

# “C++程序设计与训练”课程大作业

## 项目报告

项目名称：酒店预订与管理系统

姓名：     缪谨蔚    

学号：     2017012002    

班级：     自 75    

日期：     2018 年 8 月 4 日

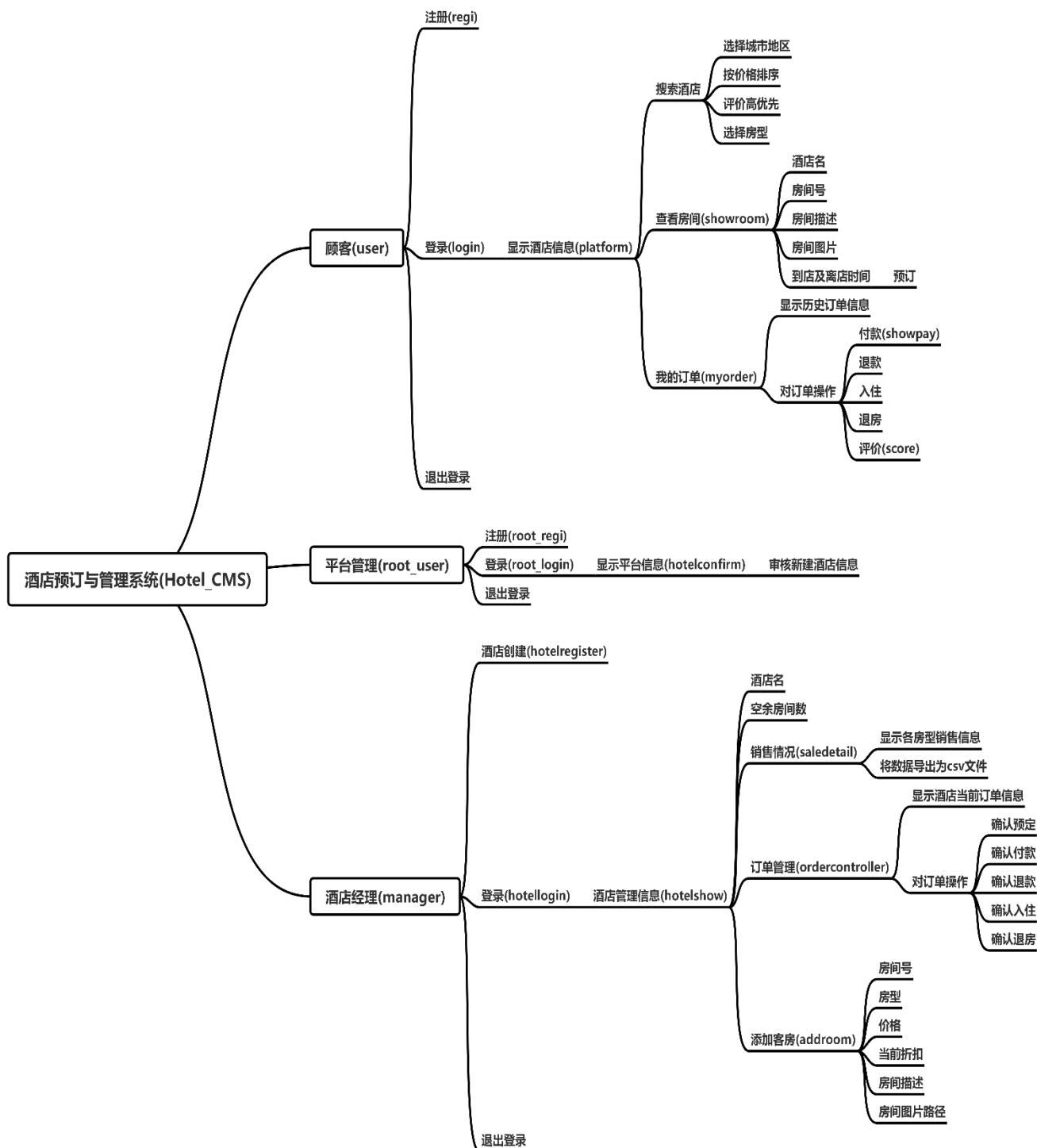
## 目录

1	系统功能设计 .....	3
1.1	总体功能描述 .....	3
1.2	功能流程描述 .....	4
1.2.1	登录/注册流程 .....	4
1.2.2	顾客订房流程 .....	4
1.2.3	酒店经理功能 .....	5
1.2.4	平台管理功能 .....	5
2	系统结构设计 .....	5
2.1	数据模块 .....	5
2.1.1	酒店类(包含经理类) .....	6
2.1.2	房间类 .....	6
2.1.3	订单类 .....	6
2.1.4	酒店注册类 .....	7
2.1.5	用户类、平台管理类 .....	7
2.1.6	其他数据相关类 .....	7
2.2	界面模块 .....	7
3	系统详细设计 .....	7
3.1	类结构设计 .....	7
3.1.1	用户、经理、平台管理类 .....	7
3.1.2	酒店类 .....	8
3.1.3	房间类 .....	8
3.1.4	订单类 .....	8
3.1.5	注册界面类 .....	8
3.1.6	登录界面类 .....	8
3.1.7	订房界面类 .....	9
3.1.8	查看房间界面类 .....	9
3.1.9	我的订单类 .....	9
3.1.10	审核界面类 .....	9
3.1.11	经理界面类 .....	9
3.1.12	销售情况界面类 .....	10
3.1.13	订单管理界面类 .....	10
3.1.14	添加客房界面类 .....	10
3.2	界面结构设计 .....	10
3.2.1	界面跳转关系 .....	10
3.2.2	界面风格设计 .....	11
3.3	容错功能设计 .....	11
4	项目总结 .....	12
4.1	项目亮点 .....	12
4.1.1	界面美观 .....	12
4.1.2	功能完善 .....	12
4.2	项目难点 .....	12
5	心得体会 .....	13
6	相关问题的说明 .....	13

# 1 系统功能设计

## 1.1 总体功能描述

本项目实现了一套多用户酒店预订与管理系统。利用该系统，顾客可以方便快捷地在平台上查询酒店信息，预订客房；酒店经理可以创建酒店并设置酒店的各项信息，可以接受顾客预订客房并管理订单的生命周期，还可以查看各房型营业情况数据，并导出为 CSV 数据报表；平台管理可审核验证新建酒店信息，并整合平台上的酒店信息。



## 1.2 功能流程描述

### 1.2.1 登录/注册流程

程序开始时显示欢迎界面。可以通过点击按钮来进入用户登录、用户注册、平台管理登录、平台管理注册、酒店经理登录和酒店创建等六个界面。

新用户注册时需要输入用户名、手机号和密码，并确认密码，两次密码一致时方可注册成功；已有账号的用户通过手机号和密码登录，当密码正确时，可以登录到对应用户类别的界面；酒店创建时需要输入酒店的名称、城市、地区、经理姓名、经理密码等信息，待平台审核通过后方可运营。

### 1.2.2 顾客订房流程

顾客订房是本项目需要实现的核心功能，其流程也是所有环节的中心。在本环节中，顾客与平台、酒店之间进行交互，实现消息的收发和状态的更新。整体流程具体如图 1-1 所示。

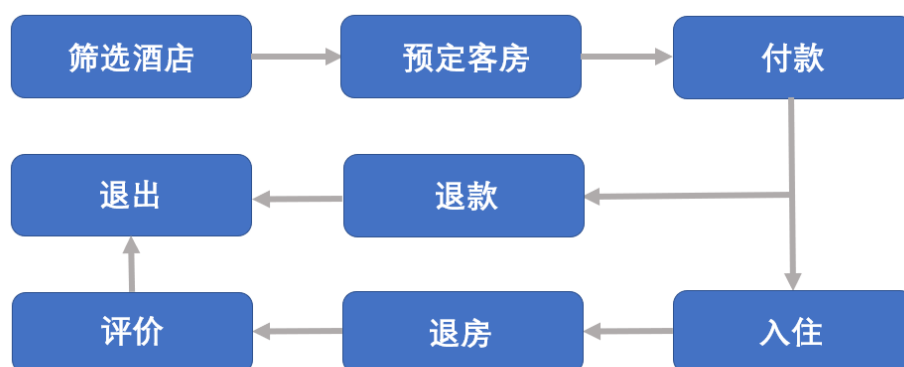


图1-1 顾客订房流程

**筛选酒店：**在本环节，顾客可以查看平台上的酒店信息，包括酒店的位置、电话、房间描述、客房定价、客房评价等信息；可以对符合特定条件的酒店和客房进行搜索，例如查询某一城市、某一地区的酒店、某种房型的客房等；可以对搜索到的酒店或客房按照一定的规则排序显示，例如价格高低、评价好坏等。

**预订客房：**顾客在筛选酒店和房型后，输入入住和离店时间后点击预订即可生成客房订单，并立刻发往酒店管理平台，此时订单状态为“预订待确认”。酒店经理在确认房间空余后，选择“确定预订”，此时顾客可以看到订单状态更新为“预订已确认”。

**付款：**顾客可以选择支付宝或微信支付。顾客点击支付完成后，会自动给酒店管理平台发送“付款”消息。酒店经理解决该消息后，订单状态变为“付款已确认”。

**退款：**顾客在付款后，在入住酒店前，如无法按时入住酒店，可选择点击退款终止订单，酒店经理解决该消息后，订单状态变为“退款已确认”。

**入住：**顾客在预定日期抵达酒店，在酒店前台登记入住后，由酒店经理进行订单信息完善，此时订单状态变为“入住已确认”。

**退房：**顾客在酒店入住至离店日期，在酒店前台办理退房后，由酒店经理进行订单信息完善，此时订单状态变为“退房已确认”。

**评价：**顾客在完成退房环节后，点击进入评价界面，对本次住店体验进行打分评价。点击提交后，评论会呈现在对应房型的历史评价数据中。程序退出，该订单的生命周期结束。

### 1.2.3 酒店经理功能

**酒店创建：**作为酒店管理层的核心，酒店经理可以进行酒店的创建，设置酒店的原始信息，包括酒店名称、所处城市、所在地区、经理姓名、经理密码。

**添加客房：**待酒店成功创建后，酒店经理可以对酒店及客房信息进行管理和维护，可添加不同类型的客房，并分别输入房间号、价格、当前折扣和房间描述，选择房型和房间图片上传路径。

**订单管理：**酒店前台从订单生成开始，贯穿着订单的全生命周期，是酒店预订与管理系统十分重要的一环。平台向酒店发起订单后，酒店经理负责顾客的预订、付款、退款、入住登记、退房结算的一系列工作，并全程参与订单流程，负责订单状态更新和管理，并可全程查看订单状态：预订待确认、预订已确认、付款待确认、付款已确认、退款待确认、退款已确认、入住待确认、入住已确认、退房待确认、退房已确认。

**销售情况：**酒店经理可以查看所有房型的销售数据，包括评分和销售次数，并能够选择指定路径，将所有数据导出为 CSV 数据报表。

### 1.2.4 平台管理功能

平台管理负责着衔接顾客和酒店的核心任务。平台既能显示酒店及房型的相关信息，也能对接顾客的订单需求，并及时反馈给酒店。此外，平台还要负责对酒店的创建信息的审核。

**酒店信息整合：**平台需要对酒店信息进行整合，并按照类型进行显示，顾客在订房时可通过筛选城市、酒店名、房型，按价格、评价排序，选择心仪的房间预订。

**酒店信息审核：**平台需要对酒店提交的创建信息和各项目变更信息进行审核。在酒店提交创建信息和变更信息后，平台将会显示该条信息，且信息状态为“待审核”。平台管理登录系统，对酒店提交的信息点击“确认”后，信息状态变更为“注册已通过”，同时平台上整合的酒店信息会进行更新。

## 2 系统结构设计

### 2.1 数据模块

本系统采用 Qt 5.10.1 编写，使用 SQLite 数据库实现数据的持续化，在多 App 模式下，数据库承担着多用户之间共享数据的任务。

本系统出于功能应用和界面设计考虑，将数据分为酒店类、房间类、订单类、用户类、平台管理类、酒店注册类。笔者采用 Navicat Premium 12 对 SQLite 数据库实现可视化，切换视图为 ER 图表，显示如下：

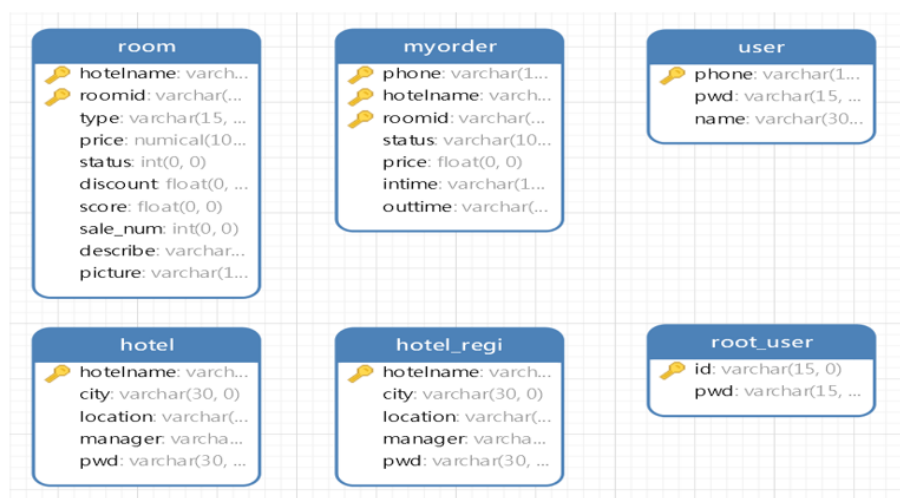


图2-1 数据库hotel的ER图表显示

### 2.1.1 酒店类(包含经理类)

酒店数据一共包含五个成员属性：酒店名称 **hotelname**，所处城市 **city**，所在地区 **location**，酒店经理 **manager**，密码 **pwd**。其中，**hotelname** 为主键，与 **hotel\_regi** 类和 **room** 类联动。酒店类的对象可以是平台上 (**hotel**, **room**) 显示的某一家酒店，也可以是顾客订房 (**myorder**) 显示的酒店。

### 2.1.2 房间类

房间类 (**Room**) 可以用来管理酒店房间的数据：酒店名 **hotelname**、房间号 **roomid**、房型 **type**、价格 **price**、房间状态 **status**、当前折扣 **discount**、评分 **score**、历史销量 **sale\_num**、房间描述 **describe**、房间图片路径 **picture**。其中，**hotelname** 和 **roomid** 为联合主键，与 **hotel** 类和 **hotel\_regi** 类联动。房间状态 **status** 的值有 {0,1}，对应的订单状态分别为 {空闲、有客}。房间类的对象可以是平台 (**room**) 上显示的某一个房间，也可以是顾客订房 (**myorder**) 显示的房间。

### 2.1.3 订单类

订单类 (**myorder**) 可以用来管理当前订单数据：电话号码 **phone**、酒店名 **hotelname**、房间号 **roomid**、价格 **price**、订单状态 **status**、入住时间 **intime**、离店时间 **outtime**。其中，**hotelname**、**phone** 和 **roomid** 为联合主键，与 **hotel** 类、**room** 类、**user** 类、**hotel\_regi** 类联动。房间状态 **status** 的值有 {00,10,01,11,02,12,03,13,04,14}，对应的订单状态分别为 {预订待确认、预订已确认、付款待确认、付款已确认、退款待确认、退款已确认、入住待确认、入住已确认、退房待确认、退房已确认}。

### 2.1.4 酒店注册类

酒店注册类(hotel\_regi)可以用来管理当前酒店注册数据：酒店名称 hotelname，所处城市 city，所在地区 location，酒店经理 manager，密码 pwd。其中，hotelname 为主键，与 hotel\_regi 类和 room 类联动。酒店注册类在酒店经理进行酒店创建时生成记录，在平台管理员审核通过后删除记录。

### 2.1.5 用户类、平台管理类

用户类(user)和平台管理类(root\_user)可以分别用来管理用户和平台管理人员的个人数据：电话号码 phone、用户名 name、平台管理用户名 id、密码 pwd。其中，phone 和 id 分别为用户类和平台管理类的主键，与 myorder 类联动。

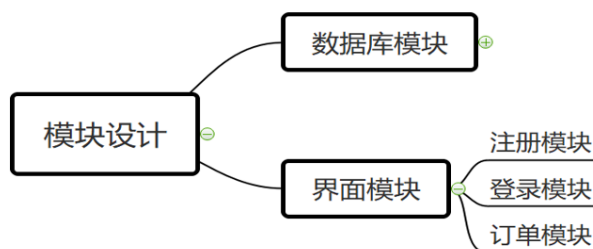
### 2.1.6 其他数据相关类

数据报表导出类(导出为 csv 文件)：封装了对 CSV 的基本操作，用于在酒店经理界面的销售情况子界面导出数据报表。

## 2.2 界面模块

本项目的界面部分主要由四个类构成，即开始界面和三种类型用户的界面。在开始界面的子界面可以进行不同类别用户的登录和注册。三种类型用户的界面下包含各自的子界面，以实现各自的不同功能。

总体模块设计如图 2-2 所示：



## 3 系统详细设计

### 3.1 类结构设计

#### 3.1.1 用户、经理、平台管理类

用户类型一共分为三种，依次为：用户、酒店经理、平台管理。user.h 中定义了静态的 QString 数组，存储每种类型的用户对应的名称。不同类型的用户需要包含的数据不尽相同，如图 3-1

所示。

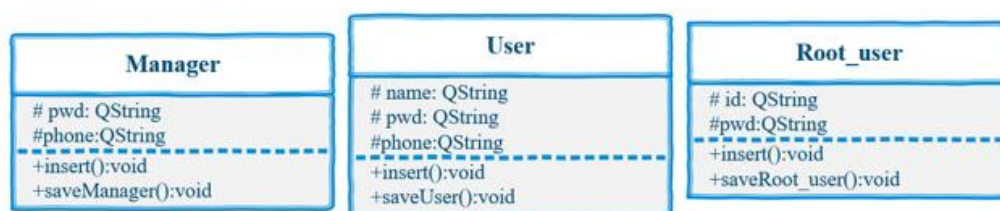


图3-1 用户、经理、平台管理类UML图

### 3.1.2 酒店类

酒店类(Hotel)可以用来管理酒店的数据：名称、城市、地区、电话。酒店类的对象可以是平台上显示的某一家酒店，也可以是顾客订房显示的酒店。

### 3.1.3 房间类

房间类(Room)可以用来管理酒店房间的数据：酒店名、房间号、房型、价格、房间状态、当前折扣、评分、历史销量、房间描述、房间图片路径。房间状态的值有{0,1}，对应的订单状态分别为{空闲、有客}。房间类的对象可以是平台上显示的某一个房间，也可以是顾客订房显示的房间。

### 3.1.4 订单类

订单类(myorder)可以用来管理当前订单数据：电话号码 phone、酒店名 hotelname、房间号 roomid、价格 price、订单状态 status、入住时间 intime、离店时间 outtime。其中，hotelname、phone 和 roomid 为联合主键，与 hotel 类、room 类、user 类、hotel\_regi 类联动。房间状态 status 的值有{00,10,01,11,02,12,03,13,04,14}，对应的订单状态分别为{预订待确认、预订已确认、付款待确认、付款已确认、退款待确认、退款已确认、入住待确认、入住已确认、退房待确认、退房已确认}。

### 3.1.5 注册界面类

- void regi::on\_reg\_clicked(): 输入注册信息，同时判断非空；判断手机号码格式正确性，并与数据库中现有数据比较，检验该手机号是否被注册过；判断二次输入密码的一致性。成功注册后将信息写入数据库，并跳转订房平台。
- void regi::reshow(): 刷新界面，所有框中数据清空。
- void regi::on\_login\_clicked(): 跳转登录界面
- void regi::on\_main\_clicked(): 跳转系统主界面

### 3.1.6 登录界面类

- void login::on\_login\_2\_clicked(): 检验该手机号是否已注册，判断输入密码是否正确，跳转订房平台
- void login::reshow(): 刷新界面，所有框中数据清空
- void login::on\_reg\_clicked(): 跳转注册界面



- void login::on\_main\_clicked(): 跳转系统主界面

### 3.1.7 订房界面类

- void platform::reshow(QString phone): 设置用户名称, 查看空闲房间并以表格显示, 设置表格形式
- void platform::on\_order\_clicked(): 预订生成新订单, 插入数据库, 跳转查看房间
- void platform::on\_query\_clicked(): 查询房间(房型, 价格排序, 高评价优先)
- void platform::on\_pushButton\_clicked(): 跳转系统主界面
- void platform::on\_my\_order\_clicked(): 跳转我的订单

### 3.1.8 查看房间界面类

- void showroom::reshow(QString phone, QString name, QString roomid, float price): 从数据库读取信息并显示
- void showroom::on\_pushButton\_2\_clicked(): 跳转订房平台
- void showroom::on\_pushButton\_clicked(): 设置入住和离店时间, 写入数据库

### 3.1.9 我的订单类

- void myOrder::reshow(QString phone): 设置酒店名称, 查看历史订单并以表格显示, 设置表格形式
- void myOrder::on\_pushButton\_2\_clicked(): 跳转系统主界面
- void myOrder::on\_pushButton\_clicked(): 跳转订房平台
- void myOrder::on\_pushButton\_3\_clicked(): 跳转评分界面
- void myOrder::on\_pay\_clicked(): 跳转付款界面
- void myOrder::on\_money\_back\_clicked(): 退款
- void myOrder::on\_in\_room\_clicked(): 入住
- void myOrder::on\_out\_room\_clicked(): 退房

### 3.1.10 审核界面类

- void hotelConfirm::reshow(): 从数据库读取信息并显示
- void hotelConfirm::on\_pushButton\_clicked(): 从数据库 hotel\_regi 读取, 审核通过后写入数据库 hotel, 并更新状态为“注册已通过”
- void hotelConfirm::on\_pushButton\_2\_clicked(): 跳转系统主界面

### 3.1.11 经理界面类

- void hotelShow::reshow(QString name): 显示酒店名, 统计空闲房间数并输出
- void hotelShow::on\_pushButton\_clicked(): 跳转主界面
- void hotelShow::on\_add\_room\_clicked(): 跳转添加客房
- void hotelShow::on\_manage\_order\_clicked(): 跳转订单管理
- void hotelShow::on\_showSale\_clicked(): 跳转销售情况

### 3.1.12 销售情况界面类

- void saleDetail::reshow(QString id): 从数据库查询所有已完成订单并显示
- void saleDetail::on\_pushButton\_clicked(): 将销售情况导入到 csv 文件中
- void saleDetail::on\_pushButton\_2\_clicked(): 跳转经理界面
- void saleDetail::on\_toolButton\_clicked(): 设置存储路径

### 3.1.13 订单管理界面类

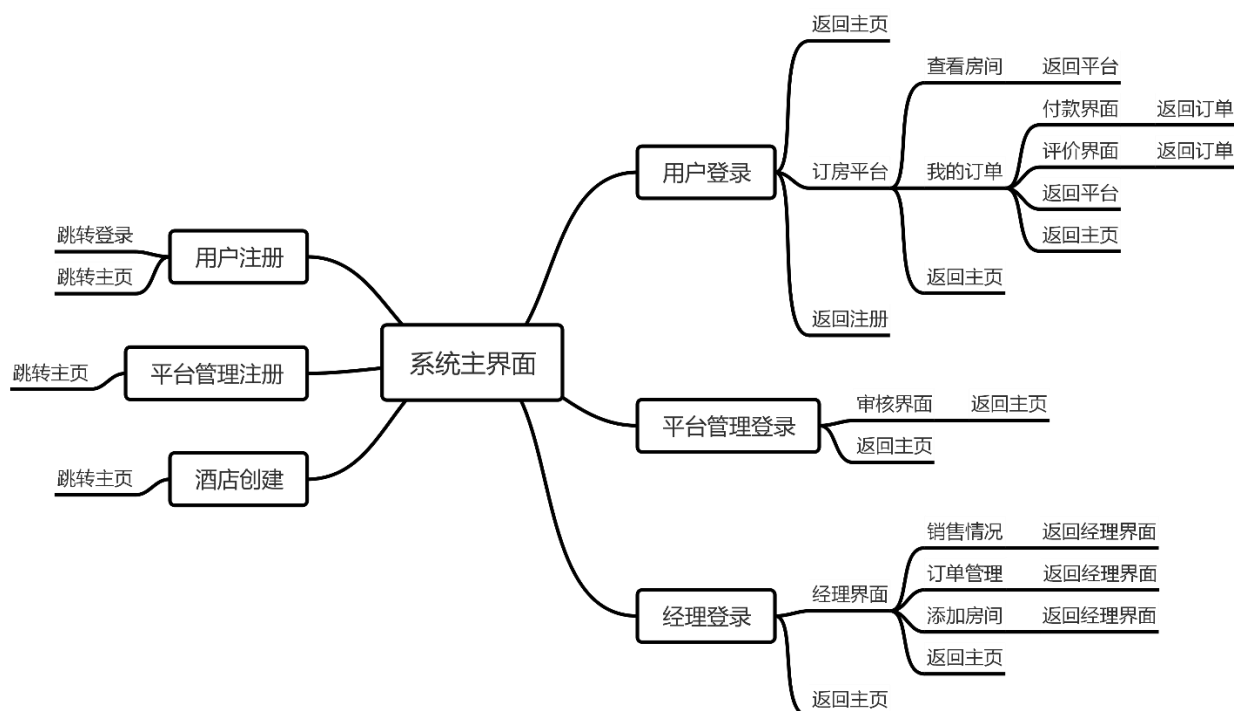
- void orderController::reshow(QString id): 查看所有未完成订单并以表格显示, 设置表格形式, status 数值表订单状态
- void orderController::on\_pushButton\_2\_clicked(): 跳转经理界面
- void orderController::on\_pushButton\_clicked(): 确认订单
- void orderController::on\_pushButton\_3\_clicked(): 确认付款
- void orderController::on\_pushButton\_4\_clicked(): 确认退款
- void orderController::on\_pushButton\_5\_clicked(): 确认入住
- void orderController::on\_pushButton\_6\_clicked(): 确认退房

### 3.1.14 添加客房界面类

- void addRoom::reshow(QString id): 刷新界面
- void addRoom::on\_back\_clicked(): 跳转经理界面
- void addRoom::on\_add\_button\_clicked(): 添加房间, 判断非空, 将房间信息写入数据库
- void addRoom::on\_toolButton\_clicked(): 设置图片路径

## 3.2 界面结构设计

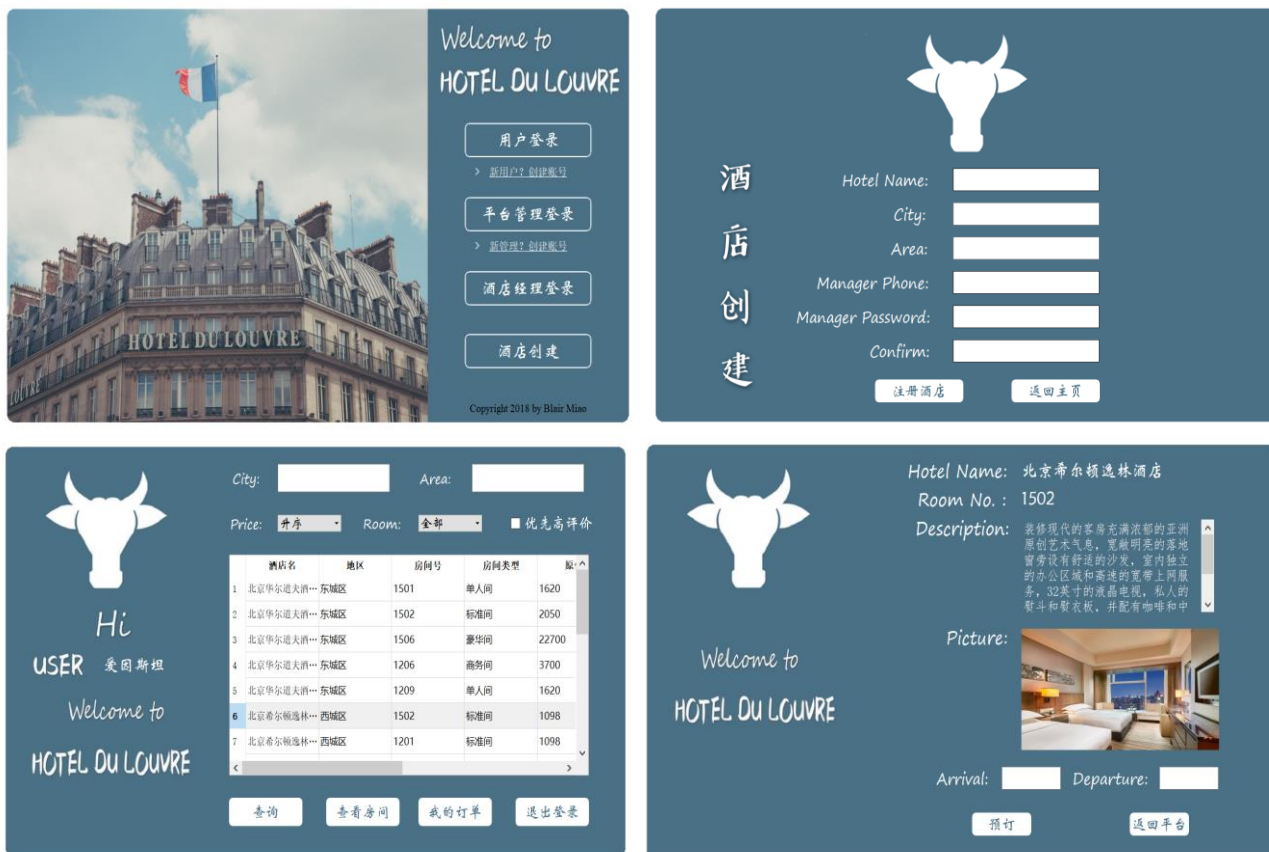
### 3.2.1 界面跳转关系



### 3.2.2 界面风格设计

本系统界面共四类，主界面，用户、平台管理、经理界面，及其各自的子界面。为美化界面，营造极简风格，对界面进行了以下处理：

- 去除各个窗口的边框和标题栏，并进行圆角化处理
- 以合适 icon 图片作为 exe 可执行文件的图标
- 在 QWidget 上另设 QFrame 层，用于放置背景图片，各图片素材来自网站 <https://pixabay.com/>，字体素材来自 <http://xiuxiu.web.meitu.com/>，由笔者自行拼接组合成图
- 共设计了两类按钮，进行了圆角化处理，设置了合适的边界宽度、边界颜色，填充颜色，字体样式及颜色
- 根据输入字数分别采用 LineEdit 和 TextEdit
- 各个平台进入界面都有显示欢迎字样。值得一提的是，用户登录进入订房平台后，左侧欢迎界面会显示用户名称
- 主界面右下角显示 copyright 字样



### 3.3 容错功能设计

由于本项目与用户之间的交互较多，所以要充分设计容错功能。本项目的容错功能主要通过以下手段实现：

#### 检查函数：

当用户输入数据时，实时调用检查函数检测内容是否合法。登录模式下，手机号未注册、手机号格式不正确会给出错误信息；注册模式下，手机号已注册、手机号格式不正确、两次密码不一致会给出错误信息。酒店经理在新建酒店和添加客房时，会判断所有栏目是否非空，

然后再将房间数据写入数据库。

**规则表达式：**当对输入格式有要求时，采用规则表达式严格控制。例如手机号输入框中只能输入首位为 1，且长度不超过 11 位的内容；酒店经理对房间管理时，房间价格只能输入数字。

**禁用按钮：**判断输入非法后，采用禁用相关按钮的方式限制。例如当用户输入两次密码不一致时，提交注册的按钮会被禁用。

## 4 项目总结

### 4.1 项目亮点

#### 4.1.1 界面美观

为了美化界面，提升用户体验，实现不同系统下的风格统一和进一步美化，本系统去除了系统自带的边框和标题栏；利用 Qt Designer 绘制了界面的圆角和阴影等效果；以美观的 icon 图片作为 exe 可执行文件的图标；在 QWidget 上另设 QFrame 层，用于放置背景图片；对按钮进行了圆角化处理，设置了合适的边界宽度、边界颜色，填充颜色，字体样式及颜色；各个平台进入界面都有显示欢迎字样；为实现最佳显示效果，根据输入字数分别采用了 LineEdit 和 TextEdit。

#### 4.1.2 功能完善

本项目实现了多用户的系统，能够做到不同用户之间的实时通信，并具有单 App 和多 App 两种模式，满足不同的需求。各个界面的逻辑、功能都考虑周详，力求完整、合理、便捷。例如，在管理员对菜品和用户的管理时，实现了动态搜索，即搜索结果随输入框中内容变化而变化。实现了搜索过程中的编辑与删除，更加方便了信息的管理。在菜品类别和用户类别的选择中，实现了“全选”框的三态效果。经理界面下，导出 Excel 的设计方便了经理对餐厅的管理。

### 4.2 项目难点

#### ● 问题分析以及类、模块、数据结构的设计

本项目中涉及到很多种类的数据，如何合理有效地组织这些数据是开发过程中的重要问题。通过分析不同数据之间的联系，经过多次的尝试和修改，才确认现有结构。

#### ● Qt Creator 开发工具、Sqlite 数据库的学习与使用

本项目涉及界面设计和数据存储，在学长的建议下，笔者采用了 Qt+Sql 组合的开发模式，由于是全新的工具，在开发前，笔者详细阅读了相关教程，并在开发过程中查询相关问题，请教师长和同学。

## 5 心得体会

### ● 自学能力的培养

本次项目的开发的大部分知识都需要通过自己查找资料来学习。例如 Sqlite 数据库、Qt 等。这些资料大部分来源于网络，需要有效地查阅并筛选资料，并将其合理地应用到自己的项目中。除了官方的教程指南，StackOverFlow、GitHub、CSDN、简书、博客园等开发者社区也是学习交流的很好平台。

### ● 界面的重要性

本项目的基础版本开发时，界面较为简单朴素，在浏览各大顶级酒店和大型订房网站后，笔者决定从界面入手，以当下年轻人的主流——极简风格作为本系统界面的基础风格，由于各个背景图片都是原创拼接组装，以及设置了许多不同的按钮、字体、文本框、标签样式，笔者用了多天时间才完成整体风格改变，在邀请若干同学试用测评后，均对界面设计反映良好。

### ● 抗压能力的提升

笔者今年五月份从化学工程系平转入自动化系进行学习，由于两个院系的课程设置极其不同，在大一年级，笔者并没有学习 C 语言的课程，而本次的 C++ 小学期又十分不幸地安排在了七月中旬，尽管在期末考试结束后开始自学 C 语言，但面对三周时间完成 C 语言学习、C++ 学习、Qt 及数据库学习、OJ 线上小作业、酒店预订与管理系统单人大作业，笔者真正体会到了语言学习“从入门到退学”这句话的真谛。

在进行思路设计的时候，笔者曾多次感到深深的无助感，但知晓无法退课后，面对挂科的威胁，不得不背水一战。直到现在写下这段话，仍不敢相信自己竟在如此恶劣的条件下交出了一个相对满意的系统软件。想起那些爆肝的夜晚，不禁觉得十分欣慰。作为进入清华以来遇到的第一个“不可能完成”的任务，这次的经历想必会成为我人生中一段影响深远的回忆。

P.S. 不妨以笔者期间的票圈吐槽截图作结



缪谨蔚

忍不住发票圈🤔  
舍友回来想问她吃不吃DQ，结果张口成“  
姐姐，Qt嘛🤔🤔”

"How are you?"  
Dolorous, Lachrymose, Lachrymose,  
Melancholy, Gloomy, Mournful, Solemn,  
Disconsolate, Grieved, Whoful, Grieved,  
Dolorous, Lachrymose, Lachrymose,  
Pitiful, Grieved, Mournful, Grieved,  
Grieved, Mournful, Mournful, Grieved,  
Grieved, Mournful, Mournful, Grieved,  
Grieved, Mournful, Mournful, Grieved,  
Grieved, Mournful, Mournful, Grieved.

Fine.

2018年7月28日 17:40 删除

## 6 相关问题的说明

由于篇幅受限，程序开发环境和执行环境的搭建方法及操作实例将在本文档所在文件夹中的附录.pdf 文件中展示。