主要论述了数据库表ER图和数据库表的创建以及设计实现。

第六部分： 系统测试。主要论述了通过软件测试用例来测试系统软件的健壮性及稳定性，对整个系统的功能测试等。

第七部分： 系统总结。主要论述了对本系统论文的研究工作和内容进行总结，并在此感谢指导老师的教导与帮助。

# 系统可行性研究

1. **经济可行性**

本公共共享自习室预约系统开发主要有前期的开发成本、后期的维护成本、测试成本以及引流等等，虽然本系统可能与产生经济效益有一定的距离，但是从长远得利益来看，本系统的设计还是拥有比较合理的经济可行性。

公共共享自习室预约系统可以方便用户找到座位，避免各种在抢座位上发生的问题，这种便利可以为自习室带来更多的用户。也可以大幅的减少人工介入的成本，尤其是座位使用期间，自习室管理员可以减少对各个教室的巡查次数，一定程度的节省了管理方面的人力资源。该系统所产生的效益也大于本人在开发系统中投入的人力与时间成本。

从成本与效益来看，本系统的研发比较具备良好的经济可行性。

1. **技术可行性**

本预约系统是以Vue3.0作为前端编程语言，以NodeJS作为后端的技术支持，使用了Koa2框架，并且结合MongoDB数据库作为数据储存的公共共享自习室预约系统。

* **B/S结构：**

B/S模式又称B/S结构，是Web兴起后的一种网络结构模式。Web浏览器是客户端最主要的应用软件。这种模式统一了客户端，将系统功能实现的核心部分集中到服务器上，大大简化了系统的开发、维护与使用；客户只需要安装一个浏览器，服务器上安装数据库；浏览器经过网站服务器与数据库作数据交互。

* **Vue3.0框架：**

Vue 是用于构建用户界面的渐进式框架。与另外的大型框架不一样的是，Vue 被设计为可以自底向上逐层应用。Vue 的核心库只关注视图层，不仅非常易于上手，还方便与第三方库或者其他项目进行整合工作。

* **WebStorm2021编辑器：**

本系统开发时，开发工具选用了webStorm2021，WebStorm 是[JetBrains](https://baike.baidu.com/item/JetBrains/7502758" \t "https://baike.baidu.com/item/WebStorm/_blank)旗下的一款[JavaScript](https://baike.baidu.com/item/JavaScript /321142" \t "https://baike.baidu.com/item/WebStorm/_blank)开发工具。它已经被广大[中国](https://baike.baidu.com/item/%E4%B8%AD%E5%9B%BD/1122445" \t "https://baike.baidu.com/item/WebStorm/_blank)的JavaScript[开发者](https://baike.baidu.com/item/%E5%BC%80%E5%8F%91%E8%80%85/239674" \t "https://baike.baidu.com/item/WebStorm/_blank)誉为“[前端的开发](https://baike.baidu.com/item/%E5%89%8D%E7%AB%AF%E5%BC%80%E5%8F%91/10009024" \t "https://baike.baidu.com/item/WebStorm/_blank)神器”、“最强大的编辑器”、“最强最为智能的JavaScript[IDE](https://baike.baidu.com/item/ IDE/8232086" \t "https://baike.baidu.com/item/WebStorm/_blank)”等等。

* **MongoDB 数据库：**

本系统的数据库系统使用了MongoDB，MongoDB是一个基于分布式文件存储的数据库。为WEB应用提供了可扩展的高性能数据储存的方案。是非关系数据库中功能最为丰富的数据库，最与关系数据库相似的。MongoDB最大的特点是它支持的查询语言非常强大，其语法类似于面向对象的查询语言，可以实现与关系数据库单表查询的绝大部分相似的功能，并且也支持对数据建立[索引](https://baike.baidu.com/item/%E7%B4%A2%E5%BC%95" \t "https://baike.baidu.com/item/mongodb/_blank)。

综上所述，本系统在技术方面具有良好的可行性。

1. **进度可行性**

在本次开发本系统前，应该提前制定系统开发计划时间进度表(预计2个月)，开发进度明细需见下表1所示。按照每个模块，科学的、合理的制作工作节点，可以使开发尽可能根据计划进度执行，顺利完成任务。所以通过合理化制定的工作节点和工作效率的角度来看，进度表的设计是可行的。

表1 测试流程管理系统开发进度表

| 阶段 | 主要工作 | 计划起始时间 | 计划终止时间 | 实际起始时间 | 实际终止时间 | 工作天数（天） | 执行  结果 |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 系统需求 | 调研需求及环境 | 2/10 | 2/11 | 2/10 | 2/11 | 1 | 通过 |
| 制定初步计划 | 2/12 | 2/14 | 2/13 | 2/15 | 3 | 通过 |
| 需求分析 | 确认系统运行环境 | 2/16 | 2/17 | 2/16 | 9/17 | 2 | 通过 |
| 建立系统功能及性能要求 | 2/18 | 2/19 | 2/18 | 2/19 | 2 | 通过 |
| 测试计划 | 2/19 | 2/21 | 2/19 | 2/21 | 2 | 通过 |
| 确定开发计划 | 2/22 | 2/23 | 2/22 | 2/23 | 2 | 通过 |
| 概要设计 | 建立系统结构，划分功能模块 | 2/24 | 2/26 | 2/24 | 2/26 | 3 | 通过 |
| 定义功能模块接口 | 2/27 | 3/2 | 2/27 | 3/2 | 4 | 通过 |
| 数据库设计与实现 | 3/3 | 3/7 | 3/3 | 3/7 | 5 | 通过 |
| 详细设计 | 设计各模块具体实现 | 3/8 | 3/10 | 3/8 | 3/10 | 3 | 通过 |
| 确定模块间详细接口 | 3/11 | 3/15 | 3/11 | 3/15 | 5 | 通过 |
| 实现 | 编写程序源代码 | 3/16 | 3/26 | 3/16 | 3/26 | 11 | 通过 |
| 进行模块调试 | 3/27 | 4/2 | 3/27 | 4/2 | 7 | 通过 |
| 系统测试 | 执行集成测试计划 | 4/3 | 4/7 | 4/3 | 4/7 | 5 | 通过 |
| 测试整个软件系统 | 4/8 | 4/10 | 4/8 | 4/10 | 3 | 通过 |
| 维护 | 纠正错误，完善应用 | 4/11 | 4/15 | 4/11 | 4/15 | 5 | 通过 |
| 对修改进行配置管理 | 4/16 | 4/20 | 4/16 | 4/20 | 5 | 通过 |
| 合计 | —— | —— | —— | —— | —— | 68 | 通过 |

1. **系统可行性分析总结**

进行系统可行性分析时，需要解析现有的系统，并以概括的形式表达出对现有系统的认知。在通过对本系统的初步分析后，在经济可行性、技术可行性、进度可行性的研究分析以及对比系统的主客观条件中得出，本系统满足以上可行性分析的条件。

# 系统需求分析

## （一）功能需求

基于第一章绪论中的分析，当前公共共享自习室中主要存在以下问题：

1. 书本等物品占座，人却不在自习室中自习的问题；
2. 前往未知情况的自习室容易跑空没座位浪费时间的问题；
3. 不易隔开座位管控疫情的问题；

针对以上研究得出的问题，本系统计划开发一种在线预约自习室座位的预约平台，能帮助用户节省时间成本，提高公共共享自习室的上座率与秩序，从而实现既方便用户进行自习又能对公共共享自习室进行科学的管理。

因此，本系统分为以下五大模块：

### 用户登录与注册：

分为用户登录、和用户注册两个子模块。

* 用户登录：用户进行登录的功能；
* 用户注册：用户进行注册的功能；

### 用户查看/查找历史创建自习室

* 查询自习室列表：查询所有自习室的使用状态的功能。
* 筛选自习室列表：可以通过自习室地点、楼层、以及创建自习室的时间查找或查看历史创建自习室的功能。

### 创建/加入/收藏自习室

分为创建、加入、以及收藏三个子模块

* 创建自习室：用户进行创建无审核/有审核制度的自习室的功能。
* 加入自习室：用户加入无审核/有审核制度的自习室的功能。
* 收藏自习室：用户收藏自习室的功能。

### **用户与管理员/用户的沟通**

与自习室管理员进行对话：与本间自习室的管理进行沟通对话的功能。

* 与其他用户进行对话：与本间自习室的已预约的用户进行沟通对话的功能。

### 5.个人中心

个人中心分为自习提醒、已预约自习室、待审核自习室、自习室收藏列表、个人资料展示及修改、自习室段位排行榜、对话/申请以及退出账号八个子模块：

* 自习提醒：提醒用户今日预约的自习室的位置的功能。
* 已预约自习室：展示已预约自习室的历史与现在的列表与操作列表的功能。
* 待审核自习室信息：展示待审核自习室信息的历史与现在的列表与操作列表的功能。
* 已收藏自习室列表：展示已收藏的自习室的历史与现在的列表与操作列表的功能。
* 个人资料展示及修改：修改用户的个人资料的功能。
* 自习室段位排行榜：查看与点赞自习室段位排行榜的功能。
* 对话/申请：接收与操作加入自习室的审核请求与接收聊天信息的功能。
* 退出账号：退出登陆状态的功能。

## （二）系统性能需求分析

**预约系统使用后，应达到以下系统性能标准：**

1. 页面设计要求简约大方、操作容易、功能完整。
2. 规范与完善的各方面基础信息设置。
3. 支持多人同时在线使用。
4. 系统正常运行不卡顿、没有安全隐患。
5. 具有良好的可维护性和易操作性。

# 系统概要设计

### 系统结构设计

本系统采用的是B/S架构，服务器数据库存放着全部的业务数据，所有用户通过web浏览器来访问本系统中的数据。

如图 4.1系统结构图所示。

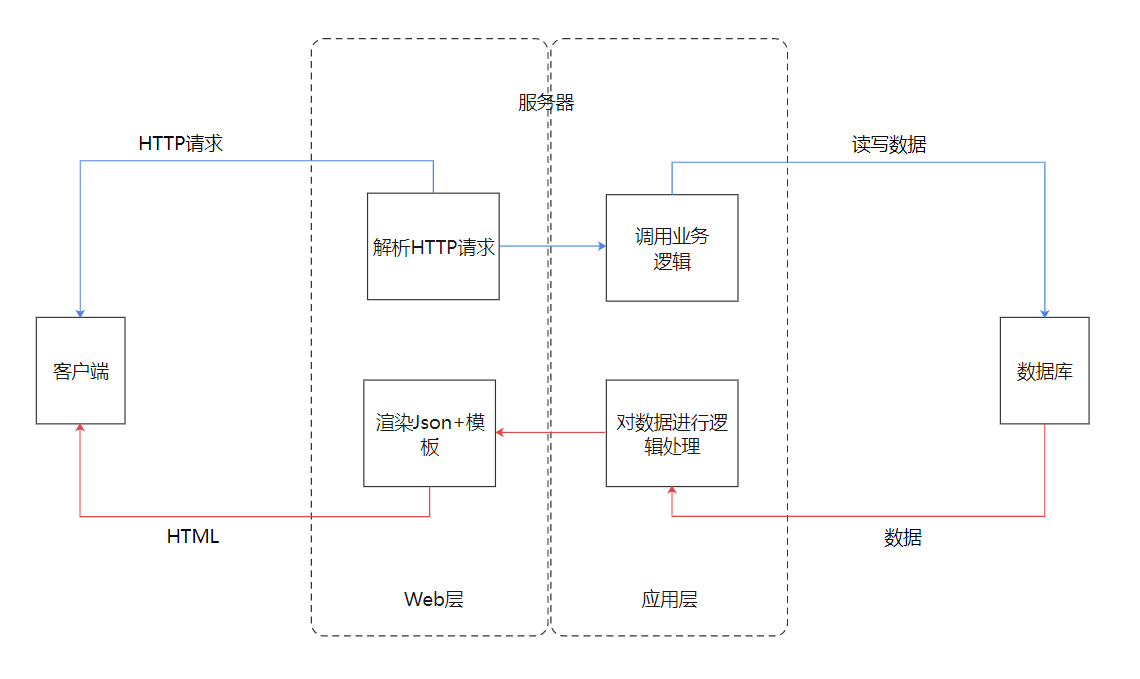


图4.1 系统结构图

### 系统功能模块设计

本系统主要分为五大功能模块，它的主要功能包括有：登陆与注册、查询自习室（浏览自习室列表、筛选自习室列表）、自习室内操作（创建自习室、加入有审核制度自习室、加入无审核制度自习室、与管理员的沟通、用户间的沟通）、工人中心（自习提醒、预约自习室列表、待审核信息列表、已收藏的自习室列表、用户资料修改、退出登录），其功能模块如图4.2所示。



图4.2 系统功能模块结构图

# **五、系统数据库的分析与设计**

## 数据关系模型及分析

为了实现以上设计，我们需要设计一套完整的数据库系统，其中最重要的是数据关系模型。它是整个自习室预约系统的第一个核心部分，用于阐述各个实体对象间彼此的联系。我们系统的处理效率以及最终实现的效果是否能达到设计要求与数据关系模型的设计密切相关。

通过对整个系统及各个子程序模块的分析，得到如图5.1所示数据关系模型：

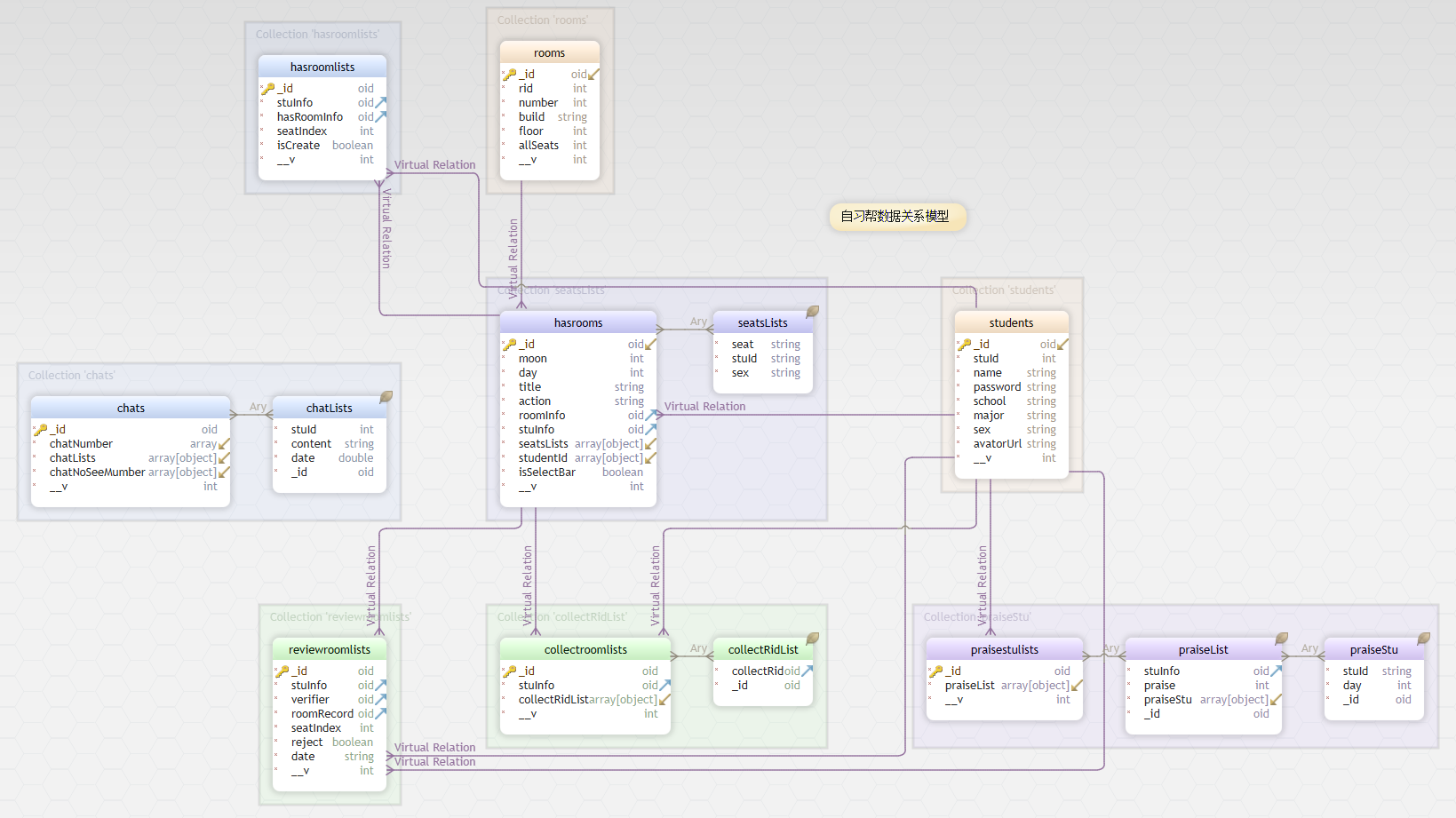


图5.1 数据模型

## 数据库表结构及分析

本系统采用MongoDB作为数据库管理系统，

本系统一共设计了八张数据库表，它们的具体的定义如下：

### 学生用户表 (stuents)

学生用户表主要用于储存用户的基础信息。如表5.1所示。

表5.1 students数据表字段解释（学生用户集合）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 是否为主键 | 是否为外键 | 是否允许为null | 说明 |
| 1 | \_id | ObjectId | 是 | 否 | 否 | 默认主键 |
| 2 | stuId | Number | 否 | 否 | 否 | 学生用户的学号 |
| 3 | name | String | 否 | 否 | 否 | 学生用户的姓名 |
| 4 | password | String | 否 | 否 | 否 | 学生用户的密码 |
| 5 | school | String | 否 | 否 | 否 | 学生用户的学校名称 |
| 6 | major | String | 否 | 否 | 否 | 学生用户的所学专业 |
| 7 | sex | String | 否 | 否 | 否 | 学生用户的性别 |
| 8 | AvotorUrl | String | 否 | 否 | 否 | 学生用户的头像 |

### 未创建自习室表（rooms）

未创建自习室表主要用于储存自习室原始的基础信息。如表5.2所示。

表5.2 rooms数据表字段解释（未创建自习室集合）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 是否为主键 | 是否为外键 | 是否允许为null | 说明 |
| 1 | \_id | ObjectId | 是 | 否 | 否 | 默认主键 |
| 2 | rid | Number | 否 | 否 | 否 | 自习室的id |
| 3 | number | Number | 否 | 否 | 否 | 自习室号 |
| 4 | build | String | 否 | 否 | 否 | 自习室地点 |
| 5 | floor | Number | 否 | 否 | 否 | 自习室楼层 |
| 6 | allSeats | Number | 否 | 否 | 否 | 自习室的座位数 |

### 已创建自习室表（hasrooms）

已创建自习室表主要用于储存创建自习室的记录信息。如表5.3所示。

表5.3 hasrooms数据表字段解释（已创建自习室集合）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 是否为主键 | 是否为外键 | 是否允许为null | 说明 |
| 1 | \_id | ObjectId | 是 | 否 | 否 | 默认主键 |
| 2 | roomInfo | ObjectId | 否 | 是 | 否 | 自习室的加密ID（关联rooms表） |
| 3 | stuInfo | ObjectId | 否 | 是 | 否 | 创建自习室的用户ID（关联students表） |
| 4 | seatLists | Array | 否 | 否 | 否 | 加入自习室的用户个人信息 |
| 5 | moon | Number | 否 | 否 | 否 | 创建自习室的月份 |
| 6 | day | Number | 否 | 否 | 否 | 创建自习室的日 |
| 7 | title | String | 否 | 否 | 否 | 已创建自习室的标题 |
| 8 | action | String | 否 | 否 | 否 | 已创建自习室的简介 |
| 9 | roomAudit | Boolean | 否 | 否 | 否 | 是否开启管理员审核制度 |

### 已预约自习室表（hasRoomLists）

已预约自习室表主要用于储存用户预约的已创建自习室的信息。如表5.4所示。

表5.4 hasRoomLists数据表字段解释（已预约自习室记录表）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 是否为主键 | 是否为外键 | 是否允许为null | 说明 |
| 1 | \_id | ObjectId | 是 | 否 | 否 | 默认主键 |
| 2 | stuInfo | ObjectId | 否 | 是 | 否 | 预约自习室的学生的id（关联students表） |
| 3 | hasRoomInfo | ObjectId | 否 | 是 | 否 | 已预约自习室的ID（关联hasroom表） |
| 4 | seatIndex | Number | 否 | 否 | 否 | 已预约自习室的座位号 |
| 5 | isCreate | Boolean | 否 | 否 | 否 | 是否为已预约自习室的管理员 |

### 待审核申请表（reviewRoomList）

待审核申请表主要用于储存用户申请加入某已创建自习室的请求记录信息。如表5.5所示。

表5.5 reviewRoomList数据表字段解释（待审核申请表）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 是否为主键 | 是否为外键 | 是否允许为null | 说明 |
| 1 | \_id | ObjectId | 是 | 否 | 否 | 默认主键 |
| 1 | stuInfo | ObjectId | 否 | 是 | 否 | 发起审核请求的学生ID |
| 2 | verifier | ObjectId | 否 | 是 | 否 | 负责接收请求的管理员ID |
| 3 | roomRecord | ObjectId | 否 | 是 | 否 | 待审核自习室ID |
| 4 | seatIndex | Number | 否 | 否 | 否 | 待审核自习室座位号 |
| 5 | reject | Boolean | 否 | 否 | 否 | 是否已经被管理员拒绝加入 |
| 6 | date | String | 否 | 否 | 否 | 时间戳 |

### 已收藏教室表（collectRoomList）

已收藏教室表主要用于储存用户收藏的自习室的记录信息。如表5.6所示。

表5.6 collectRoomList数据表字段解释（收藏教室列表集合）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 是否为主键 | 是否为外键 | 是否允许为null | 说明 |
| 1 | \_id | ObjectId | 是 | 否 | 否 | 默认主键 |
| 1 | stuInfo | ObjectId | 否 | 是 | 否 | 收藏自习室的学生id（关联students表） |
| 2 | collectRidList | Array | 否 | 否 | 否 | 收藏的自习室Id数组 |
| 3 | collectRidList.collectRid | ObjectId | 否 | 是 | 否 | 收藏的自习室ID（关联hasRoom表） |

### 段位榜单表（praiseStuLists）

段位榜单表主要用于储存段位榜单的排列数据信息。如表5.7所示。

表5.7 praiseStuLists 数据表字段解释（段位榜单集合）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 是否为主键 | 是否为外键 | 是否允许为null | 说明 |
| 1 | stuInfo | ObjectId | 是 | 否 | 否 | 自习室榜单的学生id（关联students表） |
| 2 | praise | Number | 否 | 否 | 否 | 点赞次数 |
| 3 | praiseStuLists | Array | 否 | 否 | 否 | 给用户点赞的用户的列表 |
| 4 | praiseStuLists.stuId | String | 否 | 否 | 否 | 点赞人ID |
| 5 | praiseStuLists.day | String | 否 | 否 | 否 | 点赞的日期(当前月数的几号) |

### 聊天信息表（praiseStuLists）

聊天信息表主要用于储存用户的消息记录信息。如表5.8所示。

表5.8 chats数据表字段解释（聊天信息集合）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 是否为主键 | 是否为外键 | 是否允许为null | 说明 |
| 1 | \_id | ObjectId | 是 | 否 | 否 | 默认主键 |
| 1 | chatNumber | array | 否 | 否 | 否 | 聊天双方的id |
| 2 | chatNoSeeMumber | array | 否 | 否 | 否 | 没有看到信息的人的id |
| 3 | chatLists.stuId | Number | 否 | 否 | 否 | 学生的学号（id） |
| 4 | chatLists.content | string | 否 | 否 | 否 | 双方的聊天内容 |
| 5 | chatLists.date | string | 否 | 否 | 否 | 时间戳 |

## 系统功能结构设计

### 系统功能设计全景

表5.9 系统模块功能表

|  |  |  |  |
| --- | --- | --- | --- |
| **一级模块** | **二级模块** | **功能分类** | **功能描述** |
| 登陆首页 | 登陆与注册 | 登陆 | 用户登陆。 |
| 注册 | 用户输入注册信息进行注册。 |
| 主界面 | 查询自习室 | 浏览自习室列表 | 进入主页查看所有的自习室状态。 |
| 筛选自习室列表 | 根据时间、楼层、楼号条件进行筛选自习室列表。 |
| 自习室界面 | 创建自习室 | 用户创建自习室 | 进入未使用的自习室后点击座位，即可输入自习室标题、自习室描述、自习室是否采用审核即可创建自习室。 |
| 加入自习室 | 用户加入有审核制度的自习室 | 进入有审核制的自习室后，点击座位后弹出的加入按钮，即可把加入自习室申请发送至自习室管理员。 |
| 用户加入无审核制度的自习室 | 进入无审核制度的自习室后，点击座位后即可加入该自习室。 |
| 用户聊天 | 用户与管理员之间的沟通 | 进入自习室后，点击管理员头像后，即可跳转至聊天页面进行沟通。 |
| 用户之间的沟通 | 点击已经有用户预约的座位后，弹出用户信息与前往聊天的按钮，点击即可进入与当前用户沟通。 |
| 个人中心界面 | 自习室信息管理 | 今日自习提醒 | 提示今日是否已经预约过，以及预约过的自习室位置。 |
| 已预约自习室 | 记录历史已预约自习室的记录，点击前往按钮即可进入该自习室。 |
| 已收藏自习室 | 记录历史已收藏的自习室的列表，点击前往按钮即可进入该自习室。 |
| 信息管理 | 待审核信息 | 发送给管理员的申请加入自习室的审核信息记录列表。 |
| 管理自习室申请与聊天消息 | 接收加入自习室的申请，点击同意/拒绝按钮处理申请，也接收新的聊天消息提示，点击进入聊天室进行沟通。 |
| 用户权限管理 | 用户资料修改 | 进行用户名、密码等的资料修改。 |
| 退出登陆 | 点击退出登录按钮即可清除登陆状态，跳转至登陆页面。 |

## （四）系统主要页面设计及说明

### 界面设计风格

界面以黄色和绿色为主色调，清新且富有活力；整个界面图标用的是MBE风格，通过线框型Q版卡通画进行演变的MBE风格，设计上更是采用了更为大粗的描边处理，与没有描边的扁平化风格插画进行比对中，去除了不必要的多余色块区分，更简单、更易用、易辨认。粗线条进行的描边更是起到了对界面与外界的绝对隔绝，更加的突显内容、表现清晰、化繁为简。这般卡通的形象、清新的颜色使用户能更加轻松愉悦的进行进一步的学习。

### 登录页面

**流程说明：**

用户输入用户名和密码后点击确定按钮，系统则开始对用户名与密码进行校验，正确进入后进入首页页面。登录功能的流程如下5.2图所示。

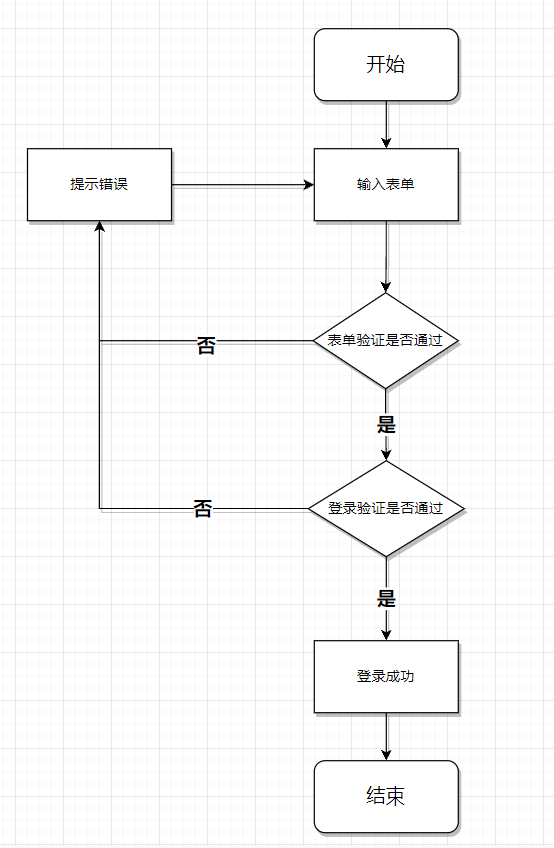


图5.2 登录功能流程图

**页面说明：**

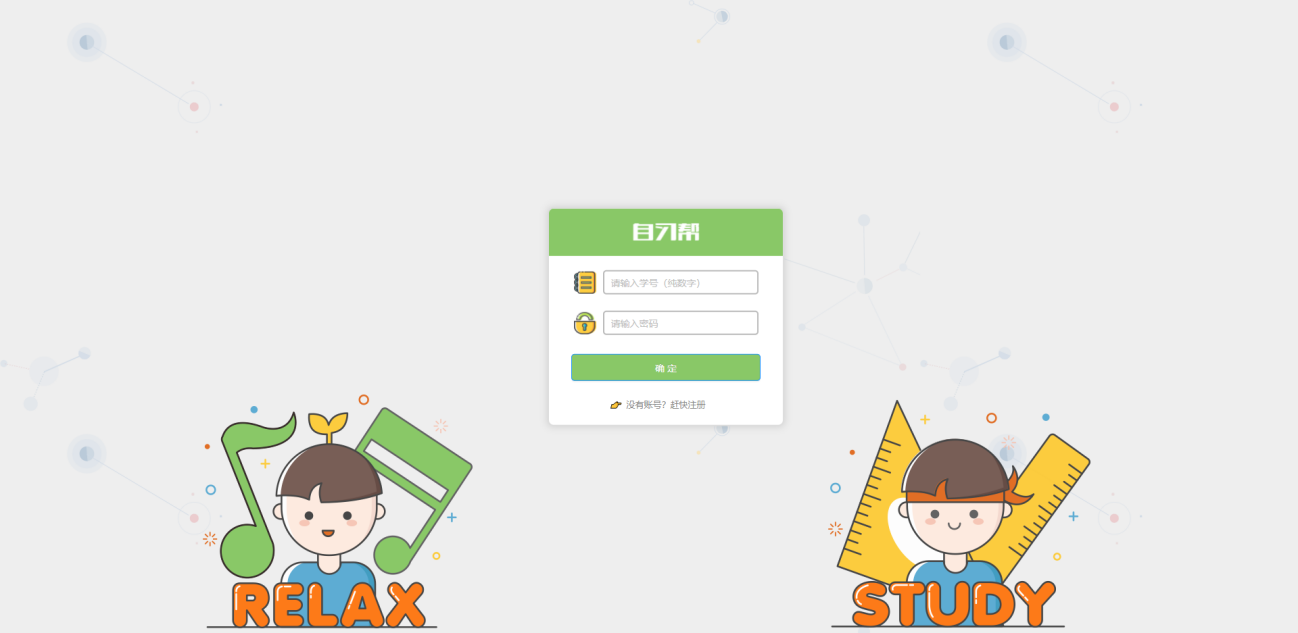
登录页面中，主要需要对表单进行表单验证的工作，验证工作在浏览器端通过JavaScript进行。在点击确定按钮或填写信息时进行校验，如果出错则发出提示。

图5.3 登录页面图

### 注册页面

**流程说明：**

用户通过输入自己的相关个人信息及学号和密码后，点击注册按钮，系统校验用户输入的信息是否符合格式，如果通过校验，则进行注册校验，判断学号是否已经注册，如果没有问题，则注册成功。注册功能的流程图如下5.3图。

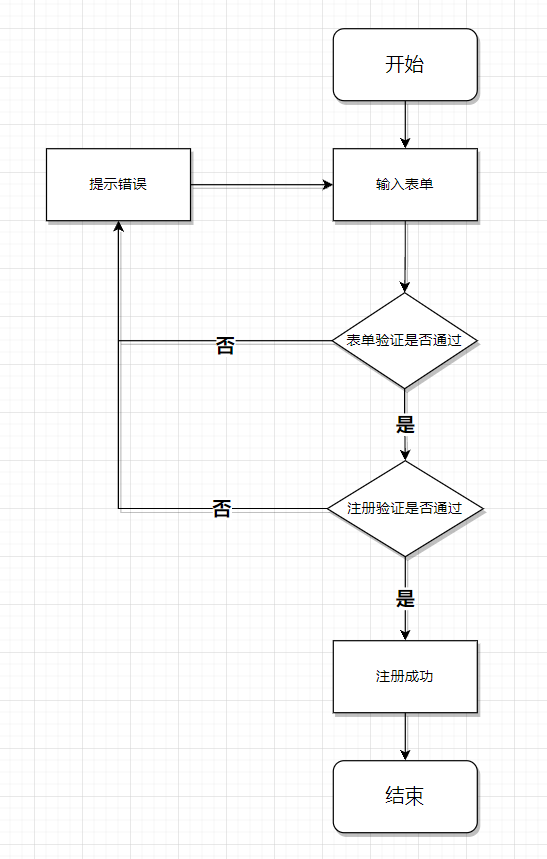


图5.4 注册功能流程图

**页面说明：**

在登录页面时，可以点击“没有账号？赶快注册”按钮，即可进入注册页面。表单输入后经过表单验证与注册验证，如果不通过则提示错误，如果通过则提示注册成功，跳转至登录页面。

具体的页面图如下图5.5所示。

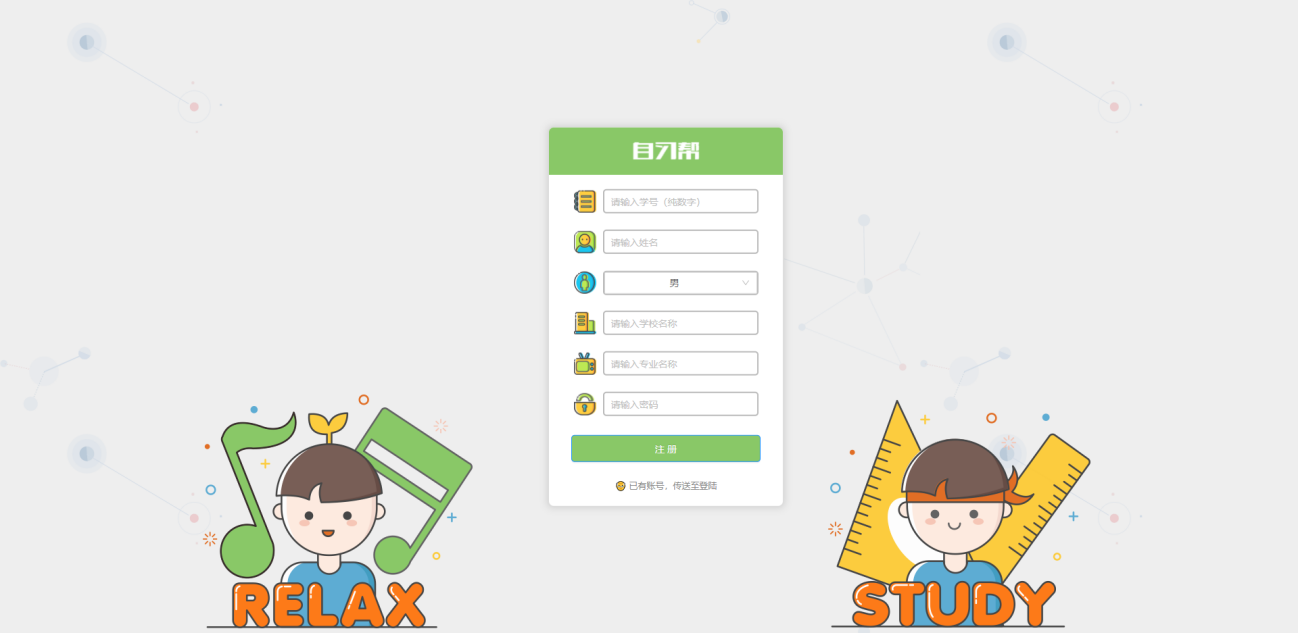


图5.5 注册页面图

### 首页页面

**流程说明：**

登录成功后则会进入首页页面，经过筛选后得出自己想要的教室列表，即可点击进入对应的自习室详细页面。

**页面说明：**

在首页页面，可通过对地点、楼层、时间的选择来进行对历史创建与未来的自习室进行筛选教室列表，未被创建的自习室呈灰色，已被创建的自习室呈绿色并且标示出创建人、教室座位使用概况、主题、自习室详情，点击即可进入点击的对应自习室的详情页面。

右上角的按钮从左往右依次为：

聊天信息与审核信息提示按钮，点击则进入对话/申请页面；

已预约教室列表按钮，点击则进入已预约的教室列表；

个人中心按钮，点击则进入个人中心首页；

具体的页面图如下图5.6所示。

图5.6首页页面图

### 自习室详细页面

**流程说明：**

在首页页面中点击自习室项之后，进入对应的自习室详细页面。用户通过点击自己所选的座位后，弹出当前座位的弹出框，如果这个自习室还未被创建过，则只能选择创建教室按钮，点击创建教室按钮后，选择是否开启审核制度后，如果选择不开启则创建一个无管理员自习室；如果选择开启审核制度，则创建自习室的用户作为该自习室的管理员，输入标题与自习室简介后则成功创建自习室。

如果该自习室已经被创建过，则可以选择加入自习按钮与收藏自习按钮。点击加入自习按钮后，如果该自习室为不启用审核制度的自习室，则直接加入该自习室。如果自习室为启用了审核制度的自习室，则加入的审核发送至本自习室的管理员处，由管理员处理申请后即可加入自习室。

**页面说明：**

未创建自习室座位都呈灰色，并且管理员信息显示无审核管理员。已创建自习室座位有颜色则是那个位置已经有人上座，如果该自习室为无审核制度自习室，管理员信息显示无审核管理员。如果为有审核制度自习室，则管理员信息为当前管理员的用户名，点击管理员的用户名，即可快速跳转至与当前管理员的聊天界面。

已被创建自习室中，点击空座位时，弹出弹窗，显示当前座位号，与加入/创建/收藏教室的操作按钮。如果点击已经有人上座的座位时，显示上座用户的基本信息，点击前往聊天按钮，即可与当前用户进行私聊。

未创建自习室中，点击空座位则只能选择创建自习室按钮，按照意愿创建自习室。

详细页面如下图5.7/5.8/5.9/5.10图所示。

图5.7未被创建的自习室



图5.8点击空座位时弹出框

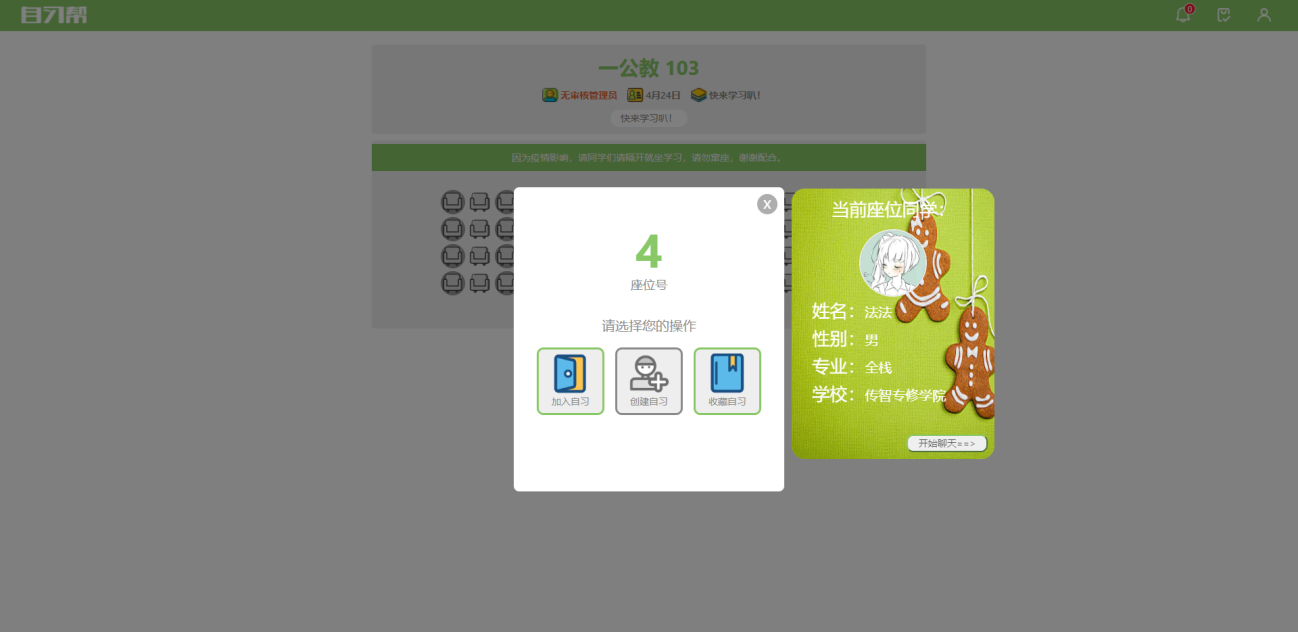


图5.9点击有用户上座的座位时弹出框



图5.10创建过的自习室

### 个人中心页面

#### **1.1个人中心首页**

**流程说明：**

点击右上角的小人形状的按钮，即可进入个人中心首页。个人中心包括自习提醒、已预约自习室列表、待审核自习室列表、收藏自习室列表、对话与申请列表、段位排行榜、修改个人资料功能。

**页面说明：**

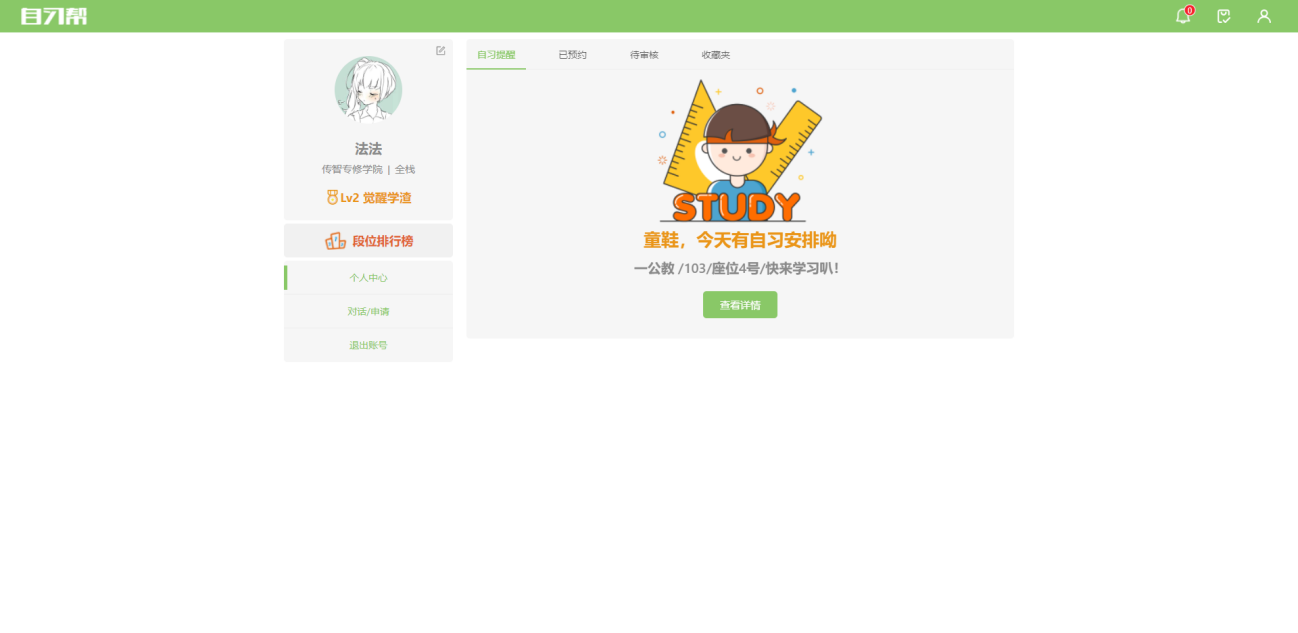
当前页面即是自习提醒功能，进入个人中心时，如果用户今天又预约自习室，则显示出预约的自习室信息，点击查看详情按钮即可进入自习室详情查看。具体的页面图如下图5.11所示。

图5.11个人中心/自习提醒

#### **1.2已预约自习室列表**

**流程说明：**

进入个人中心后，点击上方tab栏的已预约按钮，即可显示已预约自习室列表。

点击前往详情按钮时，跳转至当前选择的已预约自习室的自习室详情中。

点击删除按钮时，如果当前自习室为有审核制度自习室，并且自己是自习室管理员时，无法删除当前自习室，如果自己不是自习室管理员时，则删除自己在自习室中列表的信息，已经当前预约列表的信息。如果当前自习室为无审核制度自习室时，点击按钮则删除自己在自习室的列表信息以及当前预约列表的信息，但是你删除后教室中没有任何人处于上座状态，则当前自习室重新回到未创建自习室状态。

**页面说明：**

每一条已预约记录显示预约的自习室的预约日期、预约自习室的楼号自习室号、预约的座位号、自习室标题与简介。

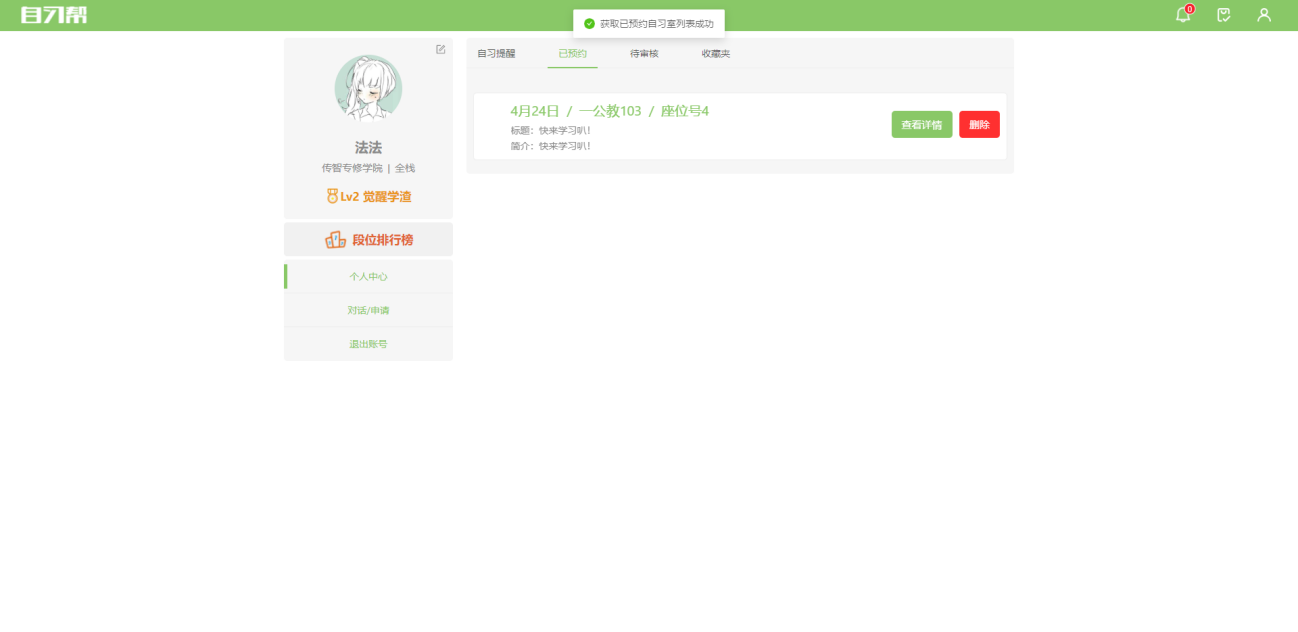
具体的页面图如下图5.12所示。

图5.12个人中心/已预约自习室列表

#### **1.3待审核自习室列表**

**流程说明：**

进入个人中心后点击上方tab栏待审核按钮，即可进入待审核自习室列表中。

点击查看详情按钮后，即可跳转至待审核的当前自习室详情页面中。点击删除按钮后，删除这条待审核的申请信息。

**页面说明：**

在已创建的有审核制度的自习室中加入自习室时，会向自习室管理员发出加入自习室的审核申请信息，并在个人中心的待审核自习室列表中显示该审核申请信息，如果被管理员拒绝则提示“该申请已被管理员无情拒绝”。如果该申请存在超过申请时日期，则会提示“该申请已过期”。如果管理员同意该申请，则审核信息消失，成功加入该自习室中。

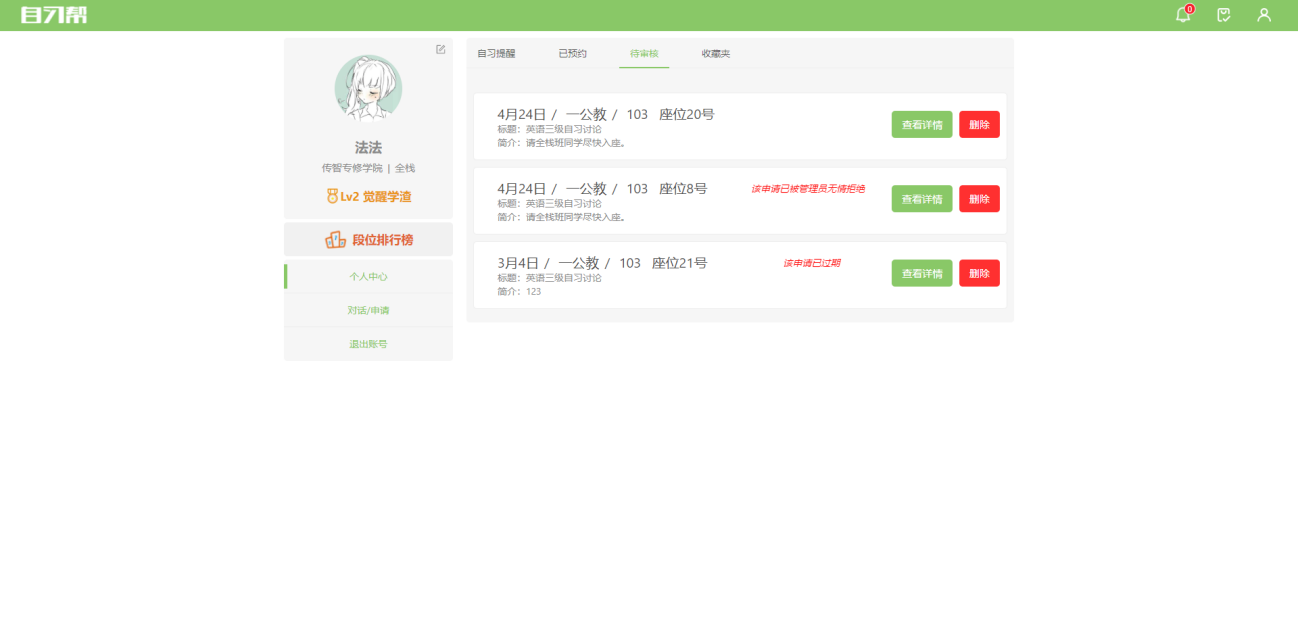
具体的页面图如下图5.13所示

图5.13个人中心/待审核自习室列表

#### **1.4收藏自习室列表**

**流程说明：**

进入个人中心时，点击上方tab栏中的收藏夹按钮，即可进入收藏自习室列表。点击查看详情按钮即可进入收藏的自习室详情页面中。点击删除按钮即可把该收藏信息从收藏自习室列表中删除。

**页面说明：**

每一条收藏记录显示收藏的自习室的日期、收藏自习室的楼号自习室号、自习室标题与简介。

具体的页面图如下图5.14所示。

图5.14个人中心/收藏自习室列表

#### **1.5对话/申请列表**

**流程说明：**

进入个人中心后，点击左边的对话/申请按钮，即可进入对话/申请列表页面。该页面会接收加入自习室的审核申请与聊天信息。

如果是管理员，接收到审核申请信息后，可以点击同意或拒绝按钮处理审核申请。

其他用户向自己发出聊天信息后，对话/申请列表会接收该消息，点击回复后跳转至聊天界面中。

**页面说明：**

对话/申请页面接收到新消息时，右上角的铃铛会提示未处理的信息数量，点击后即可进入对话/申请页面。

具体的页面图如下图5.15所示。

图5.15个人中心/对话/申请列表

#### **1.6修改个人资料**

**流程说明：**

修改表单的数据后，点击保存即可更改个人资料， 点击头像则可以上传图片作为头像。

**页面说明：**

在个人中心页面中，点击左方个人信息卡片的右上角的小按钮或点击头像，即可进入修改个人资料页面。

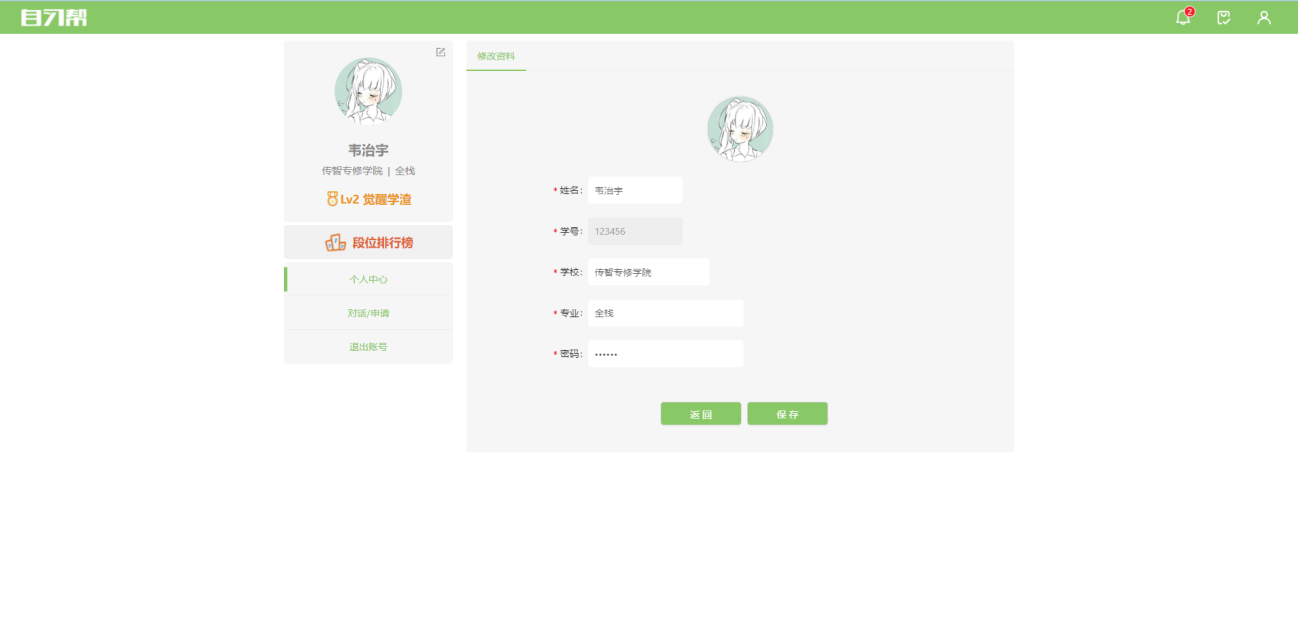
具体的页面图如下图5.16所示。

图5.16个人中心/修改个人资料

#### **1.7段位排行榜**

**流程说明：**

点击排行榜右边的大拇指即可使点赞数量加一。

**页面说明：**

在个人中心页面中，点击段位排行榜按钮即可进入段位排行榜页面。段位排行榜以该用户创建过或加入过的自习室数量为标准进行排列。

具体的页面图如下图5.17所示。



图5.17个人中心/收藏自习室列表

# **六、系统测试**

## （一）系统测试方式

系统测试是系统开发中保证了系统开发的稳定性与正确性的重要手段之一，其目的则是为了发现系统中隐藏的各种问题，校验本系统是否满足预期成果与实际成果的差别。

本系统在系统测试中主要采用黑盒测试与白盒测试结合的测试方式进行系统测试。

## 系统测试

### 黑盒测试

黑盒测试也被称作于功能测试，它是经过测试来校验每个模块的功能都能否正常的运行和使用。测试中，把程序看成无法打开的黑匣子，在不考虑程序的内部结构与特性的情况下，进行程序接口的测试，该测试只检查程序的功能是否按照预期需求那样正常使用，检验接收输入数据是否能产生正确的输出信息。

以“注册”功能为例，输入不符合规则的学号，如果系统未作有效性判断，直接将输入的错误信息录入数据库，将会引发数据各方面的异常。强健的系统通常都需要对数据进行严格的输入验证。如果输入的数据未通过校验，则应该以相应的无效错误信息进行提示。

下表是登录注册模块中针对“注册”功能设计的一个测试用例：

表6.1“注册”测试用例

|  |  |  |
| --- | --- | --- |
| **用例编号** | 4-1 | |
| **用例标题** | 用户注册 | |
| **预置条件** | 注册页面的注册表单完善 | |
| **系统** | 自习帮-自习室预约系统 | |
| **引用模块** | 部门信息管理 | |
| **用例设计者** | 韦治宇 | |
| **设计日期** | 2022-03-02 | |
|  | **操作步骤** | |
| **序号** | **操作步骤/输入数据** | **预期输出** |
| 1 | 学号项留空 | 提示：“请输入您的学号！” |
| 2 | 学号项输入“123” | 提示：“学号至少为6位！” |
| 3 | 学号项输入“abcd123” | 提示：“学号只能为数字！” |
| **判定准则** | 通过输入的学号项，进行表单校验，如果符合表单校验标准，则无红色文字提示表示数据有效，如果有红色文字提示，则表示数据错误且无效。 | |

以下是针对测试用例进行的测试，预期输出与测试用例中的预期测试输出一致，测试结果为通过测试。如下图6.1/6.2/6.3所示。



图6.1注册测试（1）

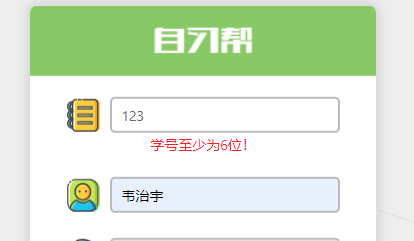


图6.2注册测试（2）



图6.3注册测试（3）

### 白盒测试

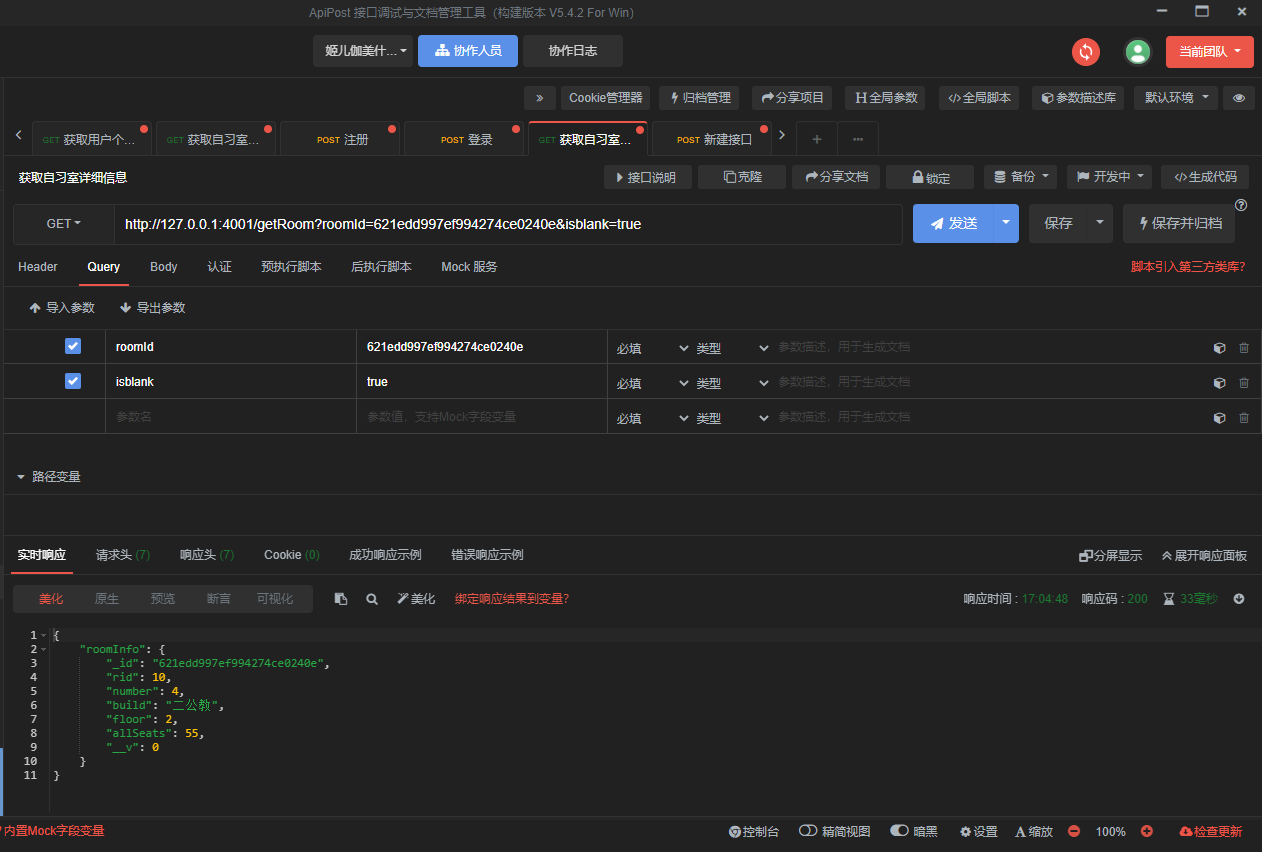
白盒测试是一种软件测试技术，也被称为结构测试，是针对被测单元内部工作的测试。

目前以两种方式为主进行系统测试：

1.前台客户端主要以浏览器的控制台调试运行的测试，经过控制台查看报错信息与断点调试来进行对代码的修改与优化。

2.后台服务端主要以API客户端工具ApiPost对服务端的接口进行路由调试，验证接口是否返回预期数据。

以下为使用API客户端测试工具ApiPost进行的接口测试示例图，通过调用“获取自习室详细信息”接口进行查看接口是否正常返回了预期结果。结果以下图6.4所示。

图6.4 API测试

## 测试总结

本系统开发过程中，每开发完成一项完整功能时，就会进行一次黑盒测试与针对参与功能的API接口的白盒测试，出现问题进行修改后再进行一次测试，直到功能完全不出现问题为止。

从测试结果来看，该系统运行情况正常，基本达成了预期的效果。

# **七、结束语**

自习帮-自习室预约系统的在线预约功能不仅减少了人员聚集，让用户足不出户就可以看到自习室的占用情况，给用户提供了安全舒适的学习环境。自习室的在线预约不仅方便了用户学习与时间分配，更给管理员的管理工作提供了更多便利。

用户自己创建自习室，让志同道合的伙伴们一起加入到自习室的学习中。自习室的对话功能还为加入自习室的小伙伴提供了交流学习的新平台。自习排行榜的设立更激发了用户的学习。

自习室充分利用了现有的自习室资源的使用情况，提升了上座率。更为学校开设教室提供依据，实现能源借阅。

随着时代的更迭，人们的需求也会变得越来越多，如各种人工智能的兴起与用户的学习慢慢关联在一起；在满足基本需求的情况下 需要考虑更多的问题。

由于开发时间有限，业务需求可能有考虑不够周全的地方，系统功能有待在后续的开发工作中予以补充和完善。本系统也会慢慢增加更多的功能，给用户提供便捷优质的体验。

参考文献

[1] 未来科技. HTML5+CSS3+JavaScript从入门到精通[M]. 水利水电出版社. 2017-07.

[2] 明日科技. Javascript开发技术大全[M]. 人民邮电出版社，2008-7.

[3] [美][Douglas Crockford](https://book.douban.com/author/1712618/). JavaScript语言精粹[M]. [电子工业出版社](https://book.douban.com/press/2573)，2009-4.

[4] 段远志. 数据库设计方法[J]. 南昌教育学院学报，2005-12-30.

[5] 田富国. 数据库备份和恢复模块的设计与实现[J]. 江苏建筑职业技术学院学报，2013-03-15.

[6] 计算机软件产品开发文件编制指南[J]. 中华人民共和国国家标准GB8567-88. 国家标准局，1988-01-7.

[7] [美][Kristina Chodorow](http://search.kongfz.com/item_result/?status=0&select=2&author=hk4bk72k69k73k74k69k6ek61k20k43k68k6fk64k6fk72k6fk77" \t "https://book.kongfz.com/261116/4785223456/_blank). [MongoDB权威指南（第2版）](https://item.kongfz.com/book/47471469.html" \t "https://book.kongfz.com/261116/4785223456/_blank)[M]. [人民邮电出版社](https://www.kongfz.com/publisher/3060/" \t "https://book.kongfz.com/261116/4785223456/_blank)，2014-01.

[8] [加][Ilya Grigorik](https://read.douban.com/search?q=[%E5%8A%A0] Ilya Grigorik). [Web性能权威指南](https://item.kongfz.com/book/47471469.html" \t "https://book.kongfz.com/261116/4785223456/_blank)[M]. [人民邮电出版社](https://www.kongfz.com/publisher/3060/" \t "https://book.kongfz.com/261116/4785223456/_blank)，2014-08.

[9] [[美] Ethan Brown](https://read.douban.com/search?q=[%E7%BE%8E]  Ethan Brown). Node与Express开发[M]. [人民邮电出版社](https://www.kongfz.com/publisher/3060/" \t "https://book.kongfz.com/261116/4785223456/_blank)，2015-01.

[10] [杨海霞](https://read.douban.com/search?q=%E6%9D%A8%E6%B5%B7%E9%9C%9E). Node与数据库原理与设计[M]. [人民邮电出版社](https://www.kongfz.com/publisher/3060/" \t "https://book.kongfz.com/261116/4785223456/_blank)，2013-09.

# 致 谢

经过这几个月忙碌的学习，本次的毕业设计也已经完成了。作为从没有经历过撰写论文的一名大学生，由于这方面的经验匮乏，难免有很多考虑不周，思虑不全的地方，要是没有指导老师的指导与督促，想要以预期质量完成这个毕业设计是非常困难的。该论文的完成也是大学生涯的总结，也是今后学习、工作的新起点。我将会以此为基础，进一步提高自己的学术水平和能力。

首先感谢指导老师李老师，在这次的毕业设计中，作为我的指导老师，虽然平日里事务缠身很难有空，但在我毕业设计中的每个阶段，各个环节中都无私敬业的予我悉心的教导。他严谨的作风，渊博的知识，还有平易近人的人格魅力将会对我影响深远。

感谢在大学生涯中陪我一起走过的各位老师和同学们，你们的信任和慰勉是我前进的动力！也要感谢我的朋友们，感谢你们在我失意时给我鼓励，在失落时给我支持，感谢你们和我一路走来，在此过程中我倍感温暖！

最后，对各位专家学者老师们在百忙中抽出时间来评阅本文表示真诚的感谢!