软件工程实验报告

**基于SpringBoot + Vue的酒店管理系统的设计与实现**

学院： 软件学院

专业名称：软件工程（网络空间安全新工科实验班）

学生姓名：陈木华、江洵、李昕泽、杨琪、杨伟康

完成日期： 2024年 11月22日

**摘 要**

本次旨在设计一个酒店管理系统，以满足现代酒店业务的需求。该系统采用前后端分离的架构，使用Spring Boot和Vue框架实现。系统分为管理员页面和普通用户页面两个前端，其中普通用户页面为移动端样式的web项目。

随着旅游业的快速发展，酒店管理系统成为酒店业务运营的重要工具。管理员页面提供了登录功能，以确保只有授权人员可以访问系统。管理员可以通过用户管理功能，对酒店的用户进行管理，包括添加、编辑和删除用户。订单管理功能使管理员能够接受或拒绝订单请求，并对订单进行管理。房间管理功能允许管理员添加、编辑和删除酒店的房间信息。此外，管理员还可以查询和管理用户的评论，以了解用户对酒店的评价和反馈。

普通用户页面提供了登录注册功能，以便用户可以使用系统的各项服务。用户可以通过查询预定房间功能，方便地查找并预订适合自己需求的酒店房间。用户还可以查询个人信息和修改密码，以保持信息的准确性和安全性。评价功能使用户能够对酒店的服务和设施进行评价，为其他用户提供参考。

本文档将详细介绍系统的需求分析、系统架构、数据库设计、前后端接口设计以及各个模块的详细设计。通过本文档的设计，旨在实现一个功能完善、用户友好的酒店管理系统，提供方便快捷的酒店预订和管理服务，提升用户体验和酒店管理效率。

**关键字：酒店管理 Java Vue.js**

**目 录**

[第1章 绪论 4](#_Toc184576323)

[1.1 项目背景 4](#_Toc184576324)

[1.2 任务分解 5](#_Toc184576325)

[1.3 设计报告章节分布 5](#_Toc184576326)

[第2章 系统分析 5](#_Toc184576327)

[2.1 可行性分析 6](#_Toc184576328)

[2.1.1 技术可行性 6](#_Toc184576329)

[2.1.2 运行可行性 6](#_Toc184576330)

[2.2 业务流程分析 6](#_Toc184576331)

[2.2.1 业务流程分析 6](#_Toc184576332)

[2.2.2流程图 8](#_Toc184576333)

[2.2.2.1 数据流程图 8](#_Toc184576334)

[2.2.2.2 系统流程图 9](#_Toc184576335)

[2.3 需求分析 10](#_Toc184576336)

[2.3.1 需求概述 11](#_Toc184576337)

[2.3.1.1 管理员角色需求概述 11](#_Toc184576338)

[2.3.1.2 用户角色需求概述 11](#_Toc184576339)

[2.3.2 用例图 12](#_Toc184576340)

[2.3.3 系统类图 14](#_Toc184576341)

[第3章 概要设计 15](#_Toc184576342)

[3.1 系统设计 15](#_Toc184576343)

[3.1.1 系统架构设计 15](#_Toc184576344)

[3.1.2 系统功能模块图 15](#_Toc184576345)

[3.2 数据库概要设计 15](#_Toc184576346)

[3.2.1 ER图 16](#_Toc184576347)

[第4章 详细设计 16](#_Toc184576348)

[4.1 详细功能模块 16](#_Toc184576349)

[4.1.1 用户登录模块 21](#_Toc184576350)

[4.1.2 预订房间模块 22](#_Toc184576351)

[4.1.3 用户评价模块 23](#_Toc184576352)

[第5章 系统实现 24](#_Toc184576353)

[5.1 前端页面部分代码实现展示 24](#_Toc184576354)

[5.1.1 管理员登录模块 24](#_Toc184576355)

[5.1.2 管理员主页 26](#_Toc184576356)

[5.1.3 用户列表 27](#_Toc184576357)

[5.1.4 预订等待模块 28](#_Toc184576358)

[5.1.5 订单列表模块 30](#_Toc184576359)

[5.1.6 房间列表模块 31](#_Toc184576360)

[5.1.7 评论列表模块 32](#_Toc184576361)

[5.1.8 用户登录模块 33](#_Toc184576362)

[5.1.9 用户注册模块 35](#_Toc184576363)

[5.1.10 查询房间模块 38](#_Toc184576364)

[5.1.11 预订房间模块 41](#_Toc184576365)

[5.1.12 用户主页模块 46](#_Toc184576366)

[第6章 系统测试 49](#_Toc184576367)

[6.1 软硬件环境 49](#_Toc184576368)

[6.2 测试方法 49](#_Toc184576369)

[6.3 测试用例 49](#_Toc184576370)

[6.4 结果及分析 50](#_Toc184576371)

[第7章 总结与展望 54](#_Toc184576372)

[7.1 总结 54](#_Toc184576373)

[8.2 展望 54](#_Toc184576374)

[参考文献 55](#_Toc184576375)

# 第1章 绪论

# 1.1 项目背景

随着全球旅游业的快速发展和人们对旅行需求的增加，酒店业成为一个蓬勃发展的行业。酒店作为旅行的重要组成部分，扮演着提供住宿、餐饮和其他服务的角色。然而，随着酒店业务规模的扩大和竞争的加剧，传统的手工管理方式已经无法满足酒店的日常运营和管理需求。

为了提高酒店的运营效率、提升服务质量和满足客户需求，酒店管理系统应运而生。酒店管理系统是一种基于计算机技术的解决方案，旨在帮助酒店管理人员更好地管理酒店的各个方面，包括客房预订、订单管理、客户关系管理、财务管理等。

传统的酒店管理系统通常是基于桌面应用程序或本地服务器的，存在一些问题，如安装和维护成本高、数据安全性差、用户体验不佳等。随着互联网技术的发展，基于云计算和Web应用的酒店管理系统逐渐兴起。这种新型的酒店管理系统具有许多优势，如灵活性高、可扩展性强、用户体验好等。

本项目旨在设计和开发一种基于Java和Vue.js的酒店管理系统，以满足现代酒店业务的需求。该系统采用前后端分离的架构，使用Spring Boot作为后端框架，Vue.js作为前端框架。通过使用这些先进的互联网技术，我们可以实现一个功能完善、用户友好的酒店管理系统。

该系统将提供管理员页面和普通用户页面两个前端界面。管理员页面将提供登录功能，以确保只有授权人员可以访问系统。管理员可以通过用户管理功能，对酒店的用户进行管理，包括添加、编辑和删除用户。订单管理功能使管理员能够接受或拒绝订单请求，并对订单进行管理。房间管理功能允许管理员添加、编辑和删除酒店的房间信息。此外，管理员还可以查询和管理用户的评论，以了解用户对酒店的评价和反馈。

普通用户页面将提供登录注册功能，以便用户可以使用系统的各项服务。用户可以通过查询预定房间功能，方便地查找并预订适合自己需求的酒店房间。用户还可以修改个人信息和密码，以保持信息的准确性和安全性。评价功能使用户能够对酒店的服务和设施进行评价，为其他用户提供参考。

通过本项目的设计和开发，我们希望能够实现一个高效、安全、易用的酒店管理系统，提供方便快捷的酒店预订和管理服务，提升用户体验和酒店管理效率。同时，我们也希望通过这个项目，推动酒店业务的数字化转型，促进酒店行业的可持续发展。

# 1.2 任务分解

本课题根据前期的问卷调查以及实地考察，作出了适合实际的需求分析，采用了前后端分离的思想，后端使用了SpringBoot和SpringMVC框架，前端使用了Vue.js和ElementUI进行页面的搭建，并且使用了MySQL数据库进行数据交互。整个系统的使用者有两种角色，分别是酒店管理员和用户。

以下是两种不同的角色所分别拥有的功能：

1. 酒店管理员：可进行登录、用户管理、订单管理、房间管理、评论管理。
2. 普通用户：可以注册、登录、查询预定房间、修改个人信息和密码、评价酒店。

# 1.3 设计报告章节分布

本文主要设计和实现一个基于JavaEE的酒店管理系统。

以下罗列出了本论文的章节安排:

第1章 绪论：简明扼要地介绍了本课题的研究背景和意义，阐明了课题的主要任务，最后陈列出了本论文的章节安排。

第2章 系统分析：从经济、技术、运行三个方面介绍了系统可行性，包括业务流程分析和需求分析。

第3章 概要设计：介绍了本系统的设计，包括架构、功能、数据库以及接口的设计。

第4章 详细设计：对系统的主要功能以及数据库的设计进行了详细介绍。

第5章 系统实现：展示了系统的主要功能代码和主要界面截图。

第6章 系统测试：简要介绍了系统的测试方法和测试环境，编写了主要测试用例，并分析记录了测试结果。

第7章 总结与展望：对本次的课题设计进行了总结，展望该系统仍需完善的功能。

# 第2章 系统分析

# 2.1 可行性分析

## 2.1.1 技术可行性

本系统的主要开发语言为Java，基于目前较为流行的SpringBoot以及Vue.js框架进行编写。本系统采用了MySQL数据库作为数据存储基础阵地，保证了一个安全、稳定的数据使用环境。本系统使用的技术，基本上都是目前较为流行的并且是开源的技术，文档和资料在互联网上都非常健全，且有许多开发成功的项目。因此，本系统开发在技术方面是可行的。

## 2.1.2 运行可行性

本系统基于SpringBoot框架，需要搭建Java，Maven， MySQL等软件环境。在搭建的过程中，有遇到过种种的问题，但通过查询资料和请教导师，都能一一解决，并能熟练掌握。因此，本系统开发在运行方面是可行的。

# 2.2 业务流程分析

## 2.2.1 业务流程分析

本系统的使用者一共有酒店管理员和用户这两种角色:

1、酒店管理员功能：

登录：管理员可以通过登录功能进入系统，确保只有授权人员可以访问系统。

用户管理：管理员可以添加、编辑和删除酒店的用户，包括前台员工、服务员等。他们可以分配不同的权限和角色，管理员可以控制他们的访问和操作范围。

订单管理：管理员可以接受或拒绝订单请求，并对订单进行管理。他们可以查看订单详情，修改订单状态（如确认入住、取消预订等），并与客户进行沟通和协调。

房间管理：管理员可以添加、编辑和删除酒店的房间信息。他们可以设置房间的类型、价格、可用性等，并确保房间信息的准确性和更新。

评论管理：管理员可以查询和管理用户的评论。他们可以查看用户对酒店的评价和反馈，可以回复用户的评论，解决问题或提供帮助。

2、普通用户功能：

注册：用户可以通过注册功能创建自己的账户，以便使用系统的各项服务。

登录：用户可以使用注册时创建的账户登录系统，以便访问个人信息和使用其他功能。

查询预定房间：用户可以搜索和查看酒店的房间信息，包括房间类型、价格、可用性等，并进行预订。

查看个人信息和修改密码：用户可以查看自己的个人信息和修改登录密码，以保持信息的安全性。

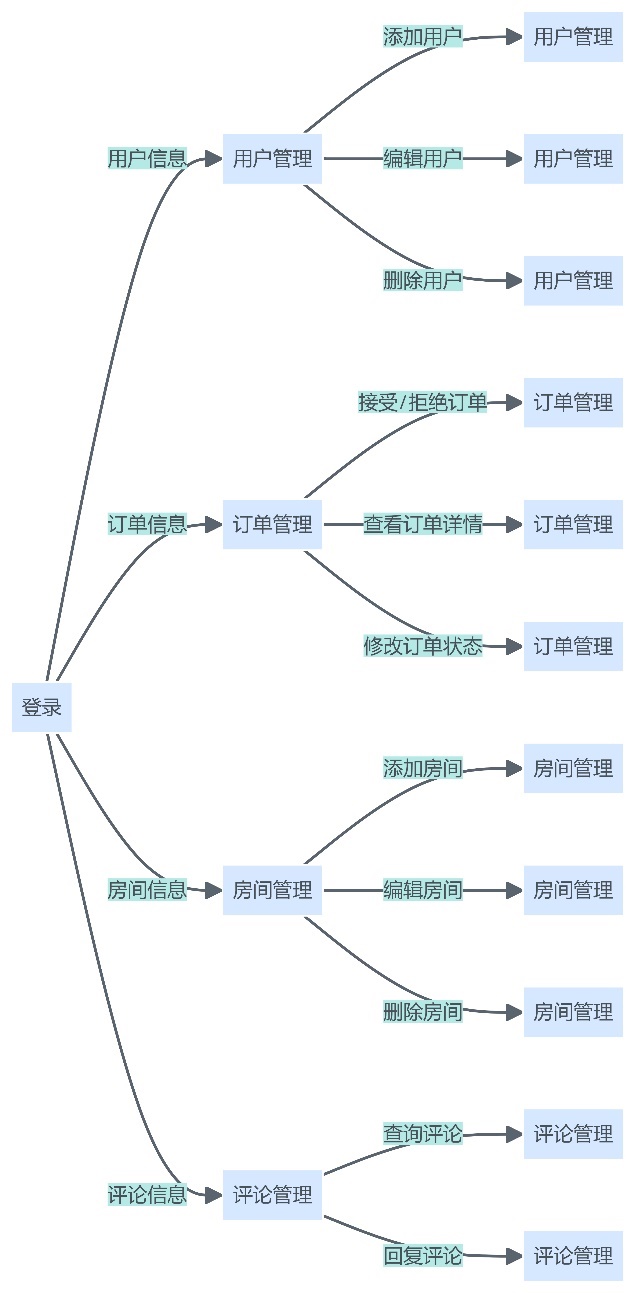
评价酒店：用户可以对酒店进行评价和打分，分享自己的体验和意见，帮助其他用户做出选择。

业务需求表

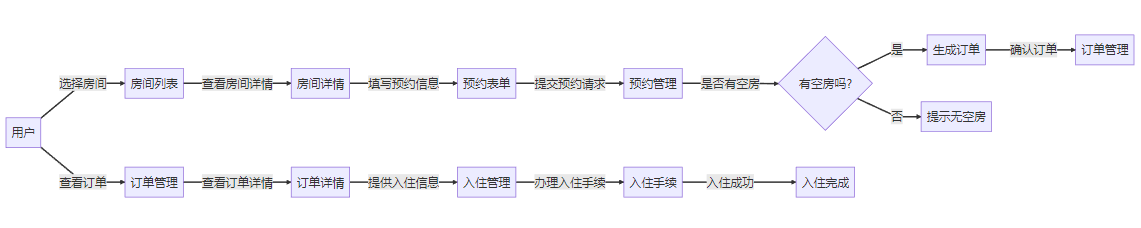
| **编号** | **业务名称** | **角色** | **功能需求** |
| --- | --- | --- | --- |
| 1 | 登录 | 管理员 | 管理员通过用户名和密码登录系统 |
| 2 | 用户管理 | 管理员 | 管理员可以添加、编辑和删除酒店用户，分配权限和角色 |
| 3 | 订单管理 | 管理员 | 管理员可以接受或拒绝订单请求，查看订单详情，修改订单状态 |
| 4 | 房间管理 | 管理员 | 管理员可以添加、编辑和删除酒店房间信息，设置房间类型、价格、可用性 |
| 5 | 评论管理 | 管理员 | 管理员可以查询和管理用户的评论，回复用户的评论 |
| 7 | 注册 | 用户 | 用户可以通过注册功能创建账户 |
| 8 | 登录 | 用户 | 用户可以使用注册时创建的账户登录系统 |
| 9 | 查询预定房间 | 用户 | 用户可以搜索和查看酒店的房间信息，进行预订 |
| 10 | 查询个人信息和密码 | 用户 | 用户可以查询个人信息和修改登录密码 |
| 11 | 评价酒店 | 用户 | 用户可以对酒店进行评价和打分 |

## 2.2.2流程图

### 2.2.2.1 数据流程图

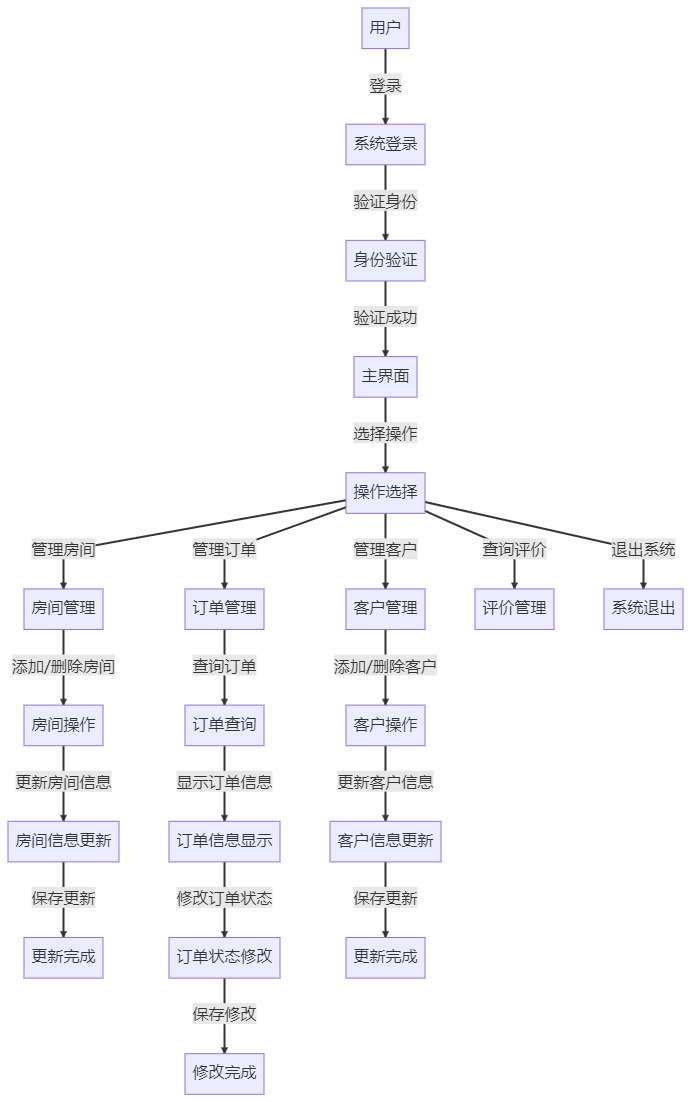


管理员系统管理数据流图



用户预定房间数据流图

### 2.2.2.2 系统流程图



管理员操作系统流程图



用户预定房间流程图

# 2.3 需求分析

## 2.3.1 需求概述

### 2.3.1.1 管理员角色需求概述

管理员是负责管理和运营酒店的人员，他们需要一个酒店管理系统来帮助他们有效地管理酒店的各个方面。根据提供的信息，以下是酒店管理员的需求概述：

1、登录功能：管理员需要能够通过用户名和密码登录到系统，以便访问管理员页面。

2、用户管理：管理员需要能够管理系统中的用户信息，包括添加新用户、编辑用户信息、删除用户等操作。管理员还需要能够查看用户列表和用户详细信息。

3、订单管理：管理员需要能够管理酒店的订单信息。他们需要能够查看订单列表，包括订单的状态（待处理、已接受、已拒绝等）、订单的详细信息（预订人信息、房间信息、预订日期等）。管理员需要能够接受或拒绝订单，并更新订单的状态。

4、房间管理：管理员需要能够管理酒店的房间信息。他们需要能够添加新房间、编辑房间信息、删除房间等操作。管理员还需要能够查看房间列表和房间的详细信息，包括房间类型、房间号码、房间状态等。

5、评论查询功能：管理员需要能够查看用户对酒店的评论信息。他们需要能够查看评论列表，包括评论的内容、评分、评论人信息等。

总体来说，酒店管理员需要一个功能完善的酒店管理系统，以便他们能够方便地管理用户、订单、房间和评论等信息，并进行相应的操作和查询。

### 2.3.1.2 用户角色需求概述

用户角色是酒店管理系统的使用者，他们需要一个方便、易用的移动端web项目来满足他们的需求。根据提供的信息，以下是普通用户的需求概述：

1、登录注册功能：用户需要能够通过用户名和密码进行登录，并且可以注册新的账号。这样他们可以访问个人信息和进行其他操作。

2、查询预定房间：用户需要能够查询酒店的可用房间，并根据自己的需求进行筛选和排序。他们需要能够查看房间的详细信息，包括房间类型、价格、房间设施等，并能够选择预定日期和人数。

3、查询个人信息：用户需要能够查询个人信息，包括姓名、联系方式等。

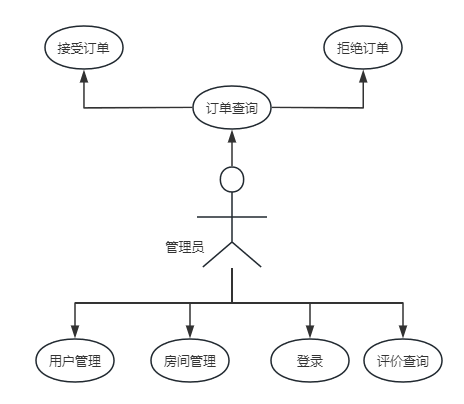
4、修改密码：用户需要能够修改登录密码，以增加账号的安全性。

5、评价功能：用户需要能够对酒店进行评价，并给出相应的评分和评论。这样他们可以分享自己的入住体验，并帮助其他用户做出决策。

总体来说，普通用户需要一个移动端的web项目，以便他们能够方便地查询和预定酒店房间，管理个人信息，并进行评价。用户希望系统界面友好、操作简单，能够提供准确和实时的信息，以满足他们的需求。

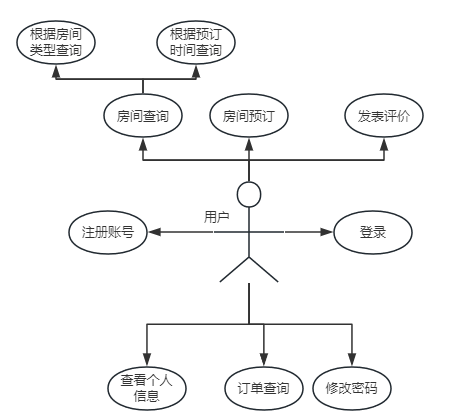
## 2.3.2 用例图

管理员进入网站的页面后，可以进行的功能用例如图所示:



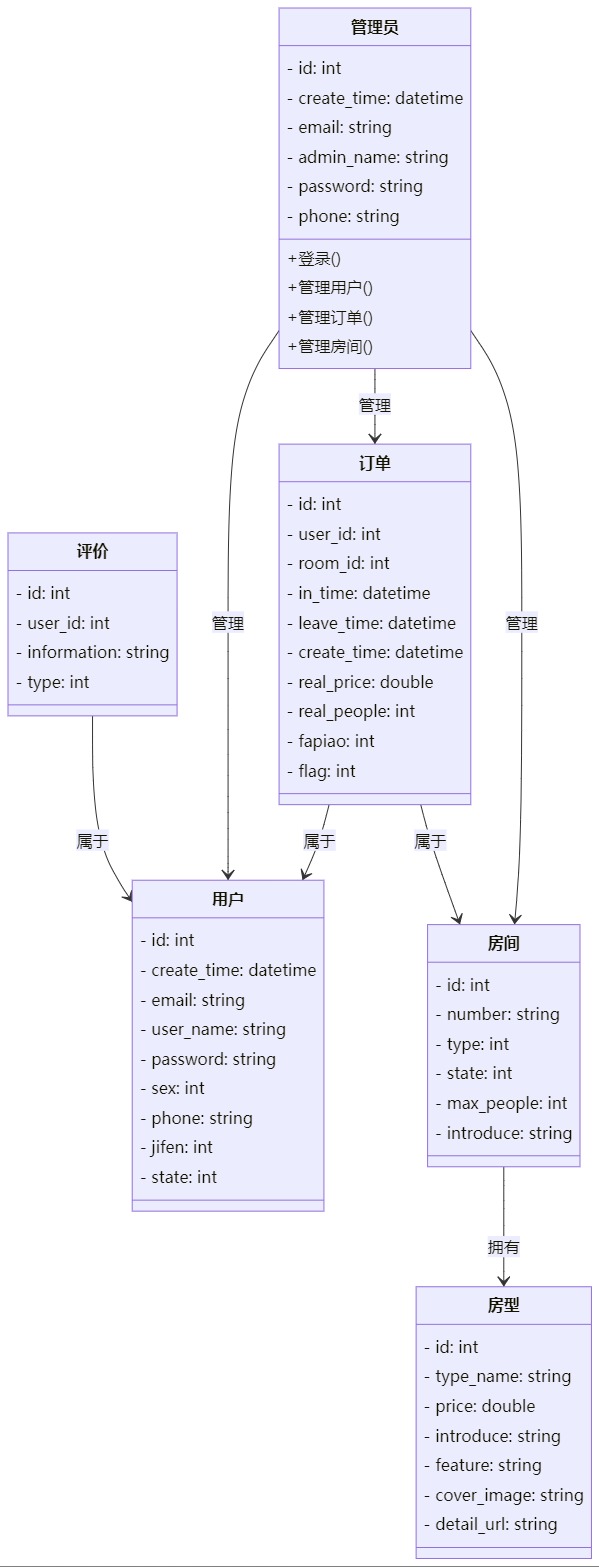
管理员用例图

用户进入网站的页面后，可以进行的功能用例如图所示:



用户用例图

## 2.3.3 系统类图

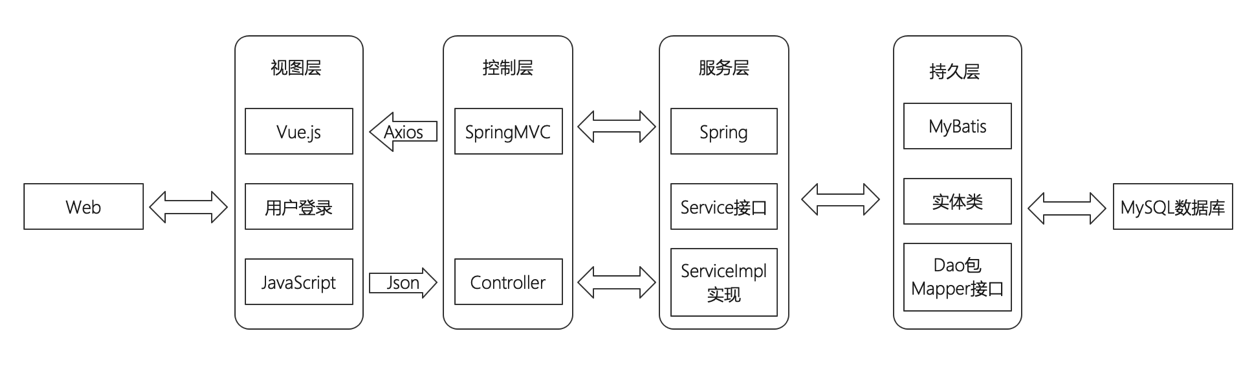


# 第3章 概要设计

# 3.1 系统设计

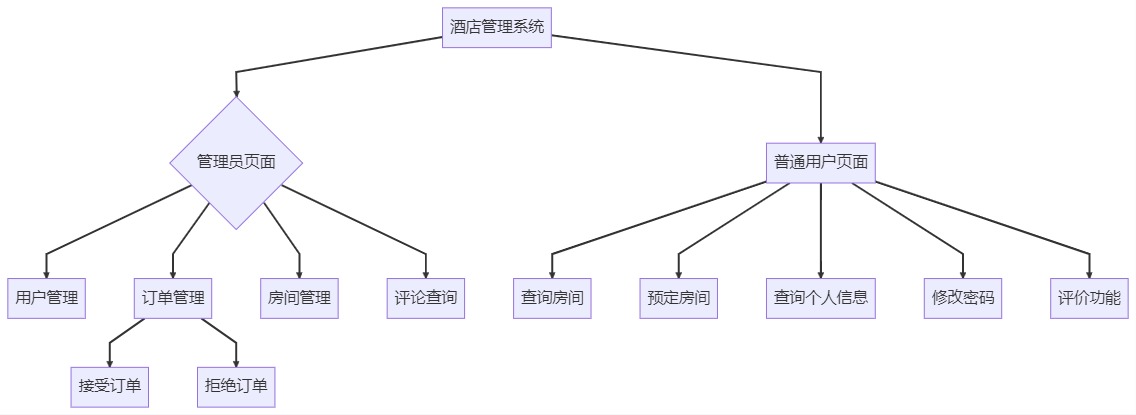
## 3.1.1 系统架构设计

本系统使用SpringBoot、Vue.js等主流开发框架进行开发，遵守MVC模式，同时使用到了Element-UI进行页面的设计。为方便前后端进行整合和日后系统的维护，本系统分为视图层、控制层、业务层和持久层。各层之间相互独立，通过之间的接口进行通信，高内聚，低耦合[[1]](#footnote-1)[6]。



系统架构图

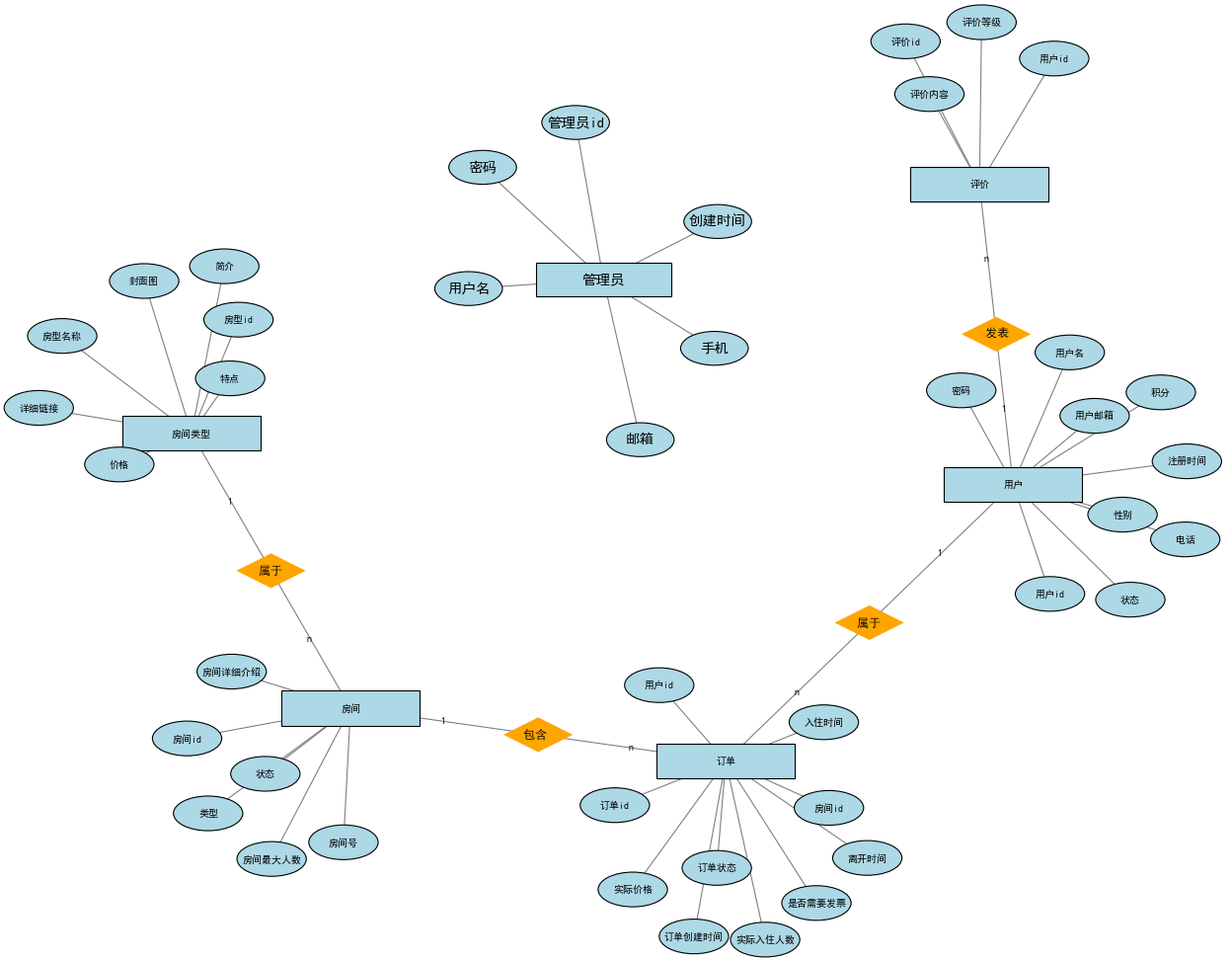
## 3.1.2 系统功能模块图



系统功能模块图

# 3.2 数据库概要设计

## ER图



# 第4章 详细设计

# 4.1 详细功能模块

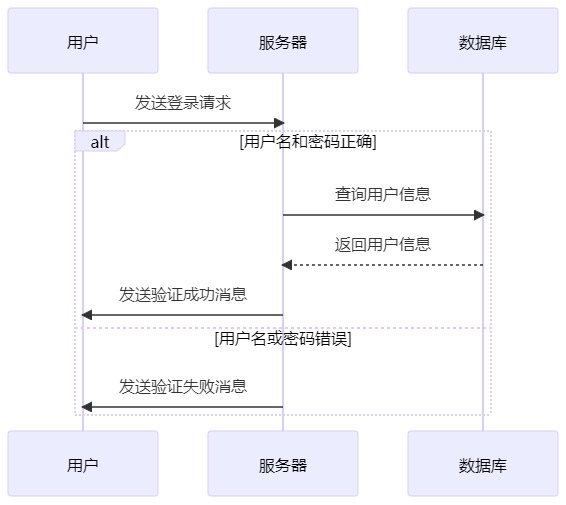
系统功能模块如表所示:

系统功能模块表

| **功能名称** | **输入** | **处理** | **输出** |
| --- | --- | --- | --- |
| 用户注册 | 用户名、密码、确认密码、邮箱（点击“注册”按钮） | 1. 检查用户名是否已存在 2. 检查密码和确认密码是否一致 3. 将用户信息存入数据库 | 1. 如果用户名已存在，弹窗提示“用户名已存在，请重新输入” 2. 如果密码和确认密码不一致，弹窗提示“两次输入的密码不一致，请重新输入” 3. 如果注册成功，弹窗提示“注册成功，请登录” |
| 用户登录 | 用户名、密码（点击“登录”按钮） | 1. 检查用户名是否存在 2. 检查密码是否正确 | 1. 如果用户名不存在，弹窗提示“用户名不存在，请重新输入” 2. 如果密码错误，弹窗提示“密码错误，请重新输入” 3. 如果登录成功，跳转到主页面 |
| 查询订单 | 订单号（点击“查询订单”按钮） | 1. 在数据库中查找与订单号对应的订单信息 | 1. 如果找到订单，弹窗显示订单详细信息 2. 如果未找到订单，弹窗提示“未找到该订单，请检查订单号是否正确” |
| 修改个人信息 | 新的用户名、密码、确认密码、邮箱（点击“修改个人信息”按钮） | 1. 检查新的用户名是否已存在 2. 检查新密码和确认密码是否一致 3. 更新数据库中的用户信息 | 1. 如果新的用户名已存在，弹窗提示“用户名已存在，请重新输入” 2. 如果新密码和确认密码不一致，弹窗提示“两次输入的密码不一致，请重新输入” 3. 如果修改成功，弹窗提示“个人信息修改成功” |
| 提交评价 | 评价内容（点击“提交评价”按钮） | 1. 将评价内容存入数据库 | 1. 如果评价提交成功，弹窗提示“评价提交成功” |
| 查看房间信息 | 房间号（点击“查看房间信息”按钮） | 在数据库中查找与房间号对应的房间信息 | 如果找到房间信息，弹窗显示房间详细信息；如果未找到房间信息，弹窗提示“未找到该房间，请检查房间号是否正确” |
| 用户管理 | 用户名（点击“查看用户”按钮） | 在数据库中查找与用户名对应的用户信息 | 如果找到用户信息，弹窗显示用户详细信息；如果未找到用户信息，弹窗提示“未找到该用户，请检查用户名是否正确” |
| 删除用户 | 用户名（点击“删除用户”按钮） | 在数据库中删除与用户名对应的用户信息 | 如果删除成功，弹窗提示“用户删除成功”；如果未找到该用户，弹窗提示“未找到该用户，请检查用户名是否正确” |
| 修改用户信息 | 新的用户名、密码、确认密码、邮箱（点击“修改用户信息”按钮） | 1. 检查新的用户名是否已存在2. 更新数据库中的用户信息 | 1. 如果新的用户名已存在，弹窗提示“用户名已存在，请重新输入” 2. 如果修改成功，弹窗提示“用户信息修改成功” |
| 查看房间管理 | 无（管理员默认可见） | / | 显示房间列表，包括房间号、房间类型、房间状态等信息 |
| 添加房间 | 房间号、房间类型、房间状态（点击“添加房间”按钮） | 将房间信息存入数据库 | 如果添加成功，弹窗提示“房间添加成功”；如果房间号已存在，弹窗提示“房间号已存在，请重新输入” |
| 删除房间 | 房间号（点击“删除房间”按钮） | 在数据库中删除与房间号对应的房间信息 | 如果删除成功，弹窗提示“房间删除成功”；如果未找到该房间，弹窗提示“未找到该房间，请检查房间号是否正确” |
| 修改房间信息 | 新的房间号、房间类型、房间状态（点击“修改房间信息”按钮） | 1. 检查新的房间号是否已存在 2. 更新数据库中的房间信息 | 1. 如果新的房间号已存在，弹窗提示“房间号已存在，请重新输入” 2. 如果修改成功，弹窗提示“房间信息修改成功” |
| 接受订单 | 订单号（管理员在订单列表中选择需要接受的订单并点击“接受订单”按钮） | 根据订单号在数据库中找到对应的订单信息，将其状态设置为已接受 | 如果接受成功，弹窗提示“订单接受成功”；如果未找到该订单，弹窗提示“未找到该订单，请检查订单号是否正确” |
| 拒绝订单 | 订单号（管理员在订单列表中选择需要拒绝的订单并点击“拒绝订单”按钮） | 根据订单号在数据库中找到对应的订单信息，将其状态设置为已拒绝，并发送通知给相关方（如酒店、客户等） | 如果拒绝成功，弹窗提示“订单拒绝成功”；如果未找到该订单，弹窗提示“未找到该订单，请检查订单号是否正确”；如果发送通知失败，弹窗提示“通知发送失败，请稍后重试” |
| 订单管理 | 无（管理员默认可见） | / | 显示订单列表，包括订单号、房间号、入住时间、退房时间等信息；提供搜索功能 |
| 评价查询 | 无（管理员默认可见） | / | 显示所有评价列表，包括评价内容、评价人、评价时间等信息 |

## 4.1.1 用户登录模块

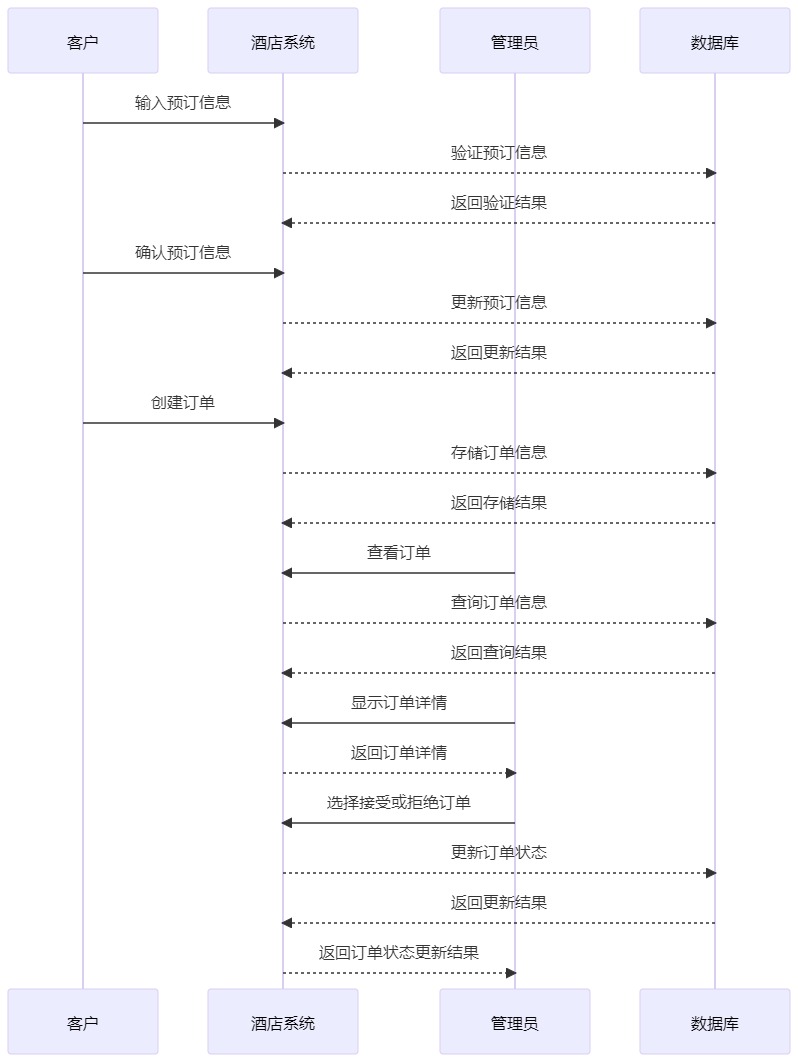
用户输入帐号和密码后，传至系统后台进行校验。如果帐号密码不匹配，提示错误。账号密码都匹配的话，跳转主页，如图所示：



登录时序图

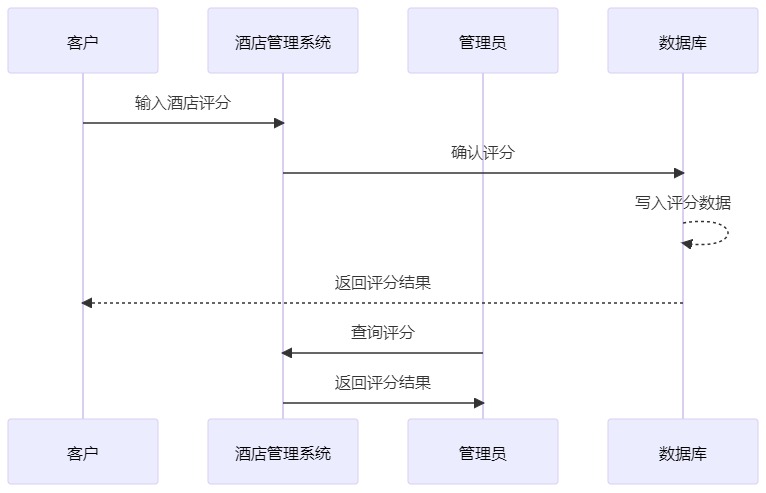
## 4.1.2 预订房间模块

客户首先向酒店系统输入预订信息，酒店系统验证这些信息的准确性。如果信息无误，客户确认预订并创建订单。然后，管理员可以查看酒店系统中的订单，并选择接受或拒绝该订单。最后，酒店系统根据管理员的选择更新订单状态。如图所示：



客户预订房间时序图

## 4.1.3 用户评价模块

客户输入酒店评分，酒店管理系统确认评分并将其发送给数据库写入数据，数据库返回评分结果给酒店管理系统，再由酒店管理系统返回给客户。管理员可以查询酒店评分，酒店管理系统将查询请求发送给数据库，数据库返回评分结果给管理员。如图所示：

用户评价时序图

# 第5章 系统实现

# 5.1 前端页面部分代码实现展示

## 5.1.1 管理员登录模块

    <div class="login-container">

        <el-card class="box-card">

            <div class="text item">

                <el-form

                    ref="loginForm"

                    :model="loginForm"

                    class="login-form"

                    auto-complete="on"

                    label-position="left"

                >

                    <div class="title-container">

                        <h2 class="title">酒店管理系统</h2>

                    </div>

                    <el-form-item prop="username">

                        <!-- <span class="svg-container">

                            <svg-icon icon-class="user" />

                        </span> -->

                        <i

                            class="iconfont icon-r-user1"

                            style="font-size: 26px; margin-left: 10px"

                        ></i>

                        <el-input

                            id="account"

                            ref="username"

                            v-model="loginForm.username"

                            name="username"

                            type="text"

                            tabindex="1"

                        />

                    </el-form-item>

                    <el-form-item prop="password">

                        <!-- <span class="svg-container">

                            <svg-icon icon-class="password" />

                        </span> -->

                        <i

                            class="iconfont icon-r-lock"

                            style="font-size: 26px; margin: 0 0 -5px 10px"

                        ></i>

                        <el-input

                            id="psw"

                            :key="passwordType"

                            ref="password"

                            v-model="loginForm.password"

                            :type="passwordType"

                            name="password"

                            tabindex="2"

                            @keyup.enter.native="handleLogin"

                        />

                        <span class="show-pwd" @click="showPwd">

                            <svg-icon

                                :icon-class="

                                    passwordType === 'password'

                                        ? 'eye'

                                        : 'eye-open'

                                "

                            />

                        </span>

                    </el-form-item>

                    <el-button

                        id="login\_btn"

                        type="primary"

                        style="

                            width: 100%;

                            margin-bottom: 30px;

                            font-size: 22px;

                        "

                        @click.native.prevent="handleLogin"

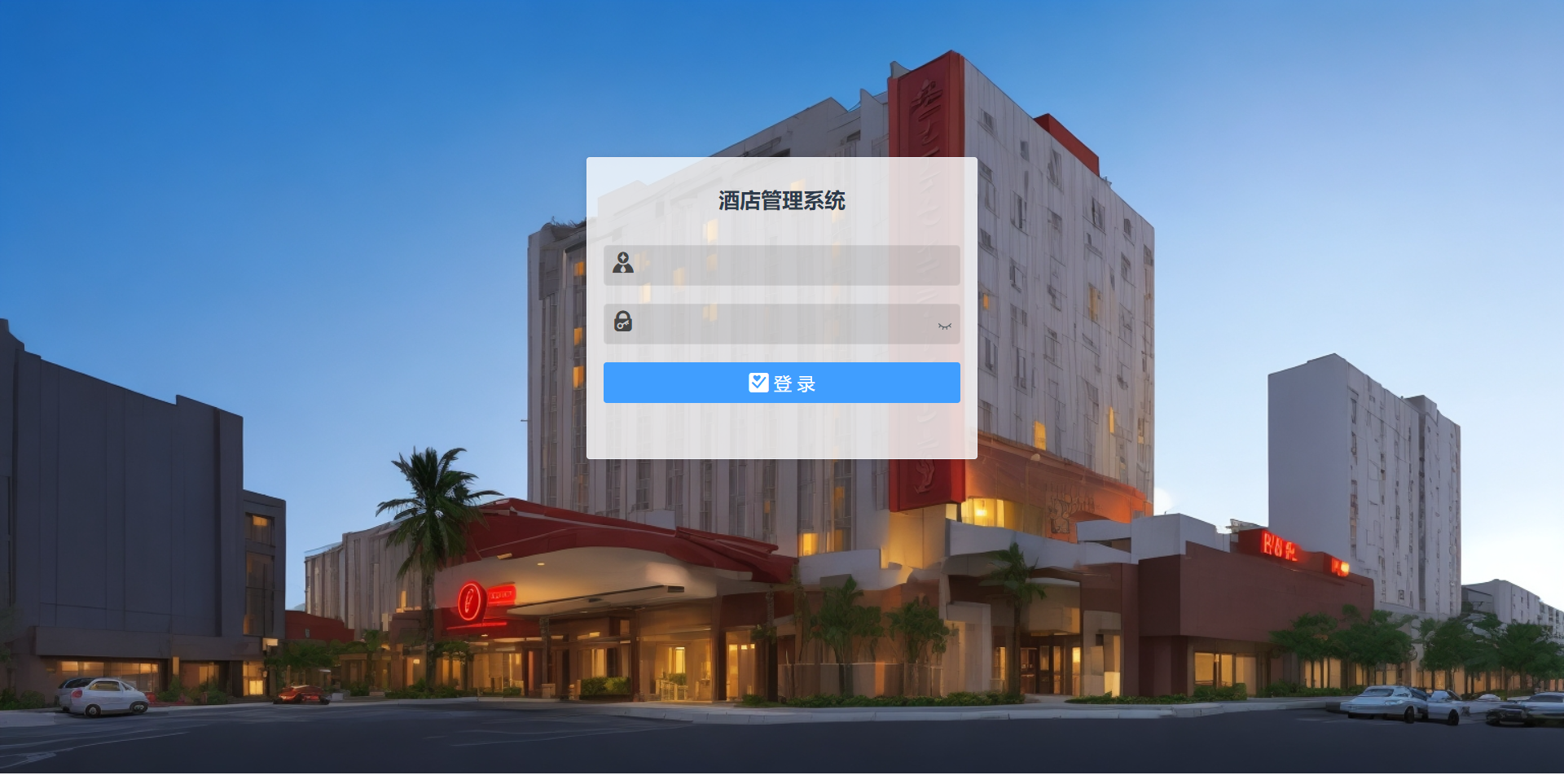
                    >

                        登 录</el-button

                    >

                </el-form>

            </div>



管理员登录页面

## 5.1.2 管理员主页

<template>

  <el-row>

    <el-col :span="24" class="text-center">

      <p class="title1">

        欢迎使用酒店管理系统

      </p>

    </el-col>

    <el-col :span="24" class="text-center" id="homeCol">

      <img :src="require('@/assets/home.jpg')" style="width: 80%;">

    </el-col>

  </el-row>

</template>



管理员主界面

## 5.1.3 用户列表

<template>

    <div class="user-table">

        <el-table :data="tableData" stripe style="width: 100%">

            <el-table-column prop="id" label="id" class="text-center">

            </el-table-column>

            <el-table-column prop="createTime" label="创建时间">

            </el-table-column>

            <el-table-column prop="email" label="邮箱"> </el-table-column>

            <el-table-column prop="userName" label="用户名"> </el-table-column>

            <el-table-column prop="sex" label="性别"> </el-table-column>

            <el-table-column prop="phone" label="手机号码"> </el-table-column>

            <el-table-column prop="jifen" label="账户积分"> </el-table-column>

            <el-table-column prop="state" label="状态"> </el-table-column>

            <el-table-column label="操作" width="140" fixed="right">

                <template slot-scope="scope">

                    <el-button

                        type="danger"

                        @click="deleteUser(scope.row.id)"

                        style="font-size: 18px;"

                        >

                         删除</el-button

                    >

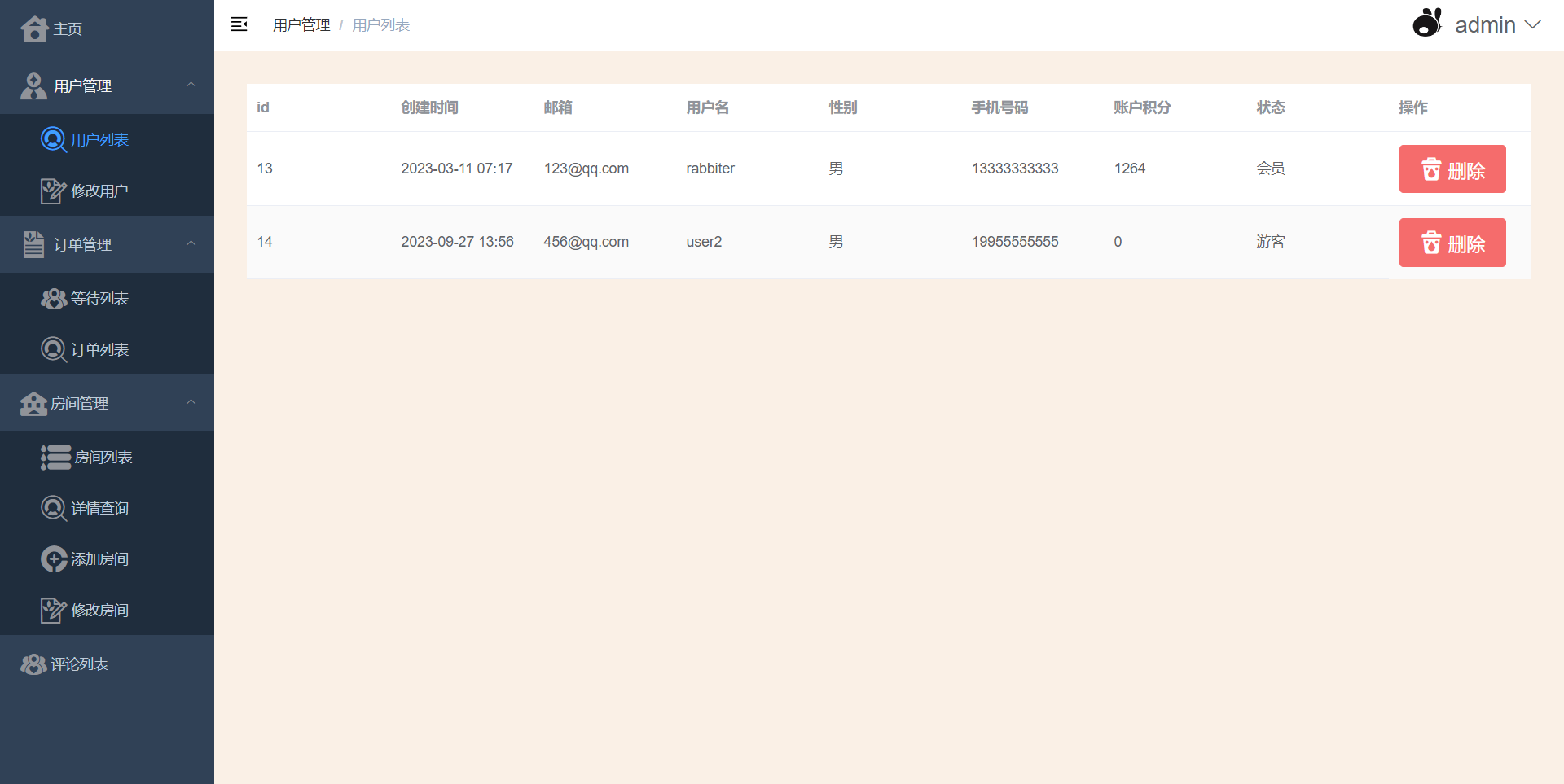
                </template>

            </el-table-column>

        </el-table>

    </div>

</template>



用户管理界面

## 5.1.4 预订等待模块

<template>

    <div class="user-table">

        <el-table :data="tableData" stripe style="width: 100%">

            <el-table-column prop="id" label="id"> </el-table-column>

            <el-table-column prop="userId" label="用户id"> </el-table-column>

            <el-table-column prop="roomId" label="房间id"> </el-table-column>

            <el-table-column prop="createTime" label="下单时间">

            </el-table-column>

            <el-table-column prop="inTime" label="入住时间"> </el-table-column>

            <el-table-column prop="leaveTime" label="离店时间">

            </el-table-column>

            <el-table-column prop="realPrice" label="总计价格">

            </el-table-column>

            <el-table-column prop="realPeople" label="入住人数">

            </el-table-column>

            <el-table-column label="发票">

                <template slot-scope="scope">

                    <span v-if="scope.row.fapiao == 1">需要</span>

                    <span v-if="scope.row.fapiao == 0">不需要</span>

                </template>

            </el-table-column>

            <el-table-column prop="flag" label="受理状态"> </el-table-column>

            <el-table-column label="操作" width="250" fixed="right">

                <template slot-scope="scope">

                    <el-button

                        type="success"

                        @click="accept(scope.row.id)"

                        style="font-size: 18px;"

                        >

                         接受</el-button

                    >

                    <el-button

                        type="danger"

                        @click="cancel(scope.row.id)"

                        style="font-size: 18px;"

                        >

                         取消</el-button

                    >

                </template>

            </el-table-column>

        </el-table>

    </div>

</template>



预订等待界面

## 5.1.5 订单列表模块

<template>

    <div class="user-table">

        <el-table :data="tableData" stripe style="width: 100%">

            <el-table-column prop="id" label="id"> </el-table-column>

            <el-table-column prop="userId" label="用户id"> </el-table-column>

            <el-table-column prop="roomId" label="房间id"> </el-table-column>

            <el-table-column prop="createTime" label="下单时间">

            </el-table-column>

            <el-table-column prop="inTime" label="入住时间"> </el-table-column>

            <el-table-column prop="leaveTime" label="离店时间">

            </el-table-column>

            <el-table-column prop="realPrice" label="总计价格">

            </el-table-column>

            <el-table-column prop="realPeople" label="入住人数">

            </el-table-column>

            <el-table-column label="发票">

                <template slot-scope="scope">

                    <span v-if="scope.row.fapiao == 1">需要</span>

                    <span v-if="scope.row.fapiao == 0">不需要</span>

                </template>

            </el-table-column>

            <el-table-column prop="flag" label="受理状态"> </el-table-column>

        </el-table>

    </div>

</template>



订单列表界面

## 5.1.6 房间列表模块

<template>

    <div class="user-table">

        <el-table :data="tableData" stripe style="width: 100%">

            <el-table-column prop="id" label="房间id"> </el-table-column>

            <el-table-column prop="number" label="房间号"> </el-table-column>

            <el-table-column prop="type" label="房间类型"> </el-table-column>

            <el-table-column prop="state" label="房间状态"> </el-table-column>

            <el-table-column prop="maxPeople" label="最大容纳人数">

            </el-table-column>

            <el-table-column prop="introduce" label="客房简介">

            </el-table-column>

            <el-table-column label="操作" width="140" fixed="right" >

                <template slot-scope="scope">

                    <el-button

                        type="danger"

                        @click="deleteRoom(scope.row.id)"

                        style="font-size: 18px;"

                        >

                         删除</el-button

                    >

                </template>

            </el-table-column>

        </el-table>

    </div>

</template>



房间列表界面

## 5.1.7 评论列表模块

<template>

  <div class="mb-1">

    <el-row class="mt-1" v-for="(comm, index) in comment" :key="index">

      <el-col :span="12" :offset="6">

        <el-card class="box-card">

          <div slot="header" class="clearfix">

            <span>#{{comm.comment.id}} From {{comm.user.userName}} (userid: {{comm.user.id}})</span>

            <span>Phone: {{comm.user.phone}} Email: {{comm.user.email}}</span>

          </div>

          <div class="text item">

            <p>留言内容： {{comm.comment.information}}</p>

            <p>评分等级： {{comm.comment.type}}星</p>

          </div>

        </el-card>

      </el-col>

    </el-row>

  </div>

</template>



评论列表界面

## 5.1.8 用户登录模块

<template>

    <div>

        <el-container class="wrapper">

            <el-main class="section text-center">

                <div class="login-bg"></div>

                <el-card class="box-card login-card">

                    <div class="text item">

                        <p class="login-title">

                            <i

                                class="iconfont icon-r-yes"

                                style="font-size: 32px"

                            ></i>

                            酒店预订系统

                        </p>

                        <el-form

                            :model="login"

                            status-icon

                            ref="login"

                            label-width="80px"

                        >

                            <el-form-item label="账户邮箱" prop="email">

                                <el-input

                                    type="text"

                                    v-model="login.email"

                                    autocomplete="off"

                                ></el-input>

                            </el-form-item>

                            <el-form-item label="密码" prop="pass">

                                <el-input

                                    type="password"

                                    v-model="login.password"

                                    autocomplete="off"

                                ></el-input>

                            </el-form-item>

                        </el-form>

                    </div>

                </el-card>

                <el-button

                    circle

                    :type="btnType"

                    @click="loginBtn"

                    class="loginbtn"

                    :disabled="disabled"

                >

                    <i :class="iconstyle" style="font-size: 54px"></i>

                </el-button>

                <div class="register">

                    <p>

                        <router-link to="/register" style="color: black"

                            >注册账号</router-link

                        >

                    </p>

                </div>

            </el-main>

        </el-container>

        <el-footer class="footer text-center">

            <copyright></copyright>

        </el-footer>

    </div>

</template>



用户登录界面

## 5.1.9 用户注册模块

<template>

    <div>

        <el-container class="wrapper">

            <el-main class="section text-center">

                <div class="register-bg"></div>

                <el-card class="box-card register-card">

                    <div class="text item">

                        <p class="register-title">账号注册</p>

                        <el-form

                            :model="register"

                            status-icon

                            :rules="rules"

                            ref="register"

                            label-width="80px"

                        >

                            <el-form-item label="用户名" prop="username">

                                <el-input

                                    type="text"

                                    v-model="register.username"

                                    suffix-icon="el-icon-user"

                                ></el-input>

                            </el-form-item>

                            <el-form-item label="密码" prop="password">

                                <el-input

                                    type="password"

                                    v-model="register.password"

                                    suffix-icon="el-icon-lock"

                                ></el-input>

                            </el-form-item>

                            <el-form-item label="确认密码" prop="confirm">

                                <el-input

                                    type="password"

                                    v-model="register.confirm"

                                    suffix-icon="el-icon-lock"

                                ></el-input>

                            </el-form-item>

                            <el-form-item label="手机号码" prop="phone">

                                <el-input

                                    type="text"

                                    v-model="register.phone"

                                    suffix-icon="el-icon-mobile-phone"

                                ></el-input>

                            </el-form-item>

                            <el-form-item label="邮箱" prop="email">

                                <el-input

                                    type="email"

                                    v-model="register.email"

                                    suffix-icon="el-icon-message"

                                ></el-input>

                            </el-form-item>

                            <el-form-item label="性别" prop="gender">

                                <el-radio v-model="register.gender" label="1"

                                    >男</el-radio

                                >

                                <el-radio v-model="register.gender" label="0"

                                    >女</el-radio

                                >

                            </el-form-item>

                        </el-form>

                        <el-button

                            :type="btnType"

                            @click="registerBtn('register')"

                            class="registerbtn"

                            :disabled="disabled"

                        >

                             注册账号

                        </el-button>

                    </div>

                </el-card>

                <div class="register">

                    <p>

                        <router-link to="/login" style="color: black"

                            >已有账号？立即登录</router-link

                        >

                    </p>

                </div>

            </el-main>

        </el-container>

        <el-footer class="footer text-center">

            <copyright></copyright>

        </el-footer>

    </div>

</template>



用户注册界面

## 5.1.10 查询房间模块

<template>

    <div>

        <el-container class="wrapper">

            <!-- <el-header class="header">找房</el-header> -->

            <el-main class="section">

                <div class="bg-header"></div>

                <el-card class="box-card pb-1" style="opacity: 0.95;">

                    <div class="text item" style="margin-bottom: 20px;">

                        <el-row>

                            <el-col :span="16">

                                <p class="text-left"><i class="el-icon-location-outline"></i>北京大学城星级宾馆</p>

                            </el-col>

                        </el-row>

                        <el-form ref="form" :model="form">

                            <el-row :gutter="20">

                                <el-col :span="12">

                                    <el-form-item label="入住时间">

                                        <el-date-picker type="datetime" placeholder="选择日期" v-model="form.indate" style="width: 100%;"></el-date-picker>

                                    </el-form-item>

                                </el-col>

                                <el-col :span="12">

                                    <el-form-item label="离店时间">

                                        <el-date-picker type="datetime" placeholder="选择日期" v-model="form.leavedate" style="width: 100%;"></el-date-picker>

                                    </el-form-item>

                                </el-col>

                            </el-row>

                        </el-form>

                    </div>

                </el-card>

                <el-row class="mt-1">

                    <el-card class="mb-1 ml-1 mr-1">

                        <div class="text item">

                            <el-button type="primary"  style="font-size: 16px;" class="btn-block" @click="searchBtn"> 查询空余房间</el-button>

                        </div>

                    </el-card>

                    <el-col :span="22" v-for="(tp, index) in listdata" :key="index" :offset="1" class="mb-1">

                        <el-card :body-style="{ padding: '0px' }">

                            <el-row>

                                <el-col :span="6">

                                    <img :src="tp.coverImage" class="image">

                                </el-col>

                                <el-col :span="18">

                                    <div style="padding: 14px;" class="text-left room">

                                        <!-- :href="tp.detailUrl"  -->

                                        <p class="room-title">

                                            {{tp.typeName}}

                                        </p>

                                        <p class="room-intro">{{tp.introduce}}</p>

                                        <p class="room-price">￥{{tp.price}} 起</p>

                                        <p class="room-comment">“{{tp.feature}}”</p>

                                    </div>

                                </el-col>

                            </el-row>

                        </el-card>

                    </el-col>

                </el-row>

            </el-main>

            <el-footer class="footer">

                <footbar></footbar>

            </el-footer>

            <el-dialog title="提示" :visible.sync="dialogVisible" width="95%">

                <span>您还没有选择完整的时间段！</span>

                <span slot="footer" class="dialog-footer">

                    <el-button type="primary" @click="dialogVisible = false"> 确定</el-button>

                </span>

            </el-dialog>

        </el-container>

    </div>

</template>



查询房间界面

## 5.1.11 预订房间模块

<template>

    <div>

        <el-container class="wrapper">

            <el-main class="section text-left">

                <div class="res-bg"></div>

                <el-card class="box-card res-form">

                    <div slot="header" class="clearfix">

                        <span>预订客房</span>

                    </div>

                    <div class="text item">

                        <el-form

                            :model="resinfo"

                            :rules="rules"

                            ref="ruleForm"

                            label-width="80px"

                            class="demo-ruleForm"

                        >

                            <el-form-item

                                label="客房类型"

                                prop="type"

                                required

                                @click="resolveData"

                            >

                                <el-select

                                    v-model="resinfo.type"

                                    placeholder="请选择"

                                    :disabled="lockCondition"

                                >

                                    <el-option

                                        v-for="item in roomtype"

                                        :key="item.id"

                                        :label="item.typeName"

                                        :value="item.id"

                                    >

                                    </el-option>

                                </el-select>

                            </el-form-item>

                            <el-form-item label="入住时间" required>

                                <el-col :span="24" class="mb-1">

                                    <el-form-item prop="indate">

                                        <el-date-picker

                                            :disabled="lockCondition"

                                            type="datetime"

                                            placeholder="入住日期"

                                            v-model="resinfo.indate"

                                            style="width: 100%"

                                        >

                                        </el-date-picker>

                                    </el-form-item>

                                </el-col>

                                <!-- <el-col class="line text-center" :span="24">至</el-col> -->

                                <el-col :span="24">

                                    <el-form-item prop="leavedate">

                                        <el-date-picker

                                            :disabled="lockCondition"

                                            type="datetime"

                                            placeholder="离店日期"

                                            v-model="resinfo.leavedate"

                                            style="width: 100%"

                                        >

                                        </el-date-picker>

                                    </el-form-item>

                                </el-col>

                            </el-form-item>

                            <el-form-item>

                                <el-button

                                    :type="searchBtnType"

                                    @click="checkroom"

                                    :disabled="checkBtnDis"

                                >

                                    <i

                                        :class="searchicon"

                                        style="font-size: 18px"

                                    ></i>

                                     查询可用房间

                                </el-button>

                            </el-form-item>

                            <!-- 提供一个房间号码对房间id的映射，即用户选择的是房间号，但存的是房间id -->

                            <el-form-item label="房间号码" prop="id" required>

                                <el-select

                                    v-model="resinfo.id"

                                    placeholder="请先选择房间号"

                                    :disabled="isNotSelect"

                                >

                                    <el-option

                                        v-for="item in roomnumber"

                                        :key="item.id"

                                        :label="item.number"

                                        :value="item.id"

                                    >

                                    </el-option>

                                </el-select>

                            </el-form-item>

                            <el-form-item label="住店人数" prop="num" required>

                                <el-input-number

                                    v-model="resinfo.num"

                                    :min="1"

                                    :max="maxp"

                                    label="请输入"

                                    :disabled="isNotSelect"

                                ></el-input-number>

                                <span class="ml-1">人</span>

                            </el-form-item>

                            <el-form-item label="提供发票" prop="invoice">

                                <el-switch

                                    v-model="resinfo.isNeedInvoice"

                                    :disabled="isNotSelect"

                                ></el-switch>

                            </el-form-item>

                            <el-form-item label="价格">

                                <span v-model="resinfo.type" class="room-price"

                                    >{{ sumprice }} 元</span

                                >

                            </el-form-item>

                            <el-form-item>

                                <el-button

                                    type="primary"

                                    @click="submitForm('ruleForm')"

                                    style="font-size: 22px"

                                    :disabled="isNotSelect"

                                >

                                    立即预订

                                </el-button>

                                <br />

                                <el-button

                                    @click="resetForm('ruleForm')"

                                    type="warning"

                                    style="font-size: 22px"

                                >

                                    重置选项

                                </el-button>

                            </el-form-item>

                        </el-form>

                    </div>

                </el-card>

            </el-main>

            <el-footer class="footer">

                <footbar></footbar>

            </el-footer>

            <el-dialog title="提示" :visible.sync="dialogVisible" width="95%">

                <span>你确定要提交预订单吗？</span>

                <span slot="footer" class="dialog-footer">

                    <el-button @click="dialogVisible = false" style="font-size: 18px;"> 取消</el-button>

                    <el-button type="primary" @click="confirmSubmitForm"

                        >确定</el-button

                    >

                </span>

            </el-dialog>

        </el-container>

    </div>

</template>



预订房间界面

## 5.1.12 用户主页模块

<template>

  <div>

    <el-container class="wrapper">

      <el-main class="section text-left">

        <div class="mine-bg"></div>

        <el-card class="box-card mine-card">

          <div class="text item">

            <el-row>

              <el-col :span="6" class="text-right">

                <img v-if="userdata.sex == 1" src="../assets/img/man.png" class="avatar" />

                <img v-else src="../assets/img/woman.png" class="avatar" />

              </el-col>

              <el-col :span="18">

                <p class="username">

                  {{userdata.userName}}

                  <i v-if="userdata.sex == 1" class="el-icon-male man"></i>

                  <i v-else class="el-icon-female woman"></i>

                </p>

                <p class="userid">id：{{userdata.id}}</p>

              </el-col>

            </el-row>

          </div>

        </el-card>

        <el-card class="box-card mine-card-tool">

          <div class="text item">

            <el-row>

              <el-col :span="6" class="text-center">

                <router-link to="/history" class="text-plain order">

                  <p class="tool-text">

                    我的订单

                  </p>

                </router-link>

              </el-col>

              <el-col :span="6" class="text-center">

                <router-link to="/checkinfo" class="text-plain info">

                  <p class="tool-text">

                    我的信息

                  </p>

                </router-link>

              </el-col>

              <el-col :span="6" class="text-center">

                <router-link to="/changepswd" class="text-plain password">

                  <p class="tool-text">

                    密码设置

                  </p>

                </router-link>

              </el-col>

              <el-col :span="6" class="text-center">

                <router-link to="/contact" class="text-plain contact">

                  <p class="tool-text">

                    联系我们

                  </p>

                </router-link>

              </el-col>

            </el-row>

          </div>

        </el-card>

        <div class="btn-logout">

          <el-button type="danger" class="btn-block" style="font-size: 18px;" @click="dialogVisible = true">

             退出登录

          </el-button>

        </div>

      </el-main>

      <el-footer class="footer">

        <footbar></footbar>

      </el-footer>

    </el-container>

    <el-dialog title="退出" :visible.sync="dialogVisible" width="95%">

      <span>你确定要退出登录吗？</span>

      <span slot="footer" class="dialog-footer">

        <el-button type="primary" @click="dialogVisible = false" style="font-size: 18px;"> 取消</el-button>

        <el-button @click="logoutBtn"> 确定</el-button>

      </span>

    </el-dialog>

  </div>

</template>



用户主页界面

# 第6章 系统测试

# 6.1 软硬件环境

操作系统：Windows 10

数据库：MySQL 8.0.19

应用软件：Idea 2021.03

浏览器：Chrome 90.0.4430.93

软硬件环境：5台电脑

# 6.2 测试方法

系统测试是指在完成项目编码后，系统正式上线前，根据系统前期的需求分析以及规格说明进行最终审查。系统测试可以验证系统是否达到需求，每个功能模块是否存在着一些缺陷或错误引发系统不正常的运行[[2]](#footnote-2)[8]，从而保证了系统的准确性与可靠性。

本系统在完成每一个小功能之后，都会进行单元测试，以此来找到存在的BUG并解决，同时可以检验功能实现是否达到了预期的效果。在系统完成后使用黑盒方法对本系统进行测试。

# 6.3 测试用例

| **编号** | **用例名称** | **预期结果** | **实际结果** |
| --- | --- | --- | --- |
| CS0101 | 用户登录 | 用户成功登录，进入用户页面 | 用户成功登录，进入用户页面 |
| CS0102 | 用户注册 | 用户成功注册，账户信息被保存 | 用户成功注册，账户信息被保存 |
| CS0103 | 查询订单 | 用户成功查询到自己的订单信息 | 用户成功查询到自己的订单信息 |
| CS0104 | 预定房间 | 用户成功预定房间并收到预定成功的通知 | 用户成功预定房间并收到预定成功的通知 |
| CS0105 | 查询个人信息 | 用户成功获取自己的个人信息 | 用户成功获取自己的个人信息 |
| CS0106 | 修改密码 | 用户成功修改密码并能使用新密码登录 | 用户成功修改密码并能使用新密码登录 |
| CS0107 | 评价酒店 | 用户成功提交酒店评价，评价信息被保存 | 用户成功提交酒店评价，评价信息被保存 |
| CS0201 | 管理员登录 | 管理员成功登录，进入管理员页面 | 管理员成功登录，进入管理员页面 |
| CS0202 | 用户管理 | 管理员成功编辑和删除用户，并查看用户列表 | 管理员成功编辑和删除用户，并查看用户列表 |
| CS0203 | 订单管理 | 管理员成功查看订单列表，接受或拒绝订单并状态正确更新 | 管理员成功查看订单列表，接受或拒绝订单并状态正确更新 |
| CS0204 | 房间管理 | 管理员成功添加、编辑和删除酒店房间信息，并查看房间列表 | 管理员成功添加、编辑和删除酒店房间信息，并查看房间列表 |
| CS0205 | 评论查询 | 管理员成功查看用户对酒店的评论列表 | 管理员成功查看用户对酒店的评论列表 |

# 6.4 结果及分析

1、CS0101 用户登录:

预期结果: 用户成功登录，进入用户页面

实际结果: 用户输入正确的用户名和密码后，成功登录并跳转到用户页面

结果分析: 用户登录功能正常工作，用户可以正确登录系统

2、CS0102 用户注册:

预期结果: 用户成功注册，账户信息被保存

实际结果: 用户填写有效的注册信息后，成功注册，账户信息被保存

结果分析: 用户注册功能正常工作，用户可以成功注册账户

3、CS0103 查询订单:

预期结果: 用户成功查询到自己的订单信息

实际结果: 用户输入正确的查询条件后，成功查询到与自己相关的订单，并展示在页面上

结果分析: 查询订单功能正常工作，用户可以准确地获取到相关订单的信息

4、CS0104 预定房间:

预期结果: 用户成功预定房间并收到预定成功的通知

实际结果: 用户选择可用的房间并填写预定信息后，成功预定房间并收到预定成功的通知

结果分析: 预定房间功能正常工作，用户可以成功预定房间并接收到相关通知

5、CS0105 查询个人信息:

预期结果: 用户成功获取自己的个人信息

实际结果: 用户打开个人信息页面后，成功获取自己的个人信息并显示在页面上

结果分析: 查询个人信息功能正常工作，用户可以正确查看自己的个人信息

6、CS0106 修改密码:

预期结果: 用户成功修改密码并能使用新密码登录

实际结果: 用户填写新的密码后，成功修改密码，并可以使用新密码登录系统

结果分析: 修改密码功能正常工作，用户可以成功修改密码并使用新密码登录

7、CS0107 评价酒店:

预期结果: 用户成功提交酒店评价，评价信息被保存

实际结果: 用户填写评价内容后，成功提交评价，评价信息被保存

结果分析: 评价酒店功能正常工作，用户可以成功提交评价并保存相关信息

8、CS0201 管理员登录:

预期结果: 管理员成功登录，进入管理员页面

实际结果: 管理员输入正确的用户名和密码后，成功登录并跳转到管理员页面

结果分析: 管理员登录功能正常工作，管理员可以正确登录系统

9、CS0202 用户管理:

预期结果: 管理员成功添加、编辑和删除用户，并查看用户列表

实际结果: 管理员通过添加、编辑和删除用户的功能成功操作用户信息，并可以查看用户列表，信息更新正确

结果分析: 用户管理功能正常工作，管理员可以有效地进行用户信息的管理和查看

10、CS0203 订单管理:

预期结果: 管理员成功查看订单列表，接受或拒绝订单并状态正确更新

实际结果: 管理员成功查看订单列表，接受或拒绝订单后，订单状态正确更新

结果分析: 订单管理功能正常工作，管理员可以正确查看和处理订单信息

11、CS0204 房间管理:

预期结果: 管理员成功添加、编辑和删除酒店房间信息，并查看房间列表

实际结果: 管理员通过添加、编辑和删除房间的功能成功操作房间信息，并可以查看房间列表，信息更新正确

结果分析: 房间管理功能正常工作，管理员可以有效地进行酒店房间信息的管理和查看

12、CS0205 评论查询:

预期结果: 管理员成功查看用户对酒店的评论列表

实际结果: 管理员成功查看用户对酒店的评论列表，并可以进行筛选和搜索

结果分析: 评论查询功能正常工作，管理员可以准确地获取用户对酒店的评论信息

通过对测试用例的执行和分析，可以确定系统的各项功能都符合预期，并没有出现明显的错误或异常。如果某些用例有问题，可以进一步调查和修复，确保系统的稳定性和功能性。

# 第7章 总结与展望

# 7.1 总结

经过几个月的艰苦奋斗，终于完成了本系统。因为之前对医院的业务需求不太熟悉，所以在开始编写本系统之前，我做了问卷调查和去了实地进行需求调研，最后确定了本系统的基本技术框架和基本业务功能。使用MySQL数据库建立完善的关系型数据库，用了SpringBoot作为后端基本框架，前端选用了Vue.js加上ElementUI进行搭建。因为对相关技术不够熟悉，在完成本系统的过程中，遇到了各种大大小小的问题。但是通过查询资料和请教我的导师，都能一一解决。

本系统的主要技术难点如下：

1、前后端分离架构实现：前端使用Vue框架，后端使用Spring Boot框架，需要正确配置和搭建两者之间的通信和交互，确保前后端能够良好地协同工作。

2、统一数据管理和交互：确保前端和后端之间的数据交互准确无误，并使用合适的API接口进行数据的传递、处理和存储，同时要考虑到数据的安全性和数据一致性。

3、订单管理和状态更新：实现订单的管理和状态更新功能，包括接受或拒绝订单，确保订单的状态正确更新并及时通知相关用户。

4、移动端适配与响应式设计：普通用户页面是一个移动端样式的web项目，在设计和开发过程中需要考虑到不同移动设备的适配和响应式设计，以提供良好的用户体验。

# 7.2 展望

展望未来，随着科技的不断进步和人们生活水平的提高，酒店管理系统将会迎来更多的发展机遇和挑战。首先，酒店管理系统将更加注重用户体验，通过优化界面设计、提高系统响应速度等方式，让用户在使用过程中感受到便捷和舒适。此外，酒店管理系统还将加强与各类智能设备的融合，如智能家居、智能客房等，为用户提供更加智能化的服务。

其次，酒店管理系统将充分利用大数据、人工智能等技术，实现对酒店业务的深度挖掘和精准分析。通过对用户行为、消费习惯等方面的数据分析，酒店可以更好地了解客户需求，提供个性化的服务。同时，酒店还可以通过数据分析优化内部管理，提高运营效率。

**参考文献**

[1]王越. JAVA编程语言在计算机软件开发中的应用[J]. 电子技术与软件工程, 2019.

[2]叶欣, 李建佳, 温煜, 赖舒婷. 基于Java语言的医疗质量管理系统设计与实现[J]. 中国数字医学, 2018.

[3]葛萌, 黄素萍, 欧阳宏基. 基于Spring MVC框架的Java Web应用[J]. 计算机与现代化, 2018.

[4]彭伟民.基于需求的酒店管理系统的建模与实现.微机发展,2005.10.1-6.

[5]张亚东.酒店企业销售管理信息系统的设计与实现[J].管理信息系统, 2000.9:45249.

[6]马秀莲,高志安.宾馆管理系统的设计.黑龙江八一农垦大学信息技术学院,2005.2.1-3.

[7]唐羽.应用型本科院校酒店管理专业人才培养模式的分析[J].中国商论,2015(34):190-192.

[8]薛华成.管理信息系统[M].清华大学出版社,1996.7.2-5.

[9]童德利,田娟,谢琪,陈世福.基于B/S模式的构件式酒店管理信息系统的设计与实现.2003.4.1-4

1. [↑](#footnote-ref-1)
2. [↑](#footnote-ref-2)