

上海理工大学光电信息与计算机工程学院

2018-2019 (2) 短学期

《计算机专业课程设计》实验报告



项目名称	酒店房间预定系统				
组长学号	1612480318	组长姓名	李冠男	组长成绩	
组员学号	1612480327	组员姓名	王明勋	组员成绩	
组员学号	1612480331	组员姓名	杨露生	组员成绩	
指导教师	张 艳	完成日期	2019.7.11	备 注	

目录

实验一 酒店房间预订系统	1
一、 项目背景	1
二、 需求分析	1
三、 总体设计	2
四、 详细设计	8
五、 数据库设计	13
六、 系统实现与测试	17
七、 总结.....	25

组长学号	1612480318	组员姓名	李冠男	主要分工	服务员与用户部分后端，数据库编写，用例图，类图，报告第一、二、三.3、四.3、五小节编写以及整合。
组员学号	1612480327	组员姓名	王明勋	主要分工	管理员部分前后端工作，用户与服务员交流模块，三.1、2、4、5、六.4 小节编写。
组员学号	1612480331	组员姓名	杨露生	主要分工	服务员与用户部分前端，以及前后端合并，项目整合，数据库建立，剩余部分报告编写。

实验一 酒店房间预订系统

一、 项目背景

当下是一个信息化的时代，越来越多的用户喜欢在网上进行消费活动，如网上订餐，网上购物等等。而酒店预定作为一种较为普遍的消费活动，自然也需要借助信息的力量来使其更加方便快捷，让顾客具有更好的消费体验，同时节约顾客的时间成本。

本系统是针对酒店的具体业务而开发的。业务管理以酒店前台管理为核心，通过使用网站的方式，为顾客提供高效的服务，减免手工处理的繁琐与误差，及时、准确的反映酒店的经营情况，提供给顾客以快捷、准确和可靠的信息和服务，同时，也提高了酒店自身的管理水平。

酒店房间预定系统前台和后台两个子模块。用户可以通过前台模块实现酒店信息的查询浏览以及房间的预定，服务员和管理员使用后台模块处理用户的预定订单，实现开房，退房以及订单统计等功能。

本系统的优势在于简化了预定酒店以及开房的操作流程，可以实时的统计订单、盈利等信息使酒店的收益可视化，使管理人员可以更方便的统计一些订单信息。

二、 需求分析

1. 功能需求分析

本系统的使用者有预定用户，服务员，管理员三种角色，各用户类型的可操作的功能如下：

预定用户的功能有：酒店信息查询，预定房间，查询订单，取消订单，与服务员交流，管理个人信息，对酒店进行评价；

服务员的功能有：处理用户预订订单，用户开房时分配房间，退房，修改用户等级，与用户交流，添加房间；

管理员实现的功能有：添加、查询、修改、删除员工。

2. 开发环境

软件环境：Windows10 环境，IntelliJ IDEA，tomcat8.5，MySQL，chrome 浏览器；

硬件环境：PC 机。

三、 总体设计

1. 体系结构

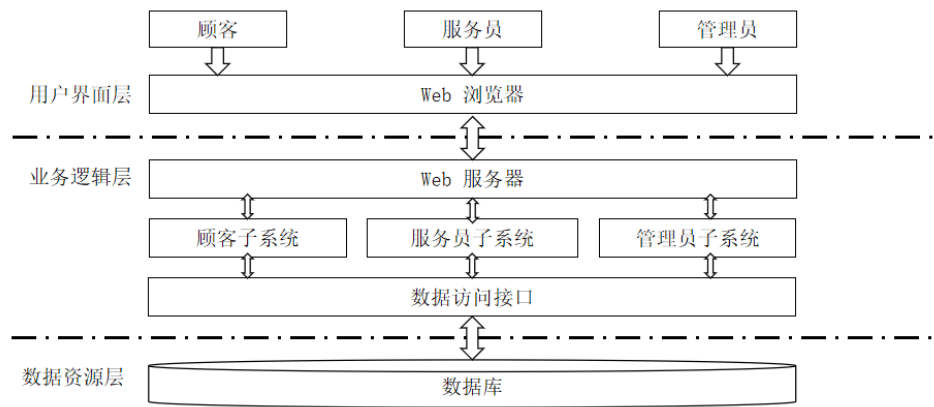


图 3-1-1 系统体系架构图

2. 功能模块图

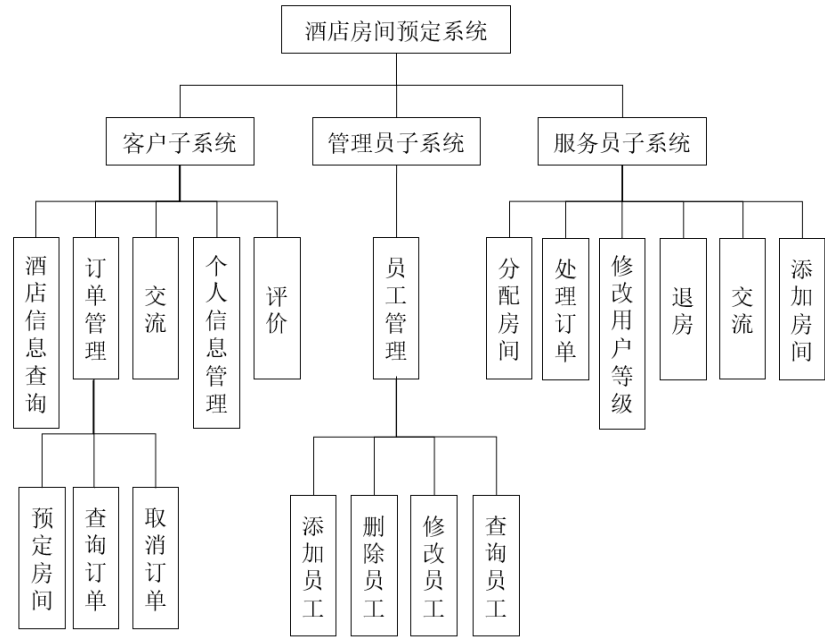


图 3-2-1 系统功能模块图

3. 用例模型

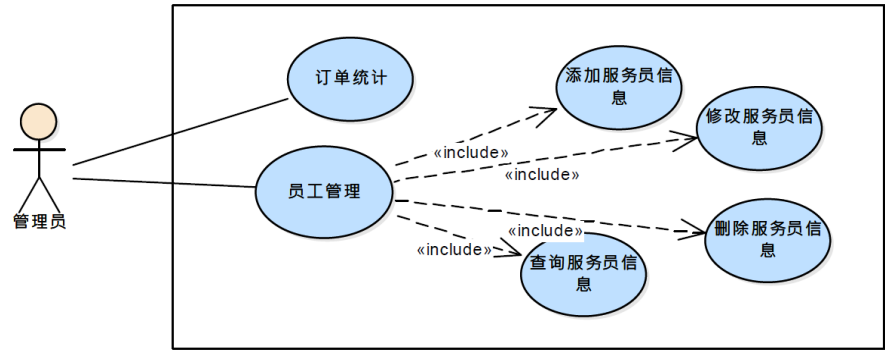


图 3-3-1 管理员子系统用例图

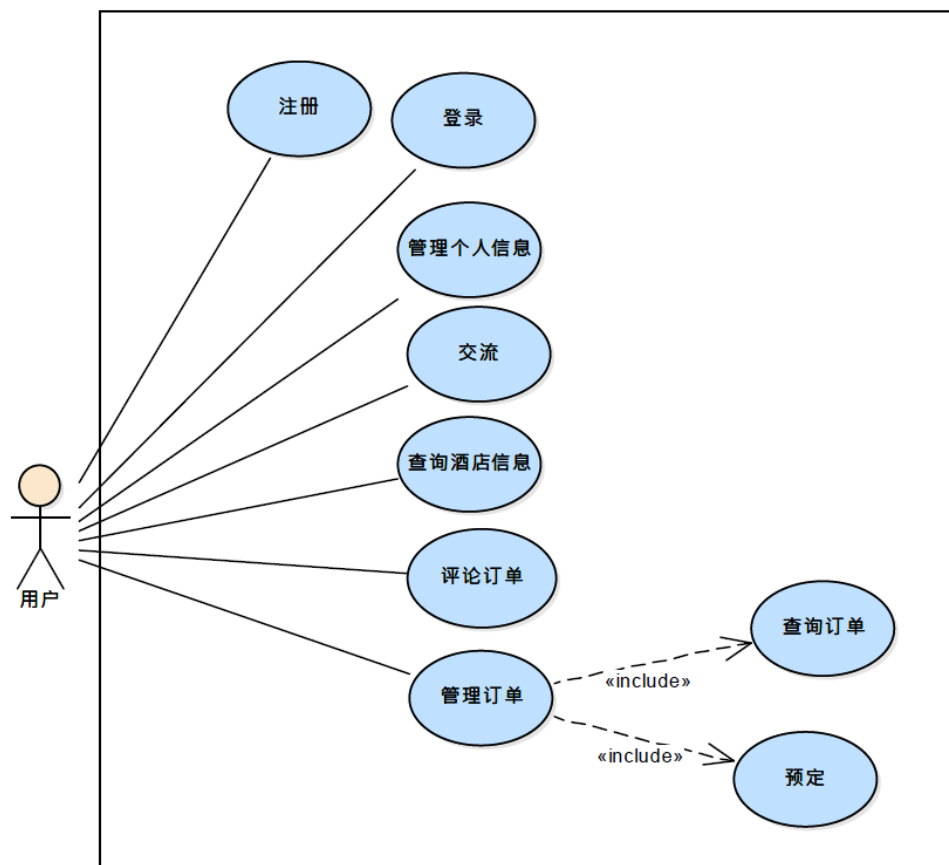


图 3-3-2 客户子系统用例图

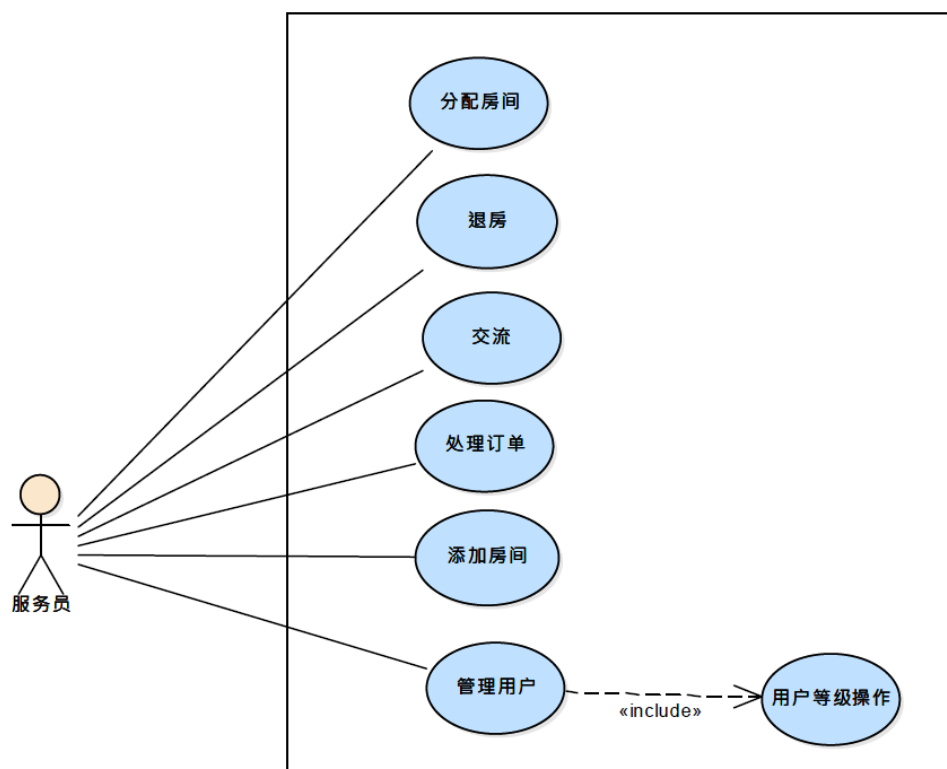


图 3-3-3 服务员子系统用例图

4. 时序图

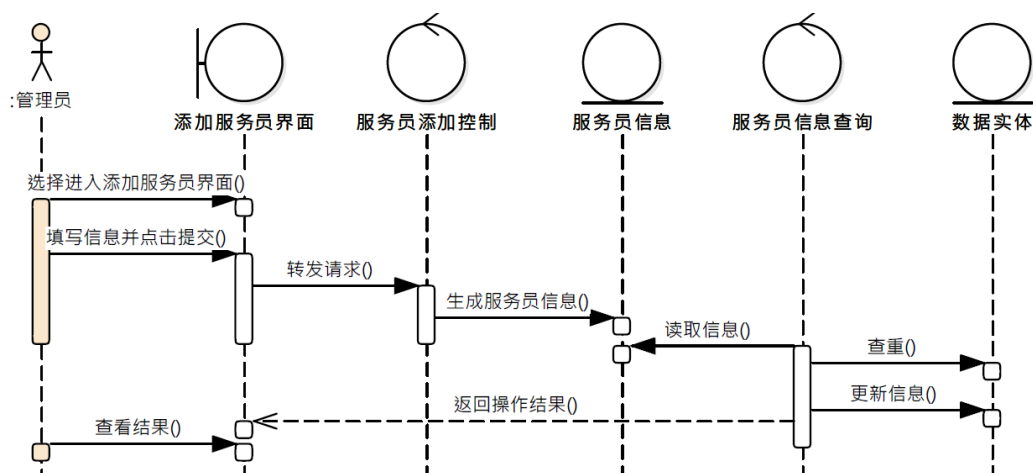


图 3-4-1 管理员添加服务员信息时序图

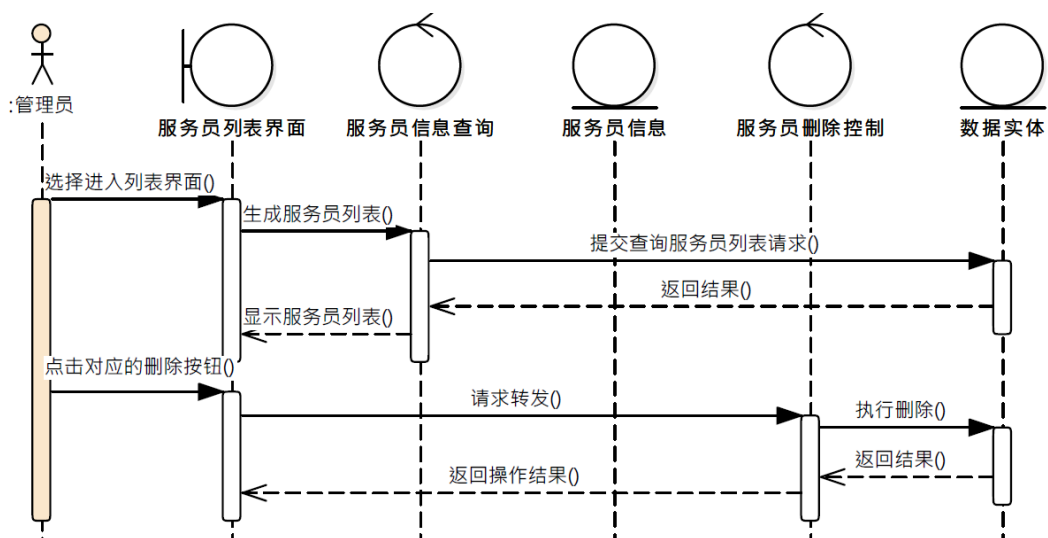


图 3-4-2 管理员删除服务员时序图

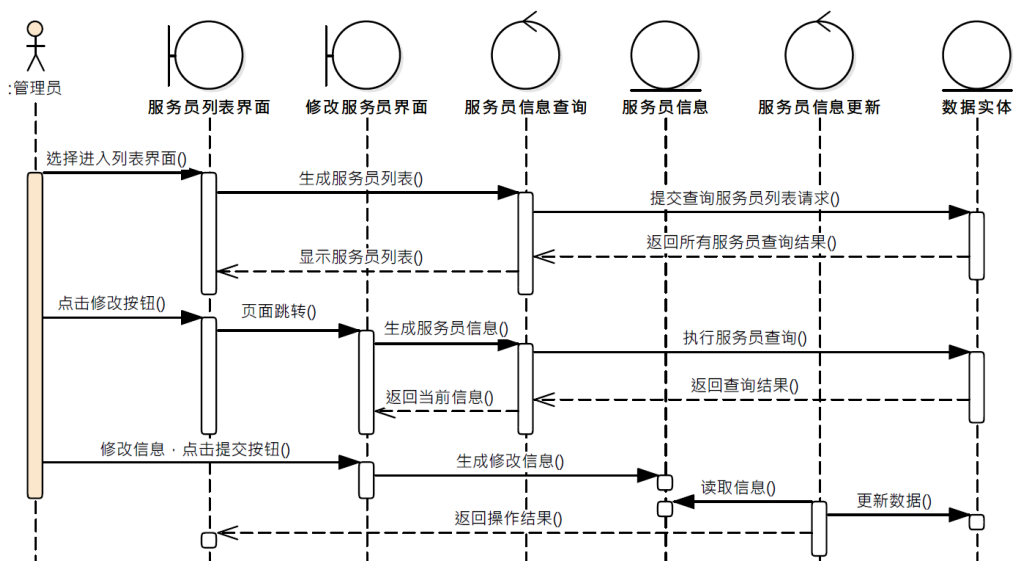
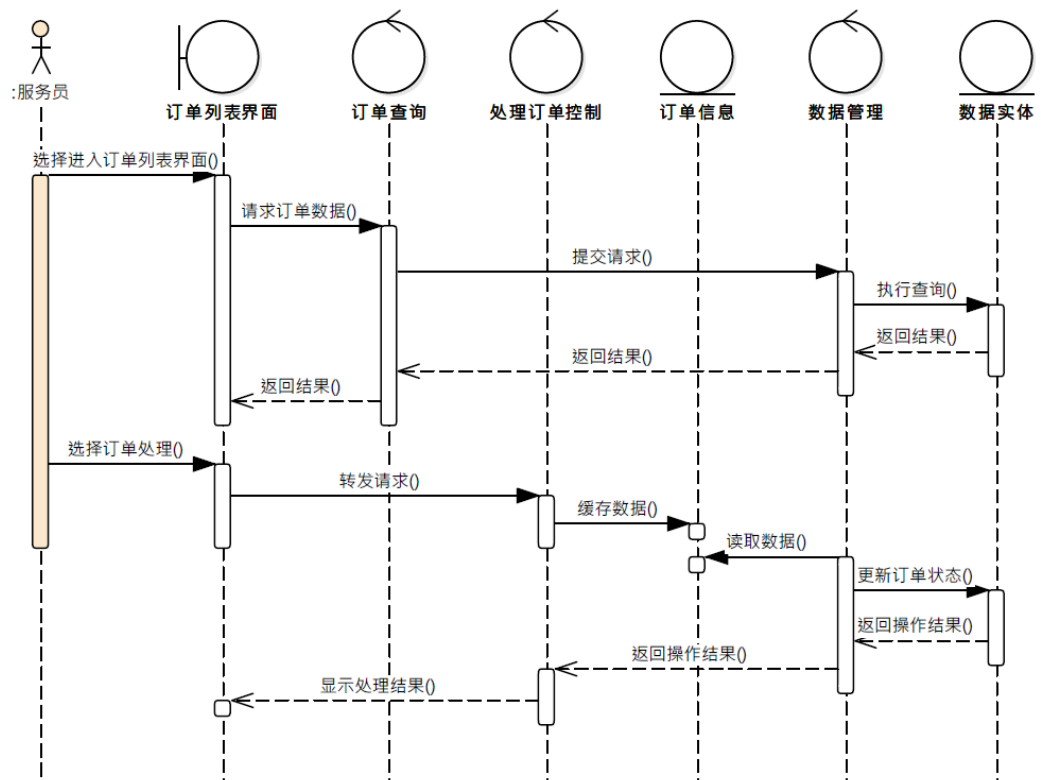
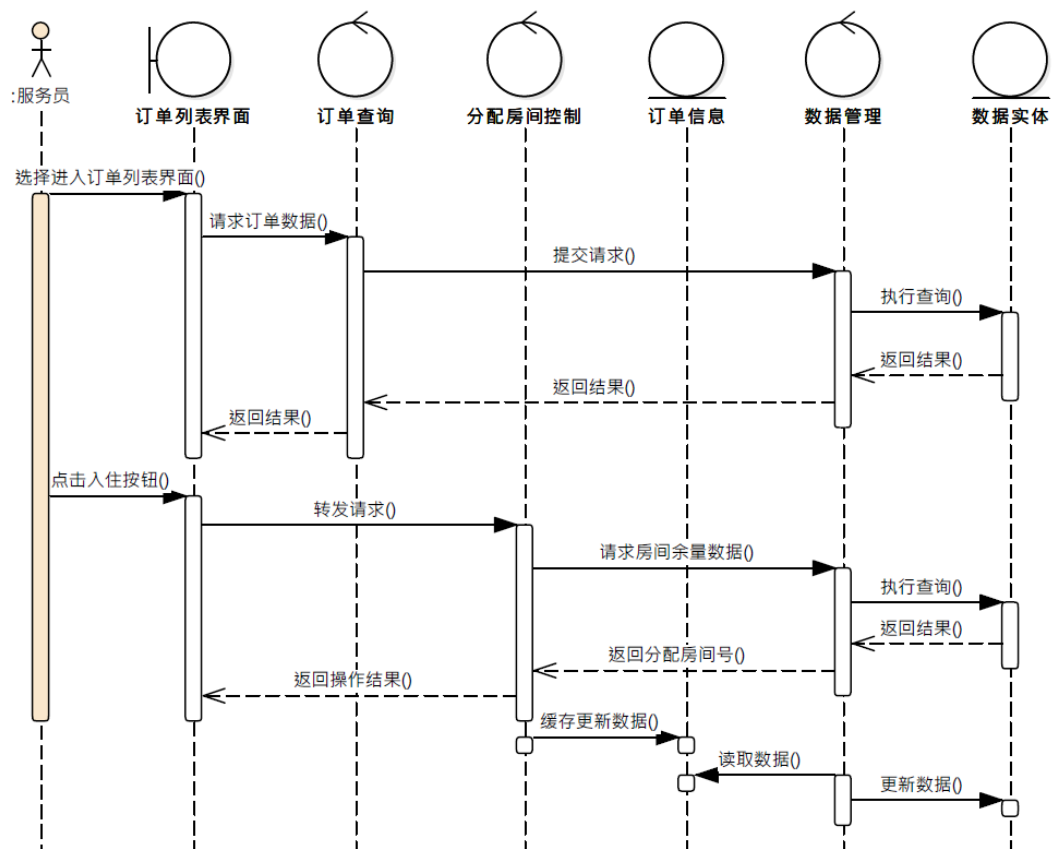


图 3-4-3 管理员修改服务员信息时序图



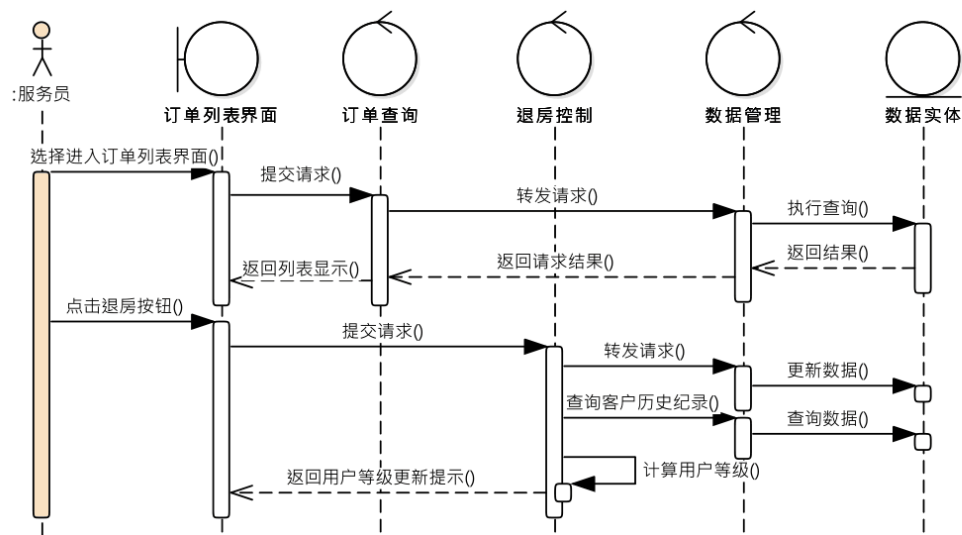


图 3-4-7 服务员退房以及处理用户等级时序图

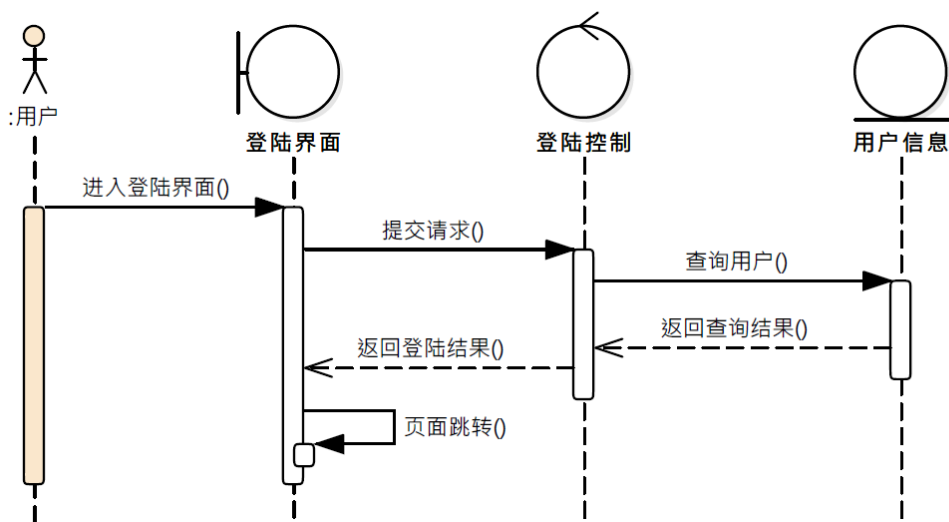


图 3-4-8 用户登录时序图

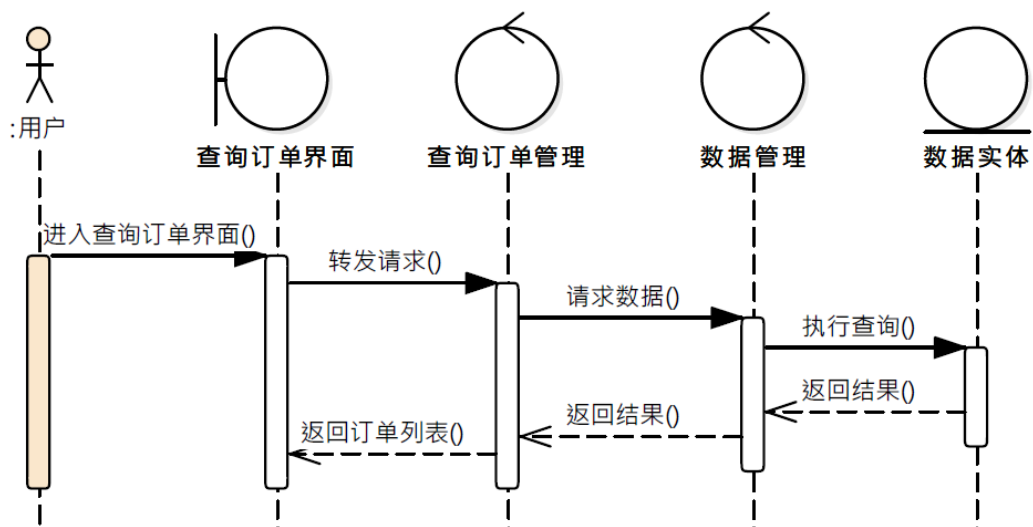


图 3-4-9 用户查询订单时序图

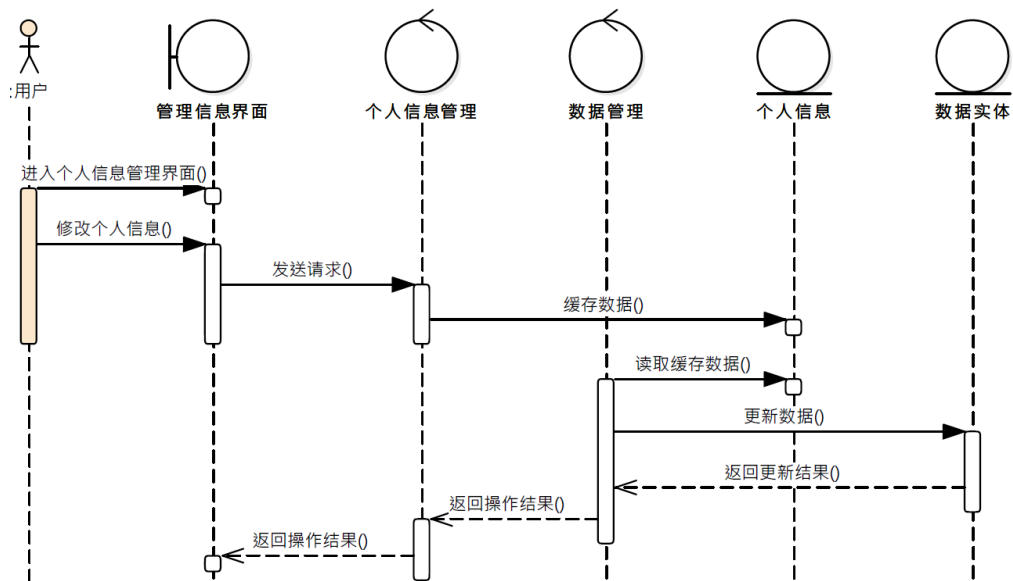


图 3-4-10 用户管理个人信息时序图

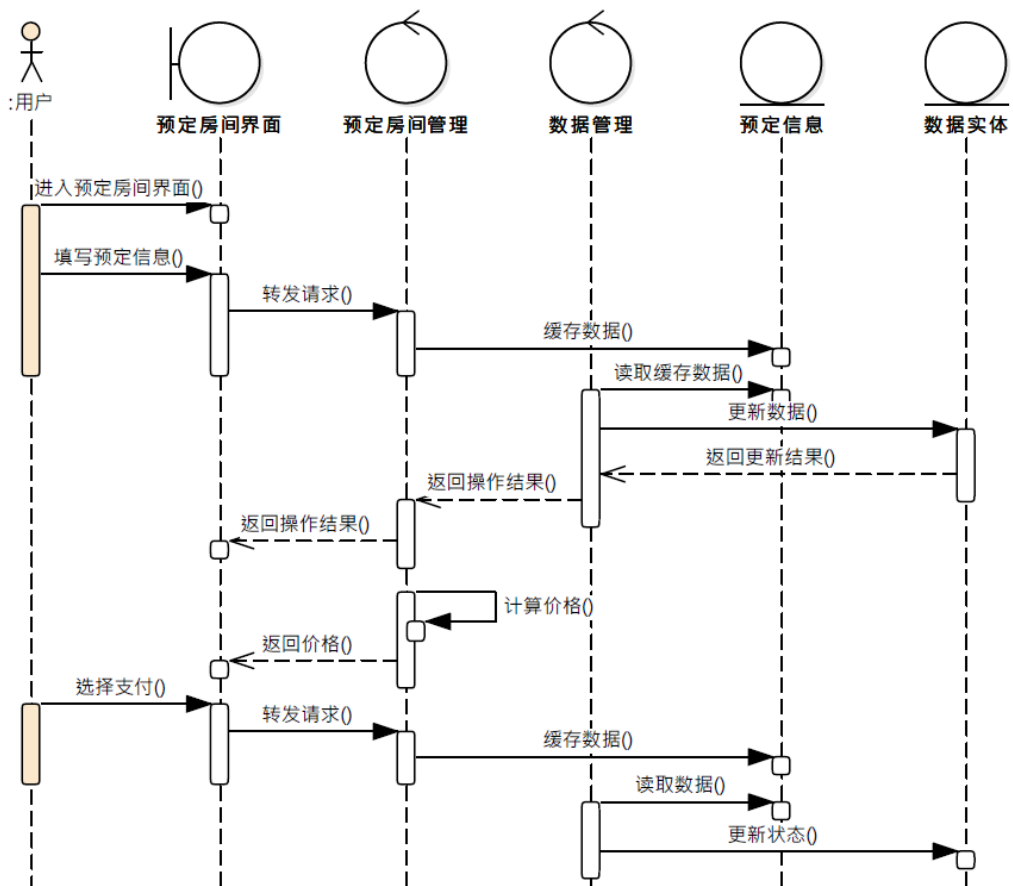


图 3-4-11 用户预订房间时序图

5. 业务流程

系统整体业务流程见图 3-5-1。

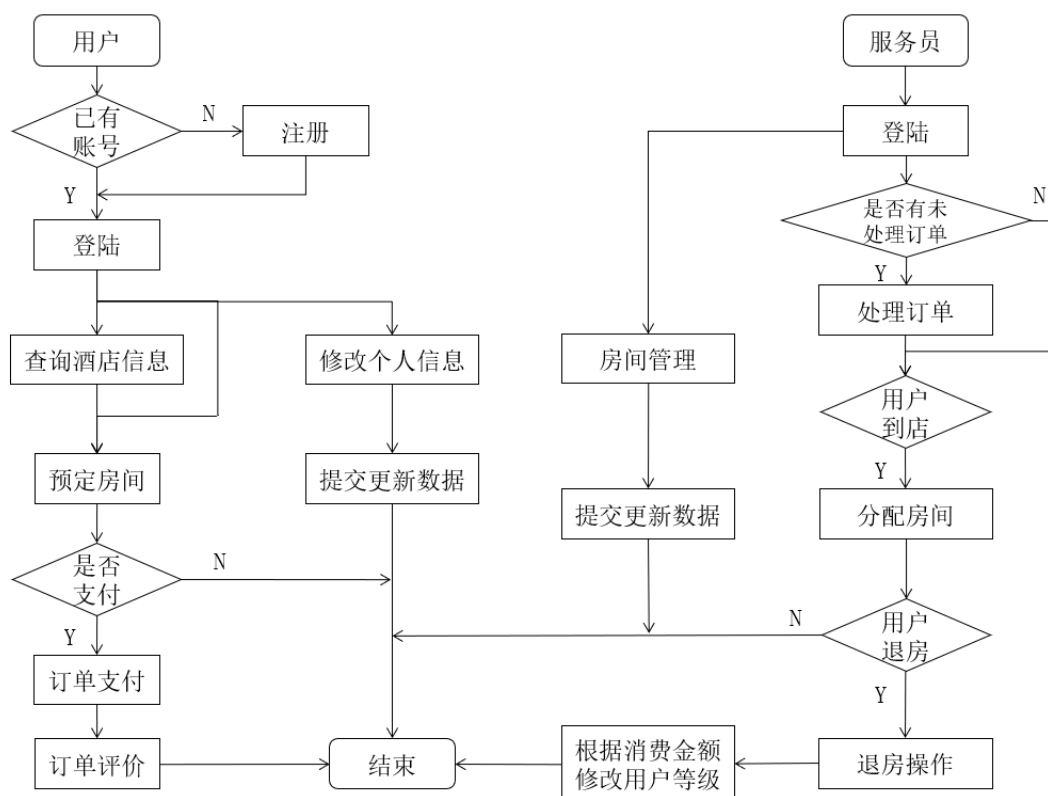


图 3-5-1 系统整体业务流程图

四、详细设计

1. 系统结构设计

本系统的实现过程中采用了 B/S 架构。前端使用 bootstrap, Vue.js, WebSocket, jQuery, thymeleaf 等开发框架与模板引擎, 后端开发使用了 Spring Boot, SpringMVC, MyBatis 开发框架。

2. 界面设计

2.1 用户登录主页

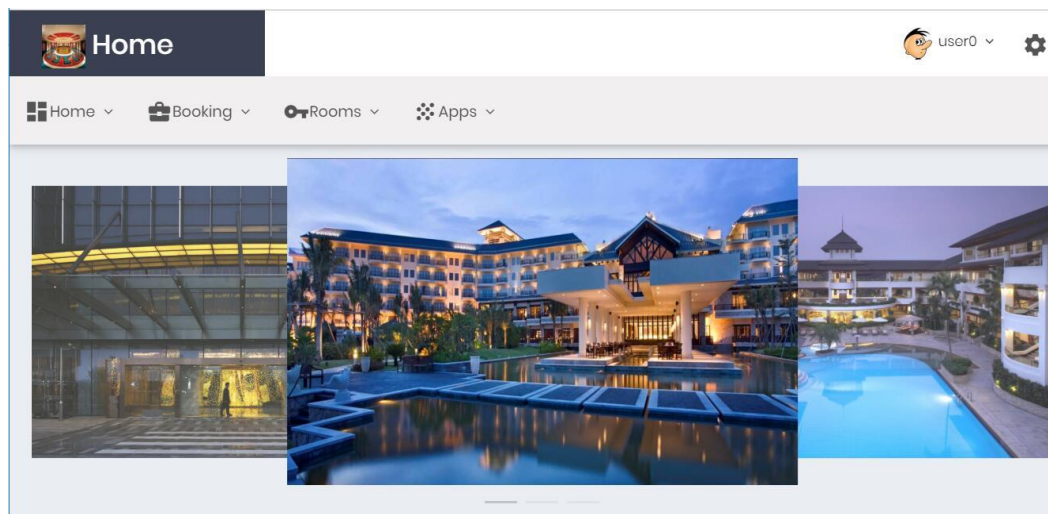


图 4-2-1 用户登录首页-1



图 4-2-2 用户登录首页-2

2.2 服务员登录主页

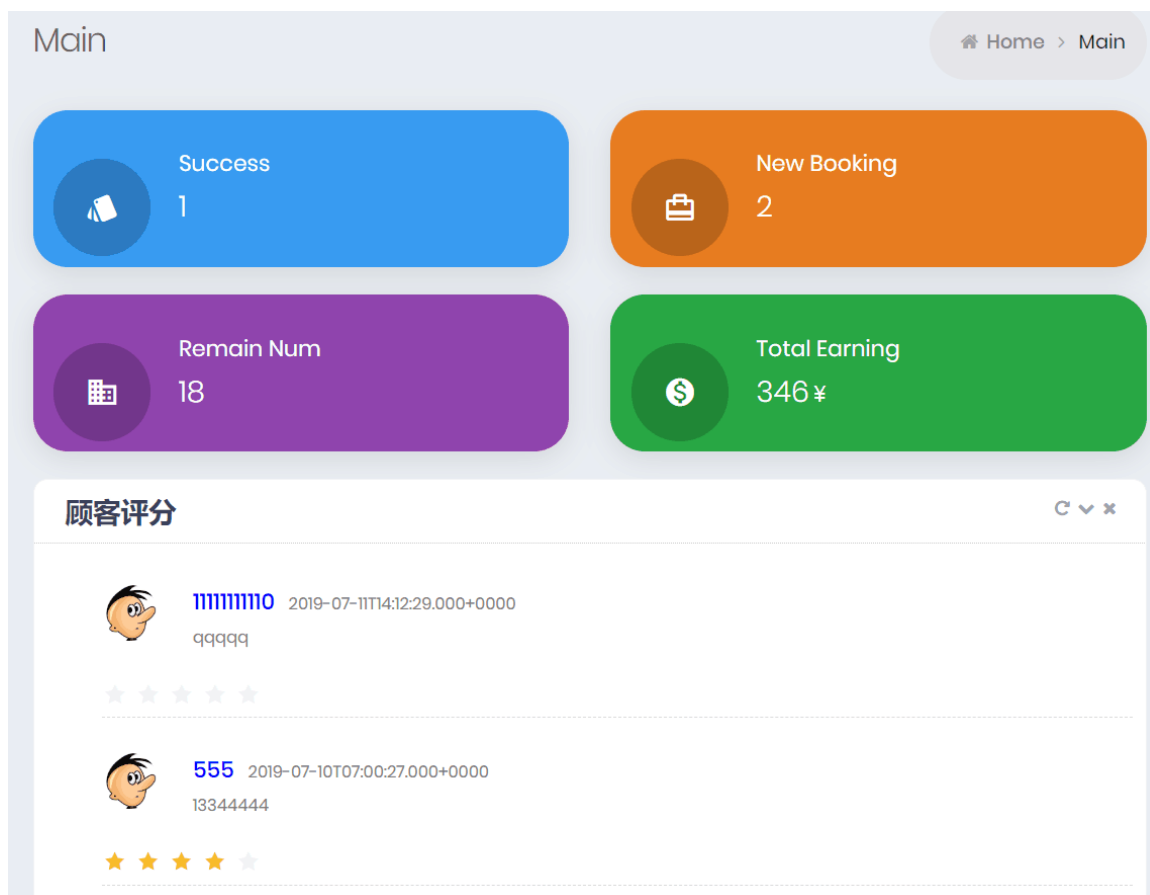


图 4-2-3 服务员登录主页

2.3 管理员登录主页

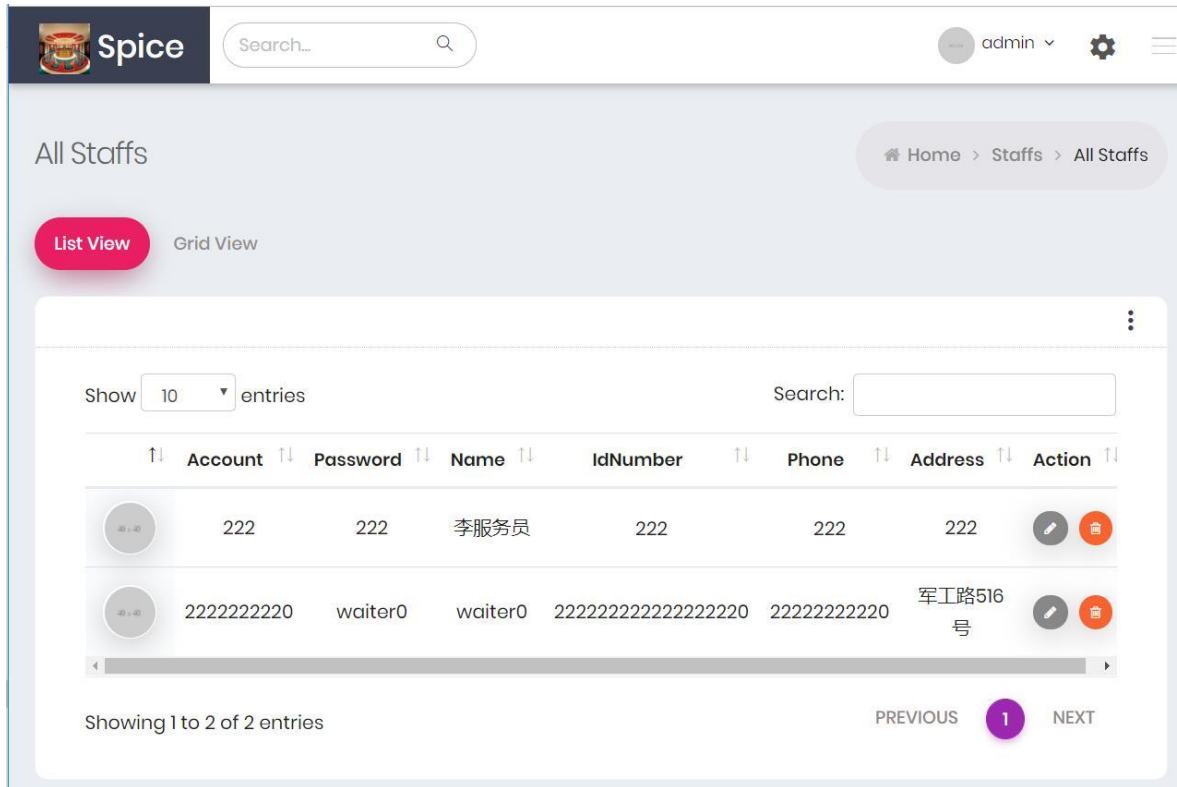


图 4-2-4 管理员登录主页

2.4 用户预定界面

The 'Add Room Booking' form contains the following fields and controls:

- Room Type:** A dropdown menu.
- No Of Room:** A dropdown menu.
- Arrive:** A date selection field with a calendar icon and the text '选择日期'.
- Depart:** A date selection field with a calendar icon and the text '选择日期'.
- Buttons:** A pink 'SUBMIT' button and a grey 'CANCEL' button.

图 4-2-5 用户订房界面

2.5 服务员开房界面

All Bookings									
ADD NEW +		Search: 请输入内容							
Phone	Room Type	Room Number	Arrive	Depart	Amount	Discount Price	Payment	Action	
1111111110	豪华标准间	1	2019-7-3	2019-7-20	¥4556	¥4100.4	待处理	同意	不同意
1111111110	单人房	2	2019-7-13	2019-7-14	¥256	¥256	待处理	同意	不同意
1111111110	单人房	1	2019-7-30	2019-7-31	¥128	¥128	预定成功	入住	
1111111110	行政房	1	2019-7-13	2019-7-14	¥458	¥458	预定成功	入住	

图 4-2-6 服务员开房界面类图

3. 类设计

3.1 整体类结构：

由于在每个功能模块中类之间的相互调用关系一致，所以在此只给出一个总体的类结构图，如下图所示：

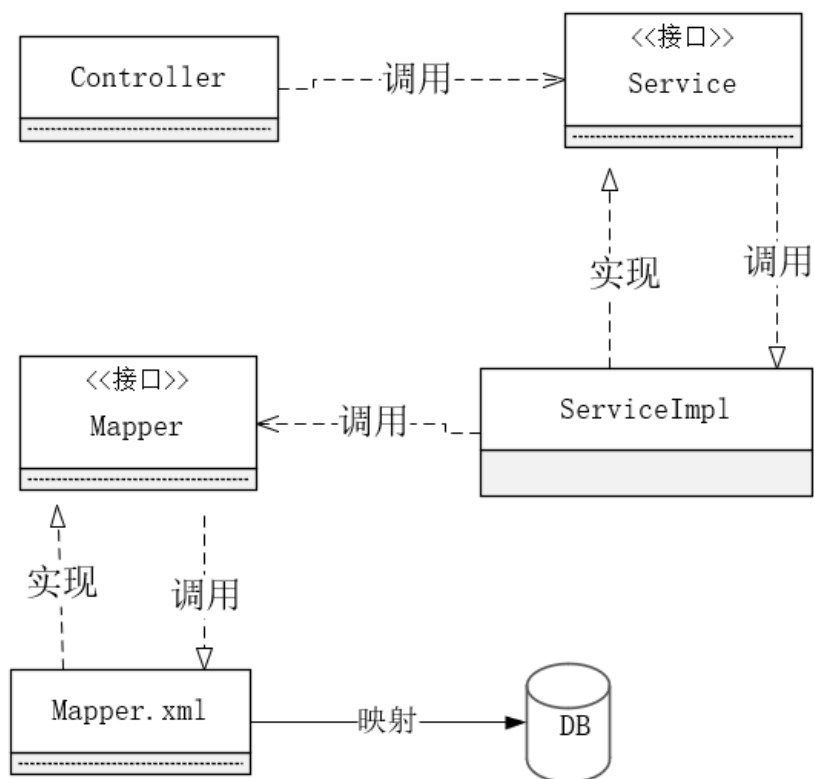


图 4-3-1 系统整体类结构图

3.2 整体类说明:

由于每个功能模块中 Controller, Service,ServiceImpl,Mapper 中方法以及方法的参数,返回值都相同,所以在此只列举一次方法说明,该方法说明适用于该功能模块下所有的类。

3.2.1 用户预定房间部分类中方法说明:

表 1 预定房间类方法说明

方法名	访问权限	参数	返回值	方法说明
queryRemainByDate	public	Date date, Stringt ypeName	remainRo om	通过日期以及房间类型查询对应类型房间的余量。
queryRemainByName	public	String typeName	remainRo om	通过房间类型查询对应类型房间的余量。
sumPrice	public	String phone	int	计算该用户消费总额。
changeBookState	public	int bookId	int	入住与退房时修改房间状态。
bookFailed	public	book book1, String waiter	int	处理预定失败流程。
queryBookByBookId	public	int bookId	book	通过订单编号查询订单。
queryAll	public	无	List<book >	查询所有订单。
bookToHistory	public	book book	int	将订单从预定表中迁移到历史列表中。
deleteBook	public	int bookId	int	删除订单。
payOrder	public	int bookId	int	支付订单。
queryBookByPhone	public	String phone	List<book >	查询该用户的所有订单。
addOrder	public	book book	int	添加订单。

3.2.2 到店开房以及退房类方法说明

表 2 到店开房类方法说明

方法名	访问权限	参数	返回值	方法说明
-----	------	----	-----	------

queryAll	public	无	List<checkIn>	查询所有在住列表信息
bookToCheckIn	public	int bookId, String waiter	int	入住时将预定表中信息迁移到在住表中。
checkInToCheckOut	public	String roomId, String checkOutWaiter	int	退房时将在住列表中信息迁移到历史列表中。
changeBookState	public	int bookId	int	入住与退房时修改房间状态。

3.2.3 服务员添加房间类方法说明

表 3 服务员添加房间类方法说明

方法名	访问权限	参数	返回值	方法说明
queryAllRoom	public	无	List<room>	查询所有房间
addRoom	public	String roomId, int typeId	int	添加房间

3.2.4 员工管理类方法说明

表 4 员工管理类方法说明

方法名	访问权限	参数	返回值	方法说明
AddStaff	public	Waiter waiter	int	管理员添加员工。
ViewStaff	public	无	List<waiter>	查询所有员工。
Delete	public	String account	int	删除对应员工。
Edit	public	String account	int	修改对应员工

3.2.5 添加评价类方法说明

表 5 添加评价类方法说明

方法名	访问权限	参数	返回值	方法说明
addComment	public	Comment comment	int	添加评价。
queryAllComment	public	无	List<comment>	查询所有评价。

五、 数据库设计

1. 数据流图

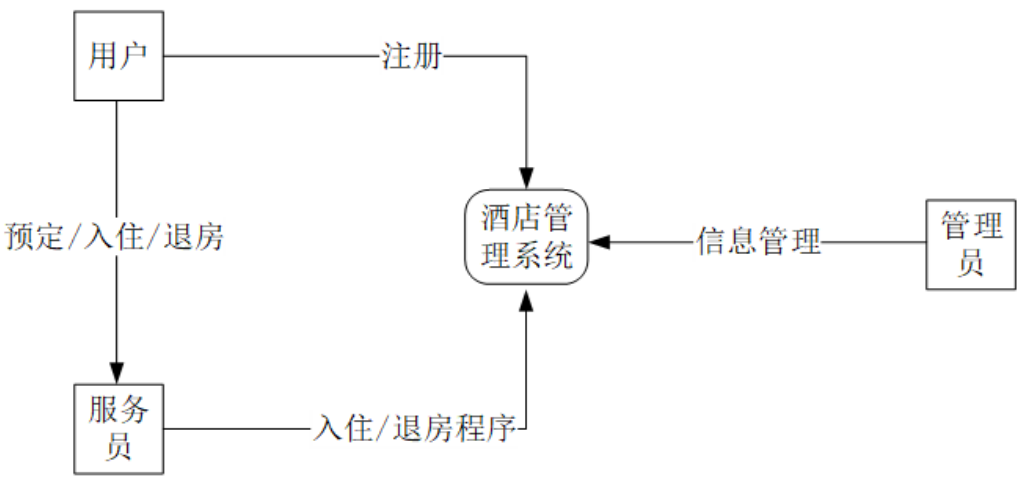


图 5-1-1 顶层数据流图

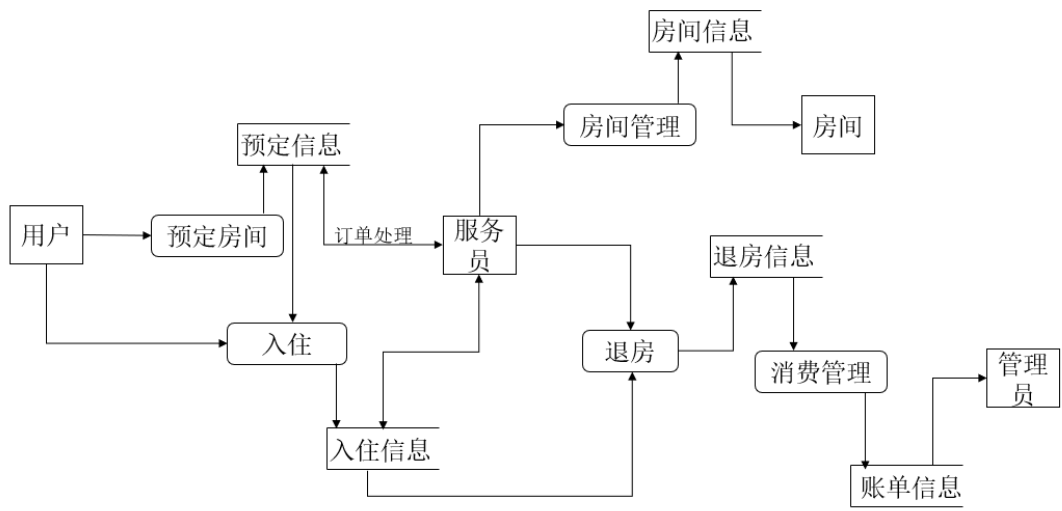


图 5-1-2 第一层数据流图(预定-离店流程)

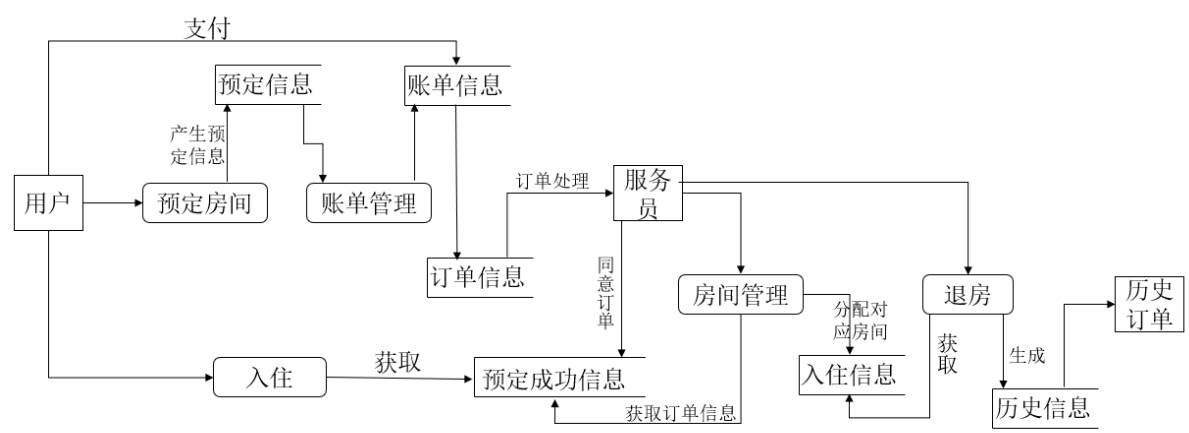


图 5-1-3 第二层数据流图(订房-离店流程)

2. ER 图

2.1 分 ER 图

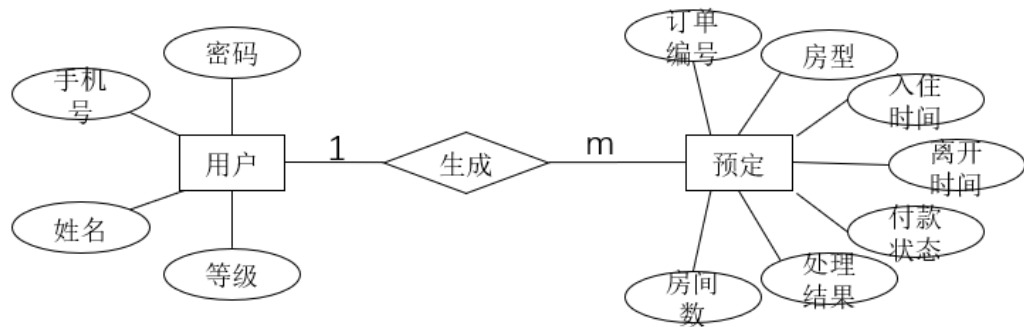


图 5-2-1 用户-预定实体联系图

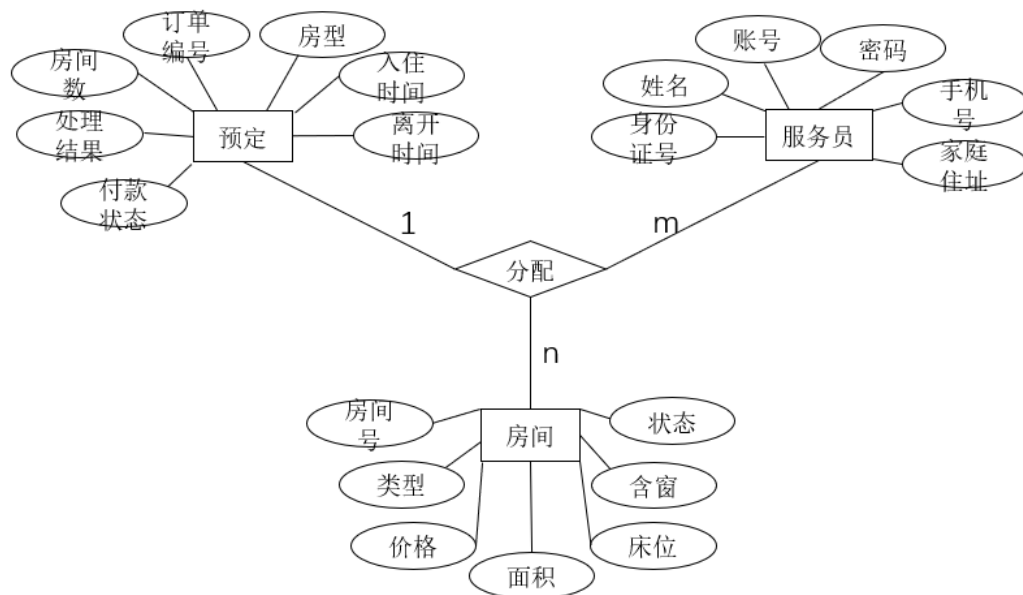


图 5-2-2 预定-服务员-房间实体联系图

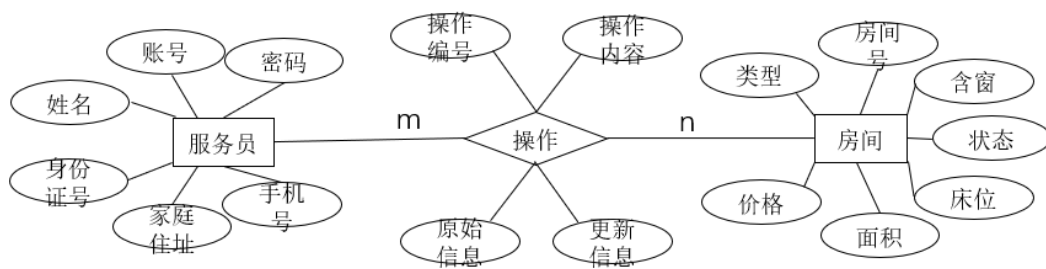


图 5-2-3 服务员-房间实体联系图

2.2 整体 ER 图

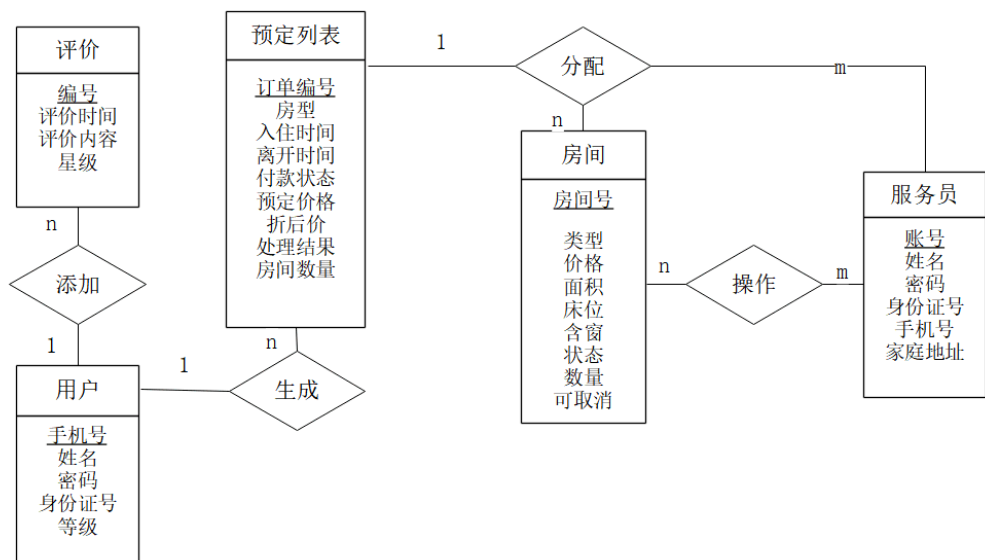


图 5-2-4 整体 ER 图

3. 关系模式

用户（手机号¹，姓名，密码，身份证号，等级）

服务员（账号，密码，姓名，身份证号，手机号，家庭住址）

预定列表（订单编号，房型，房间数，入住时间，离开时间，订单金额，实付金额，处理结果，手机号）

房间（房间号，状态，类型）

房间类型（类型编号，类型名称，价格，面积，数量，入住人数，床数，是否可取消，是否含早）

入住列表（订单编号，入住时间，离开时间，支付金额，房间号，手机号，服务员）

历史记录（订单编号，入住时间，离开时间，支付金额，状态，房间号，手机号，入住服务员，退房服务员，用户是否已读）

评论（评论编号，内容，评论时间，星级，评论人）

4. 数据库结果截图

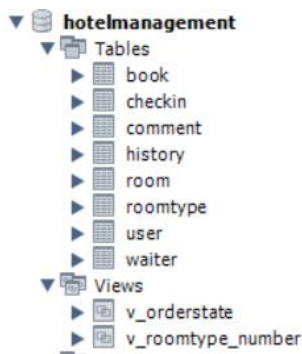


图 5-4-1 数据库结果截图

¹ 关系表中开头加下划线的属性表示主键，结尾加下划线的属性表示外键。

说明：数据库中两个视图分别保存了房间总数量与当前已占用的房间数量，为了方便计算房间余量而设计。

六、 系统实现与测试

1. 开发环境

表 6 开发环境规定

操作系统	Windows7 及以上系统
DBMS	Mysql
软件开发语言	Java
服务器部署环境	Tomcat
集成开发环境	IDEA

2. 运行环境

表 7 运行环境规定

操作系统	Windows7 及以上系统
浏览器	IE7 以上及各种主流浏览器
运行时数据库环境	MySQL

3. 开发技术

采用 B/S 架构。前端使用 bootstrap, Vue.js, WebSocket, jQuery, thymeleaf 等开发框架与模板引擎，后端开发使用了 Spring Boot, SpringMVC, MyBatis 开发框架。

4. 测试用例

4.1 接口测试用例

表 8 接口测试用例

接口 A 的函数原型	queryAllRoom	
输入/动作	期望的输出/相应	实际情况
localhost:8080/room/queryAllRoom	返回 room 表的所有内容	与预期相同
接口 B 的函数原型	addRoom	
输入/动作	期望的输出/相应	实际情况
localhost:8080/room/addRoom?roomId=0202&typeId=2	返回添加失败参数 state=1	与预期相同
localhost:8080/room/addRoom?roomId=0207&typeId=2	返回添加成功参数 state=0	与预期相同
接口 C 的函数原型	List<User> queryAll()	
输入/动作	期望的输出/相应	实际情况
localhost:8080/user/queryAll	返回 user 表所有内容	与预期相同

接口 D 的函数原型	queryByPhone		
输入/动作		期望的输出/相应	实际情况
localhost:8080/user/queryByPhone?phone=1111111115		返回查询失败参数 state=1	与预期相同
localhost:8080/user/queryByPhone?phone=1111111110		返回查询成功参数 state=0 与用户 user 信息	与预期相同
接口 E 的函数原型	queryByAccount		
输入/动作		期望的输出/相应	实际情况
localhost:8080/waiter/queryByAccount?account=222222225		返回查询失败参数 state=1	与预期相同
localhost:8080/waiter/queryByAccount?account=222222220		返回查询成功参数 state=0 与用户 waiter 信息	与预期相同
接口 E 的函数原型	queryRemainByDate		
输入/动作		期望的输出/相应	实际情况
localhost:8080/book/queryRemainByDate?date=2019/7/5&typeName= 单人间		返回查询结果：房间类型 typeName 与余量 typeNumber	与预期相同
接口 F 的函数原型	bookFailed		
输入/动作		期望的输出/相应	实际情况
localhost:8080/book/bookFailed?bookId=39&waiter		返回操作成功结果 state=0	与预期相同
接口 G 的函数原型	bookRoom		
输入/动作		期望的输出/相应	实际情况
localhost:8080/book/bookRoom 参数 book.roomType=单人间		返回预定结果 state=2，无余量	与预期相同
localhost:8080/book/bookRoom 参数 book.roomType=行政房		返回预定结果 state=0，预定成功	与预期相同
接口 H 的函数原型	queryBookByPhone		
输入/动作		期望的输出/相应	实际情况
localhost:8080/book/queryBookByPhone?phone=1111111110		返回查询成功结果 state=0	与预期相同
localhost:8080/book/queryBookByPhone?phone=1111111117		返回查询失败结果 state=1	与预期相同
接口 I 的函数原型	checkInToCheckOut		
输入/动作		期望的输出/相应	实际情况
localhost:8080/check/checkOut?roomId=0201&checkOutWaiter=2		返回操作失败结果 state=1	与预期相同
localhost:8080/check/checkOut?roomId=0201&checkOutWaiter=1		返回操作成功结果 state=0	与预期相同
接口 J 的函数原型	queryAll		
输入/动作		期望的输出/相应	实际情况
localhost:8080/check/queryAll		返回操作失败结果 state=1	与预期相同
localhost:8080/check/queryAll		返回操作成功结果 state=0，返回	与预期相同

	入住表所有信息	
接口 K 的函数原型	addComment	
输入/动作	期望的输出/相应	实际情况
localhost:8080/comment/addComment?content=111111&phone=1111111110&star=4	返回操作失败结果 state=0	与预期相同
localhost:8080/comment/addComment?content=111111&phone=1111111110&star=-1	返回操作成功结果 state=1	与预期相同
接口 L 的函数原型	queryAllComment	
输入/动作	期望的输出/相应	实际情况
localhost:8080/comment/queryAllComment	返回查询成功结果 state=0	与预期相同
localhost:8080/comment/queryAllComment	返回查询失败结果 state=1	与预期相同
接口 M 的函数原型	queryAll	
输入/动作	期望的输出/相应	实际情况
localhost:8080/roomType/queryAll	返回查询成功结果 state=0	与预期相同
localhost:8080/roomType/queryAll	返回查询失败结果 state=1	与预期相同

4.2 功能测试用例

表 9 功能测试用例

功能 A 描述	用户订房，选择房间类型，房间数量，到达日期，离开日期，点击 submit 按钮	
用例目的	测试用户订房功能是否已完善，是否满足客户要求	
前提条件	用户已注册，登陆进入预定房间界面	
输入/动作	期望的输出/相应	实际情况
选择 RoomType=单人间 No of Room=1 Arrive=2019-7-3 Depart=2019-7-15	页面显示预定成功 提示是否转跳到订单列表页面	与预期相同
选择 RoomType=单人间 No of Room=6 Arrive=2019-7-3 Depart=2019-7-15	页面显示错误信息： 房间余量不足，预定失败！ 单人间当前余量为: xxxx	与预期相同
选择 RoomType=单人间 No of Room=1 Arrive=2019-7-15 Depart=2019-7-3 (离开日期晚于到达日期)	输入框检测到日期选择错误 页面显示提示信息	与预期相同
选择	输入框检测日期选择错误	与预期相同

RoomType=单人间 No of Room=1 Arrive=2018-7-3 Depart=2018-7-15 (日期早于当前时间)		页面显示提示信息	
功能 B 描述	用户评论，输入评论内容，选择星数，点击 submit 按钮		
用例目的	测试用户评论功能是否已完善，是否满足客户要求		
前提条件	用户已注册，登陆进入主界面		
输入/动作		期望的输出/相应	实际情况
输入 评论内容: null 星级: 1		评论失败，评论不提交	与预期相同
输入 评论内容: 22222222222222222222 星级: 1		评论成功，页面显示评论信息	与预期相同
输入 评论内容: 22222222222222222222 星级: 0		评论成功，页面显示评论信息	与预期相同
输入 评论内容: 2222222.....(100 字符) 星级: 4		评论成功，页面显示评论信息	与预期相同
功能 C 描述	服务员处理订单，点击同意或不同意按钮		
用例目的	测试服务员处理订单功能是否已完善，是否满足客户要求		
前提条件	服务员存在，登陆进入预定信息界面		
输入/动作		期望的输出/相应	实际情况
点击同意按钮		页面显示处理订单成功 Action 列按钮变为入住	与预期相同
点击不同意按钮		页面显示退回订单成功 列表该行信息消失	与预期相同
Search 框 输入: 单人间		列表改变，显示字段包含单人间的行	与预期相同
功能 D 描述	服务员开房，点击入住按钮		
用例目的	测试服务员开房功能是否已完善，是否满足客户要求		
前提条件	服务员存在，登陆进入预定信息界面		
输入/动作		期望的输出/相应	实际情况
点击入住按钮		页面显示入住成功 列表该行信息消失	与预期相同
功能 E 描述	服务员退房，点击退房按钮		
用例目的	测试服务员退房功能是否已完善，是否满足客户要求		
前提条件	服务员存在，登陆进入退房信息界面		
输入/动作		期望的输出/相应	实际情况
点击退房按钮		页面显示退房成功 列表该行信息消失	与预期相同

列表无数据	页面显示通知 所有房间处于空置状态	与预期相同
功能 F 描述	管理员添加服务员	
用例目的	测试管理员添加服务员功能是否已完善，是否满足客户要求	
前提条件	管理员登陆进入添加服务员界面	
输入/动作	期望的输出/相应	实际情况
输入 所有数据为空	页面提示必填字段 数据不提交	与预期相同
输入 Name=1 ID Number=1 Account=222222220 （已存在） Password=1	页面显示用户已存在错误 数据不提交	与预期相同
输入 Name=1 ID Number=222222222222221 （已存在） Account=1 Password=1	页面显示用户已存在错误 数据不提交	与预期相同
输入 正常数据	页面显示添加 staff 成功	与预期相同
功能 G 描述	管理员删除服务员	
用例目的	测试管理员删除服务员功能是否已完善，是否满足客户要求	
前提条件	管理员登陆进入服务员列表界面	
输入/动作	期望的输出/相应	实际情况
点击某一行删除按钮	页面显示操作成功结果 数据库更新	与预期相同
点击某一行删除按钮 该服务员存在正在处理的订单	页面显示操作失败结果 数据库不更新	与预期相同
功能 H 描述	管理员修改服务员信息	
用例目的	测试管理员修改服务员功能是否已完善，是否满足客户要求	
前提条件	管理员登陆进入服务员列表界面	
输入/动作	期望的输出/相应	实际情况
前置操作： 点击某一行修改按钮	页面跳转，显示服务员当前信息	与预期相同
修改： ID Number 为 222222222222220 （已存在）	页面显示操作失败结果 数据库不更新 返回服务员列表界面	与预期相同
修改： ID Number 为 222222222222226 （不存在）	页面显示操作成功结果 数据库更新 返回服务员列表界面	与预期相同

5. 测试结果

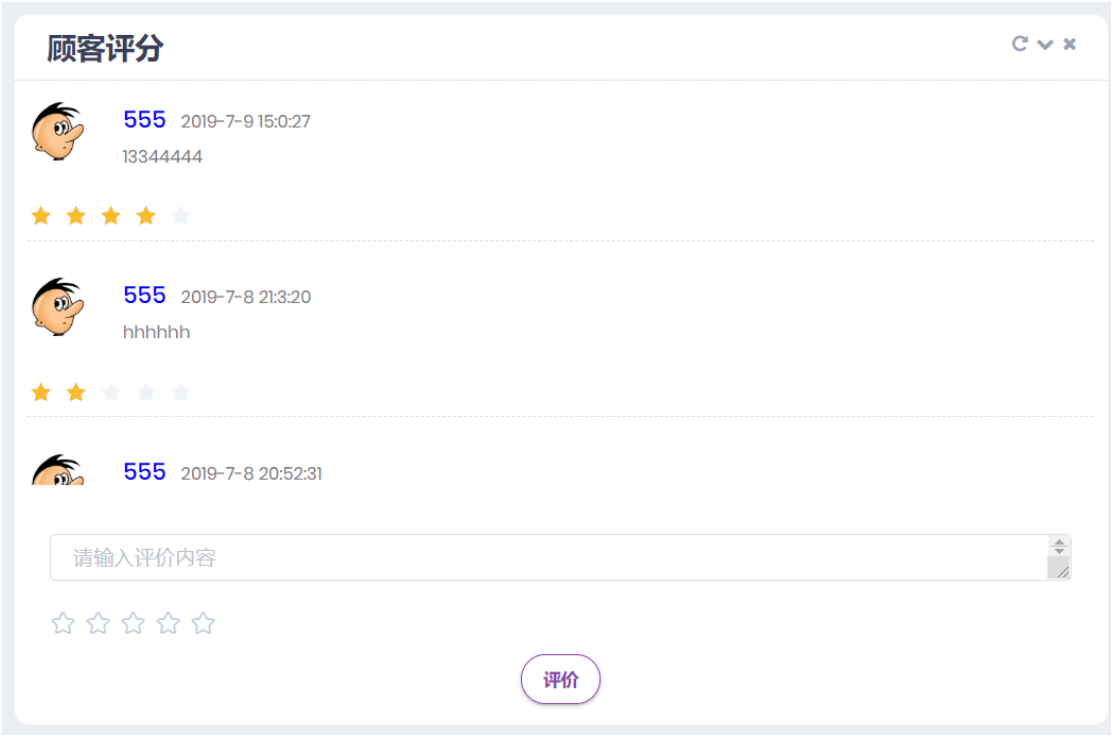


图 6-5-1 评价界面



图 6-5-2 聊天界面-用户部分

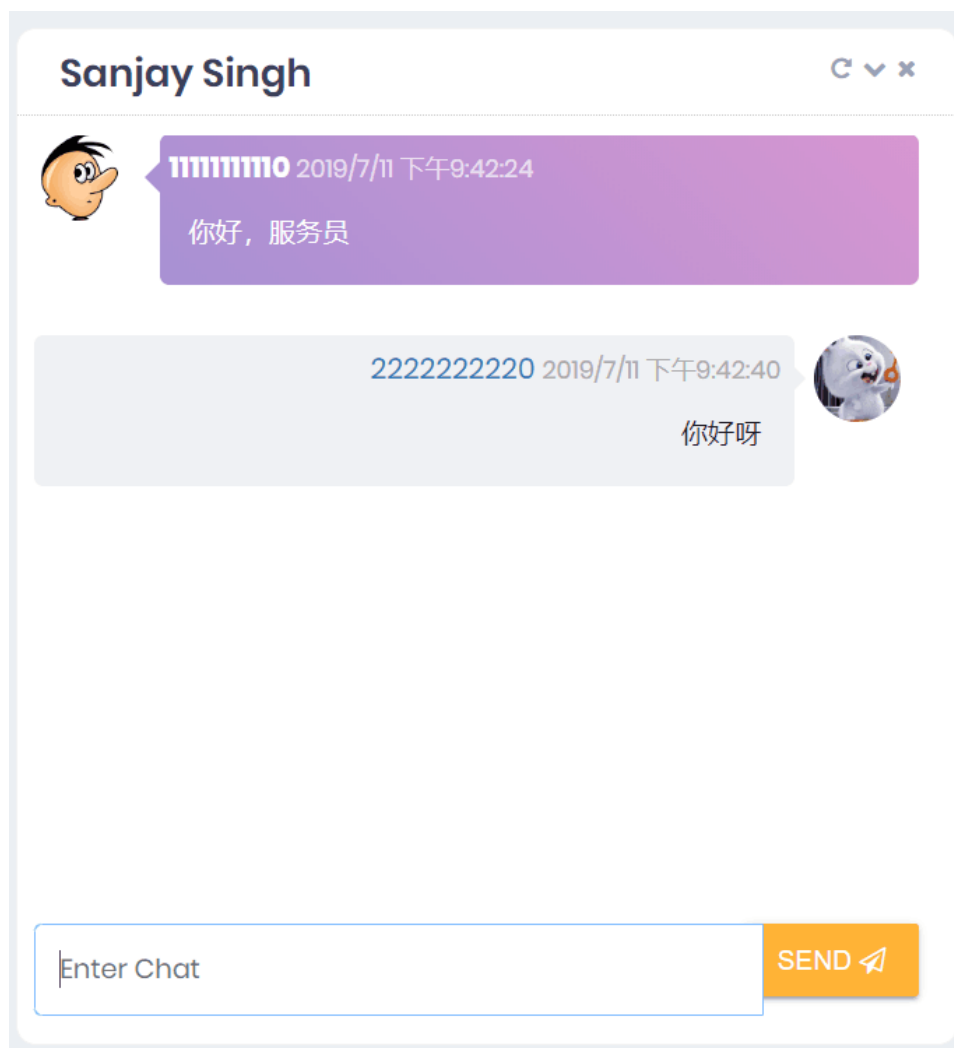


图 6-5-3 聊天界面-服务员部分

All Bookings ⌂ ⌵ ✕

[ADD NEW +](#)

	RoomType	RoomNumber	Arrive	Depart	Amount	Discount Price	Payment	Action
	单人间	1	2019-7-30	2019-7-31	¥128	¥128	待处理	取消订单
	豪华标准间	1	2019-7-3	2019-7-20	¥4556	¥4100.4	待处理	取消订单
	单人间	2	2019-7-13	2019-7-14	¥256	¥256	待支付	支付 取消订单
	行政房	1	2019-7-13	2019-7-14	¥458	¥458	待支付	支付 取消订单

图 6-5-4 用户支付与取消订单页面

All Bookings

ADD NEW +

Search: 请输入内容

Phone	Room Type	Room Number	Arrive	Depart	Amount	Discount Price	Payment	Action
	111111110 豪华标准间	1	2019-7-3	2019-7-20	¥4556	¥4100.4	待处理	<div>同意</div> <div>不同意</div>
	111111110 单人间	2	2019-7-13	2019-7-14	¥256	¥256	待处理	<div>同意</div> <div>不同意</div>
	111111110 单人间	1	2019-7-30	2019-7-31	¥128	¥128	预定成功	入住
	111111110 行政房	1	2019-7-13	2019-7-14	¥458	¥458	预定成功	入住

图 6-5-5 服务员处理订单与开房页面

Back Room

Home > Booking > Back Room

All Bookings

ADD NEW +

Search: 请输入内容


Phone	Arrive	Depart	Money	Room Id	Action
 111111110	2019-7-30	2019-7-31	¥ 128	0201	退房

图 6-5-6 服务员退房页面

List View

Grid View

Show 10 entries

Search:

Account	Password	Name	IdNumber	Phone	Address	Action
	222	222	李服务员	222	222	
	222222220	waiter0	waiter0	2222222222222220	2222222220 军工路516号	

Showing 1 to 2 of 2 entries

PREVIOUS

1

NEXT

图 6-5-7 管理员删除服务员页面

Name	李服务员
ID Number	222
Account	222
Password	222
Mobile Number	222
	222
<div>SUBMITCANCEL</div>	

图 6-5-8 管理员修改服务员信息页面

七、 总结

1. 项目收获

团队中每个成员大都没有过大型项目的开发经验，也没有过协同开发的经验，更没有一起合作的默契。但是从开始合作的陌生感到工作过程中的慢慢磨合，再到最后开发工作的圆满完成，离不开团队中每一个成员的相互配合，积极工作，最后达到圆满完成工作的目标。

在本次的合作开发中，总体工作完成的比较好，但是由于缺乏经验，在项目开发过程中难免会出现一些问题，不过在老师、同学以及团队成员的相互配合下，最终克服重重困难，得到了一个较为满意的结果，回顾整个开发过程，有以下的经验可以供以后的开发工作参考：

1.1 项目的分析阶段

在项目分析阶段中，首先从系统的使用对象入手，结合系统的设计目标，分

析出每个角色要实现的功能，继而得到具体的用例，为后续的计划以及设计工作做准备。用例的设计一定要准确，这样在后续的工作中才能准确的针对每一个用例设计其实现方法，如果出现用例之间概念重复或者功能重复等问题，会给后续的设计工作带来许多麻烦。

比如在我们前期的分析过程中针对顾客的用例，分析出了“预定房间”和“支付”两个用例，其实仔细分析之后会发现，“支付”其实是包含在“预定房间”用例之中的，可以将支付功能包含在预定房间之中。但是我们在后来的审查工作中才发现的这个问题，所以不得不重新分析并修改用例，这样重复的工作就在无形中浪费了开发时间，增加了工作量，减缓了工作的进度。

所以，分析阶段的工作一定要准确，否则就会给后续的工作带来许多麻烦。

1.2 项目的设计阶段

我认为在项目开发的整个过程中设计阶段所占的比重应该最大。因为设计阶段完全是为开发阶段做准备。从系统架构设计到原型设计，再到数据库设计，接口设计以及详细的类以及方法设计，开发过程离不开这个阶段的设计工作。

比如在我们的开发过程中，由于在接口设计时没有统一标准，在最后子模块组合的时候耗费了大量的精力和时间，导致后续阶段的测试工作不得不推迟。在设计数据库时，起初对一些实体的属性没有做明确的约束，或是类型确定错误，导致编写代码时，容易遇到数据类型错误，或是越界情况，使开发时的修改数据库工作变得较为频繁。如果在详细设计阶段这些工作全部都足够准确并且按照标准完成的话，就能为后续的编码阶段节省大量时间与人力，保证工作的按时完成。

1.3 项目的开发阶段

在开发阶段，我们应该严格根据设计阶段的标准进行编码，这项工作将为后续的维护以及模块的组合带来极大的便利。比如在我们的开发过程中，对于用户操作的约束以及提示信息没有第一时间考虑完整，导致后续添加工作变得较为繁琐，造成了相关工作的延误。

1.4 项目的测试阶段

测试时软件开发过程中必不可少的一个阶段，经过测试我们可以得到所开发系统的缺陷，为以后对系统的维护与改进提供依据。测试阶段的主要工作是测试用例的设计，最后是使用设计的测试用例对系统进行测试。测试工作应当按照测试计划严格进行，以达到按时交付的目的。在编写测试用例时我们要注意包含边界值、正确值以及错误值，并分别进行测试。测试用例的编写必须要注意细化到每一个功能点；在实际进行测试时，发现问题要及时进行记录，并及时反馈给相应的开发人员，防止 Bug 修改遗漏导致系统错误。

2. 系统待改进之处

由于时间比较仓促，没有做出具有团队特色的功能，也没有实现用户续房的功能，而是将续房的功能放在预定功能下来实现，功能不够完整，比如在最初计

划的根据用户的消费记录在用户每次登录系统时产生对该用户的个性化推荐房型的功能，由于时间问题最后不能实现，不得不将该功能删除，争取在课后空闲时间将该功能完善。

3. 致谢

在本次项目开发过程中，感谢各位团队成员的互相配合以及互相帮助，感谢张老师在开发过程中为我们解答疑惑，才得以使得整个项目如期完成。最后，再次感谢参与开发与指导的全体成员以及老师！