

设计说明书

魏子卜

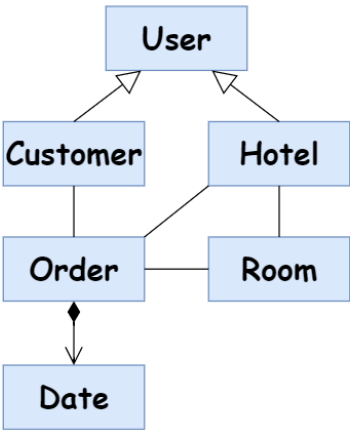
一、 系统结构设计

1. 整体设计

本项目实现了酒店管理系统，具有顾客、酒店、平台管理员三种用户，在本系统上进行订单的处理。三种用户对应三个不同的模块，而每个模块都需要用到三种用户的数据，如用户预订房间需要酒店的数据，因此模块间具有复杂的相互关系。本项目使用 Qt 进行 GUI 编程，对用户非常友好，同时大量利用 Qt 中的模板，尽量减少重复造轮子。本项目充分展现了 C++面向对象的特性，使用了大量面向对象的思想和方法，比如类的封装、继承、多态、模板等，并在部分与用户交互的地方支持文件输入。

2. 类的设计

除去 Qt 界面类，本项目实现了 User 类（用户）、Customer 类（顾客）、Hotel 类（酒店）、Room 类（房间）、Date 类（日期）、Order 类（订单）。



各个类之间的交互主要是通过不同类型的列表来实现的，因此很多类之间（比如 Hotel 和 Room）没有互相包含的关系。下面展开各个类进行详细说明。

3. User 类

User 类是 Customer 类和 Hotel 类的基类，所有使用用户名和密码的用户都会用到 User 类。编写一系列 Set 函数和 Get 函数作为接口，实现了类的封装。

User
-username:QString //用户名
-password:QString //密码
+User() //构造函数
+User(username:QString, password:QString) //构造函数的重载

+SetUsername(username:QString):void	//设置用户名
+SetPassword(password:QString):void	//设置密码
+GetUsername():QString	//获取用户名
+GetPassword():QString	//获取密码

4. Customer 类

Customer 类继承了 User 类，在此基础上顾客类添加了“余额”数据成员，同时建立有关“余额”的接口。如果顾客的余额不足，则无法订单，此时顾客可以充值，继续完成房间的预订。

Customer
-balance:float //余额
+Customer() //构造函数
+SetBalance(setBalance:float):void //设置余额
+GetBalance():float //获取余额
+Recharge(money:float):void //充值

5. Hotel 类

Hotel 类继承了 User 类，在此基础上酒店类添加了一些数据成员：“酒店名称”、“地点”（如海淀区）、“电话”、“是否通过平台审核”、“评分”、“评价个数”、“地址”（如双清路 30 号）。并通过接口实现类的封装。地址是酒店的详细信息，而考虑到顾客预订酒店时会选择更大的范围，因此设置酒店地点，方便顾客进行筛选和查询。

Hotel
-hotelName:QString //酒店名称
-place:QString //地点
-tel:QString //电话
-isPermitted:bool //是否通过审核
-comment:float //评分
-commentNum:int //评价个数
-addr:QString //地址
+Hotel() //构造函数
+Hotel(hotelName:QString, place:QString, tel:QString , isPermitted:bool, comment:float, commentNum:int, addr:QString) //重载构造函数
+SetHotelName(name:QString):void //设置酒店名称
+SetPlace(place:QString):void //设置地点
+SetTel(tel:QString):void //设置电话
+SetisPermitted(isPermitted:bool):void //设置是否通过审核
+SetComment(comment:float):void //设置评分（新注册酒店和测例初始化会用）
+SetCommentNum(commentNum:int):void //设置评价个数

```

+SetAddr(addr:QString):void //设置地址
+GetHotelName():QString //获取酒店名称
+GetPlace():QString //获取地点
+GetTel():QString //获取电话
+GetAddr():QString //获取地址
+GetIsPermitted():bool //获取该酒店是否通过审核
+GetComment():float //获取评分
+GetCommentNum():int //获取评价个数

```

6. Room 类

本项目考虑到方便顾客查询所有满足其要求（地点、价格等）的房间，没有将房间包含在酒店类中，而是在房间类中设置数据成员记录房间所属酒店。Room 类含有丰富的数据成员来表示房间信息，并设计了相应接口。

Room
-belong2Hotel:QString //房间所属酒店的名称 -id:QString //房间号 -type:QString //房间类型（单间、标间等） -description:QString //描述（WiFi、breakfast 等） -price:float //价格（一天的原价） -discount:float //折扣
+Room() //构造函数 +~Room() //析构函数 +SetBelong2Hotel(hotelname:QString):void //设置所属酒店 +SetRoomId(roomid:QString):void //设置房间号 +SetRoomType(roomtype:QString):void //设置房价类型 +SetRoomDescription(roomdescription:QString):void //设置描述 +SetRoomPrice(roomprice:float):void //设置原价 +SetRoomDiscount(roomdiscount:float):void //设置折扣 +GetBelong2Hotel():QString //获取所属酒店 +GetRoomId():QString //获取房间号 +GetRoomType():QString //获取房间类型 +GetRoomDescription():QString //获取描述 +GetOriginalRoomPrice():float //获取原价 +GetRoomDiscount():float //获取折扣 +GetPresentRoomPrice():float //获取打折后的房间价格 +HasDiscount():bool //判断房间是否有折扣

7. Date 类

Date 类仿照教材上的 Date 类创建，有一些基本的接口即为书上原有的函数，这个类编写的比较早，没有使用 Qt 封装好的 QDate 类，自己实现了基本功能，在这一部分可以得到比较不同日期的先后，判断闰年，显示日期，后移日期，并重载赋值运算符，实现对日期的赋值。

Date
-year:int //年份 -month:int //月份 -day:int //日 -totaldays:int //从公元 1 年 1 月 1 日到该日期的总天数
+Date() //构造函数 +Date(year:int,month:int,day:int) //重载构造函数 +~Date() //析构函数 <<const>>+getYear():int //获取年份 <<const>>+getMonth():int //获取月份 <<const>>+getDay():int //获取日 <<const>>+getTotalDays():int //获取总天数 <<const>>+getMaxDay():int //获取某月的最大天数（用于辅助 getTotalDays()） <<const>>+isLeapYear():bool //判断是否是闰年 <<const>>+timeSpan(date:const Date&):int //计算两个日期间隔时间 +AfterDate(d:int):Date //将该日期后移几天（用于平台管理员操作日期变化） +showDate():QString //显示日期（形式：2020-12-24） +operator = (date:const Date&):void //重载赋值运算符，实现日期的赋值

8. Order 类

Order 类是将前面几个不同类结合起来的键类。订单需要大量的数据成员描述，同时编写了大量对应的接口。另外设置四个数据成员，commented 面向顾客，表示顾客是否已经对该订单进行评分，refund 面向顾客，表示顾客是否退单，existence 面向酒店，描述该订单是否过期，neworder 面向酒店，描述该订单是否为新订单。经过考虑，两个面向酒店的数据成员存在的价值不大，但可能对今后本项目的功能拓展有帮助，因此留在这里，没有删去。

Order
<<static>>+idnum:int //静态数据成员，用于记录现有订单数量 -id:int //订单号 -hotelname:QString //订单所属酒店 -roomid:QString //订单的房间号 -roomtype:QString //订单的房间类型 -customerid:QString //订单所属顾客 -totalmoney:float //该订单需要支付的总金额

-timespan:int //订单持续时间 -beginDate:Date //开始日期 -endDate:Date //结束日期 -existence:bool //该订单是否过期，面向酒店 -neworder:bool //该订单是否为新订单，面向酒店 -commented:bool //是否已经评分，面向顾客 -refund:bool //是否退单，面向顾客
+Order() //构造函数 +~Order() //析构函数 +GetOrderId():int //获取订单号 +GetHotelName():QString //获取订单所属酒店 +GetRoomId():QString //获取订单房间号 +GetRoomType():QString //获取订单房间类型 +GetCustomerId():QString //获取订单所属顾客 +GetTotalMoney():float //获取订单总金额 +GetTimeSpan():int //获取订单持续时间 +GetBeginDate():Date //获取开始日期 +GetEndDate():Date //获取结束日期 +SetOrderId(orderid:int):void //设置订单号 +SetHotelName(hname:QString):void //设置订单所属酒店 +SetRoomId(rid:QString):void //设置订单房间号 +SetRoomType(rtype:QString):void //设置订单房间类型 +SetCustomerId(cid:QString):void //设置订单所属顾客 +SetTotalMoney(tmoney:float):void //设置订单总金额 +SetTimeSpan(tspan:int):void //设置订单持续时间（从外部传入持续时间） +SetTimeSpan():void //设置订单持续时间（重载函数，用开始和结束时间计算） +SetBeginDate(bdate:Date):void //设置开始日期 +SetEndDate(edate:Date):void //设置结束日期 +GetExistence():bool //获取订单是否过期 +GetNewOrder():bool //获取订单是否为新订单 +GetCommented():bool //获取是否已经评分 +GetRefund():bool //获取是否退单 +SetExistence(exist:bool):void //设置订单是否过期 +SetNewOrder(norder:bool):void //设置订单是否为新订单 +SetCommented(comment:bool):void //设置是否已经评分 +SetRefund(rfund:bool):void //设置是否退单 +GetOrderInfo():QStringList //获取该订单的全部信息

二、 系统详细设计

1. 全局变量

在“list.h”和“list.cpp”中设置几个全局变量，每个类（包括界面类）能够访问和修改这些关键变量，包括顾客、酒店、房间、订单的列表，今天的日期，以及当前用户的类型（顾客、酒店、平台）和在对应列表中的位置。list.h 需要使用 extern 关键字。

list.h

```
extern QList<Customer> customerList;
extern QList<Hotel> hotelList;
extern QList<Room> roomList;
extern QList<Order> orderList;
extern QList<Hotel> unpermittedhotelList;
extern int currentUserType; //1:customer,2:hotel,3:platform
extern int currentUserRank;
extern Date today;
```

list.cpp

```
#include "list.h"

QList<Customer> customerList;
QList<Hotel> hotelList;
QList<Room> roomList;
QList<Order> orderList;
QList<Hotel> unpermittedhotelList;

int currentUserType; //1:customer,2:hotel,3:platform
int currentUserRank;
Date today;
```

2. 界面类结构设计

本项目的大多数功能都是在界面类中实现的，界面类包含该类的.ui 文件、.h 文件和.cpp 文件。通过界面类中的函数实现用户想要的功能，使用信号和槽实现 GUI 和函数的连接。

下面列出了所有界面类及其.ui 文件，并简单描述其功能。命名非常直观，Ctm 代表顾客，Htl 代表酒店，Pfm 代表平台，Reg 代表注册，Lgn 代表登录，HomPage 代表主页。

```
MainWindow  mainwindow.ui  //欢迎
CtmRegWindow  ctmregwindow.ui  //顾客注册
HtlRegWindow  htlregwindow.ui  //酒店注册
CtmLgnWindow  ctmlgnwindow.ui  //顾客登录
HtlLgnWindow  htllgnwindow.ui  //酒店登录
CtmHomPage  ctmhomepage.ui  //顾客主页
HtlHomPage  htllhomepage.ui  //酒店主页
CtmPerInfoPage  ctmperinfopage.ui  //顾客个人信息
HtlEditInfoPage  htleditinfopage.ui  //酒店编辑酒店信息
```

HtlAddRoomPage htladdroompage.ui //酒店添加房间
HtlCheckRoomPage htlcheckroompage.ui //酒店查看现有房间
CtmCheckOrderPage ctmcheckorderpage.ui //顾客查看现有订单
PfmLgnWindow pfmlgnwindow.ui //平台登录界面
PfmHomePage pfmhhomepage.ui //平台主页
PfmPmtPage pfmpmtpage.ui //平台审批新注册酒店界面
CtmKnowHtlInfoPage ctmknowhtlinfopage.ui //顾客查询酒店信息
HtlCheckOrderPage htlcheckorderpage.ui //酒店查看现有订单

下面挑选几个重点界面类进行介绍和分析。

(1) 顾客主页（CtmHomePage 类）

Today is 2020-12-21

Customer Home Page

Personal Info

Log out

	Hotel	Place	Room	Type	Price	Comment	Description
1	tsinghua	HaiDian	000	standard	100	5	wifi,breakfast
2	tsinghua	HaiDian	001	standard	100	5	wifi,breakfast
3	tsinghua	HaiDian	100	standard	100	5	wifi,breakfast
4	tsinghua	HaiDian	101	standard	100	5	wifi,breakfast
5	zijing	HaiDian	000	single	72	4	wifi
6	zijing	HaiDian	001	single	72	4	wifi
7	zijing	HaiDian	100	single	72	4	wifi
8	zijing	HaiDian	101	single	72	4	wifi
9	gugong	DongCheng	000	standard	999.9	5	breakfast

Hotel Place

Room Type

Begin Date 2020/12/20

End Date 2020/12/30

search room Sort(Comment) Sort(Price) KnowHtlInfo Submit Check My Order

顾客主页为本项目的主要功能界面，继承自 `QDialog` 类，订单的行为都是从这里发起的。界面左侧显示了今天的时间，并提供 `QLineEdit` 和 `QDateEdit` 供用户输入地点、房型、开始结束日期等信息，便于按照需求快速查询房间信息，注意开始时间不能早于今天，地点和房型可以不输入，但不要写错。使用 `QTableView` 以表格形式呈现房间信息，非常直观和友好。

初始化的表格上显示了所有房间，因此必须首先点击“search room”按键，查询可获得房间(available room)，程序对此进行了错误提示处理。在左侧输入需求后，点击“search room”按键，在表格中显示符合要求的所有房间，并可以点击相应按键，进行评分（由高到低）和价格（由低到高）的排序。选中某一房间，点击“KnowHtlInfo”按键，可查看房间所属酒店的全部信息以及酒店照片。选中某一房间，点击“Submit”按键，将会提交该订单，如果余额不够会提醒顾客进行充值。充值等个人信息操作可点击右上方的“Personal Info”按键完成。注意点击“KnowHtlInfo”按键和“Submit”按键前应选中某一房间，程序对此也进行了错误提示处理。

点击“Check My Order”按钮可查询顾客现有订单信息。

```
public:
    explicit CtmHomePage(QWidget *parent = 0);
    ~CtmHomePage();

    void ShowARList();
    void SwapRoom(Room &room1, Room &room2);

private:
    Ui::CtmHomePage *ui;

    QList<Room> availableroomList;

private slots:
    void PerInfo();
    void Logout();
    void KnowHtlInfo();
    void SearchAvailableRoom();
    void CommentSort();
    void PriceSort();
    void SubmitOrder();
    void CheckMyOrder();
```

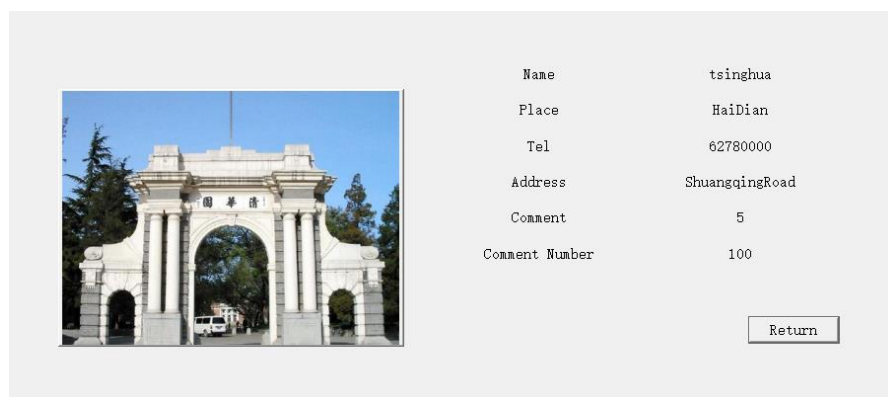
第一个重要函数是 ShowARList(), 该类含有数据成员 QList<Room> availableroomList, ShowARList()将 availableroomList 显示在表格中。

第二个重要函数是 SearchAvailableRoom(), 根据界面左侧的输入信息, 综合 orderList 和 roomList, 查找符合要求的可获得房间(available room)并添加到 availableroomList 中。

另外两个函数 CommentSort()、PriceSort()实现按照评分和价格的排序, 采用冒泡排序, 同时编写函数 SwapRoom(), 在 availableroomList 中交换两个 room 的位置。

第三个重要函数是 SubmitOrder(), 提交选中的订单并收取顾客相应金额, 建立新订单并完善新订单信息, 将新订单加入 orderList。

(2) 顾客查询酒店信息 (CtmKnowHtlInfoPage 类)



在顾客主页点击“KnowHtlInfo”按钮, 显示顾客查询酒店信息界面, 显示酒店的详细信息, 左侧还展示了酒店的照片, 这一部分涉及到文件操作, 将图片文件添加到本项目的.qrc 资源文件中, 以便读取。我采取了一种较为简单的方式显示图像, 在 QFrame 上设置 styleSheet, 将图片显示在 frame 上, 效果不

错。

(3) 顾客查看现有订单（CtmCheckOrderPage 类）

2020-12-21

	Hotel	Place	Room	Type	Description	Begin	End	Money	Comment
1	tsinghua	HaiDian	000	standard	wifi,breakfast	2020-12-20	2020-12-25	500	yes
2	zijing	HaiDian	100	single	wifi	2020-12-21	2020-12-26	360	yes
3	gugong	DongCheng	001	standard	breakfast	2020-12-22	2020-12-27	4999.5	yes
4	qingfen	HaiDian	101	single	wifi	2020-12-23	2020-12-28	480	no
5	tsinghua	HaiDian	101	standard	wifi,breakfast	2020-12-24	2020-12-29	500	yes

no chosen order, the first order will be refunded.

Refund

Comment

Exit

在顾客主页点击“Check My Order”按钮，显示顾客查看现有订单界面，显示今天的日期和现有订单的详细信息。选中想退掉的订单，点击“Refund”按钮，完成退单和退款。选中想评论的订单，点击“Comment”按钮，完成评分。需注意退单必须在订单开始前，评分必须在订单完成后。同样需要注意应首先选中某个订单，再点击按钮，程序都进行了错误提示处理。

Refund()函数对 orderList 进行处理，将订单移除，Comment()函数对酒店进行处理，加权计算酒店评分。

(4) 酒店主页（HtlHomPage 类）

Hotel Home Page

Today is

2020-12-21

Edit Info

Check Room

Check Order

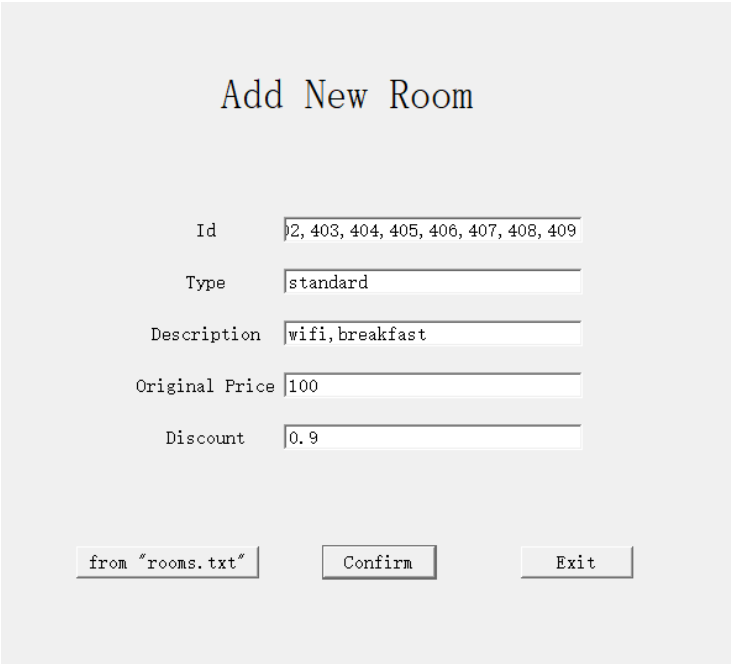
Add Room

Log out

在酒店主页点击“Edit Info”按钮，可以编辑酒店的信息；点击“Check Room”按钮查看本酒店的全部房间信息；点击“Check Order”按钮查看本酒店现有的

全部订单信息；点击“Add Room”可以添加新的房间。

(5) 酒店添加房间（HtlAddRoomPage 类）



The screenshot shows a window titled "Add New Room". It has five input fields: "Id" with a dropdown menu showing "2, 403, 404, 405, 406, 407, 408, 409"; "Type" with a text box containing "standard"; "Description" with a text box containing "wifi, breakfast"; "Original Price" with a text box containing "100"; and "Discount" with a text box containing "0.9". At the bottom, there are three buttons: "from 'rooms.txt'", "Confirm", and "Exit".

在酒店添加房间界面，输入房间的基本信息，需要注意折扣应在 0 到 1 之间。支持一次性输入多个房间号，中间用英文逗号分隔。支持文件输入，标准的房间号已经提前录入.txt 文件。使用 QFileDialog 类的 getOpenFileName 方法得到文件路径，使用 QFile 类的方法打开文件，并用 QByteArray 类的方法读取文件内容，并将文件内容写入 QLineEdit 中。需注意新房间的房间号不得与本酒店已有房间号相同，程序在这里进行了错误提示处理。

(6) 所有的登录及注册界面

在 MainWindow 界面中，将三个 QRadioButton 组成一个 QButtonGroup，点击注册或登录的按键，跳转到不同的注册和登录界面，注册时保证用户名不重复，且密码长度在 6-12 个字符之间，这里使用 QMessageBox 类加入了大量的错误提示信息。

三、 主要技术难点以及实现方案设计

1. 类结构的设计

本项目的设计没有类之间的很强的包含关系，比如 Hotel 不包含 Room 类型数据成员，而是在 Room 中记录 Hotel 的名称，然后通过不同类型的列表将各个类紧密连接。CustomerList、HotelList、RoomList、OrderList 存储了本项目的用户数据。

2. 抽象

本项目对用户这一概念抽象成 User 类，派生出顾客类和酒店类，体现了面向对象的继承特性。在抽象和继承中，充分体现面向对象编程的优势。

3. 模板类的使用

在设计本项目时，还没有学习模板类，本想使用数组，或者学习书上例题手写向量或列表，但是在学习 Qt 的过程中，了解到 QList 综合了向量和列表的优点，同时对模板类的使用也更为深入，于是决定采用 QList，使本项目能够在前人优秀工作的基础上继续进行。同时本项目还使用了 QStringList，即 QList<QString>，通过 QStringList 可获取对象的完整信息。

4. 对房间和订单的查询与筛选

查看可获得房间(available room)的函数 ShowARList()是本项目的一个关键点。通过地点、房型、开始日期、结束日期，遍历房间列表和订单列表，确定某房间是否可以使用。如果在未来进行扩展，进行更快的信息检索，可以采用更为高级的数据结构，如散列，字典等。另外对评分和价格的排序使用了冒泡排序算法，如果数据规模非常大，则考虑使用归并排序，快排等算法。

5. Date 类中 totaldays 的编写

由于闰年，以及每个月天数不同，对日期间隔的处理遇到困难。于是我设计了一个 GetTotalDays()函数，计算今天到公元 1 年 1 月 1 日的总时间间隔，将两个日期的 totaldays 进行比较，即可得到日期间隔，还可以实现日期的后移。

6. GUI 编程

本项目使用 Qt Designer、Qt Creator 进行 GUI 编程，界面非常美观，非常用户友好，明显优于使用控制台输入输出或使用 MFC。学习 Qt 花费了我不少时间，在这个过程中我提升了搜索能力，以及阅读帮助文档的能力，最终做出来效果非常好，尤其是将列表中的数据显示成表格时，第一次感到自己写了一些能给人看的东西。

Qt 中可以拖动控件进行.ui 文件的设计，有些时候不必在.cpp 文件中写大段代码。在最初的 MainWindow 界面的设计中，我还非常不熟悉，在使用 QButtonGroup 时走了不少弯路，最终发现 QButtonGroup 在.ui 中的使用方法，以前犯的错误注释后，留在程序中没有删除。

另外，Qt 的信号和槽是非常独特新颖的概念，点击某个按键即可执行对应的函数，并通过 connect 函数连接。在查阅资料的过程中，我了解到 Java 也具有非常好的 GUI 编程方法，Java 使用“事件”和“监听器”驱动程序的进行。由于 Java 具有比 C++更好更成熟的面向对象特性，在实际工程中，多使用 Java 进行面向对象程序开发，因此在本项目中学习 GUI 编程大有裨益。

7. 容错设计

考虑到用户的实际情况，本项目进行了很多容错设计，对于非法输入会给出提示，不会轻易因为非法输入而崩溃，并引导用户进行正确操作。

以下列出部分错误提示处理：

注册、登录界面：未输入用户名或密码、密码输入错误、注册时两次输入密码不同、密码长度不合要求、用户名已经被注册等。

顾客主页界面：预订日期错误、未检索可获得房间即进行其他操作、未选中房间即

点击“查看酒店信息”或“提交”按键、余额不够需要充值等。

顾客查看订单界面：禁止退掉已开始订单、禁止对未结束订单评分、评分不合要求。

酒店信息编辑界面：编辑的酒店名称已存在。

酒店添加房间界面：添加的房间号已存在。

平台审核酒店界面：未选中通过审核的酒店。

8. 程序运行的问题

本项目在进行到后期时，经常出现奇怪的报错，如下图所示。

```
errors: cannot open output file debug\THotel.exe:Permission denied
collect2: ld returned 1 exit status
```

字面意思就是没有关闭正在运行的程序，而我显然关闭了。我检查程序发现了一个非常简单的错误（没有写函数的内容，或者没有连接信号和槽），修改后依然报这个错。按照网上的方法，通过任务管理器结束掉进程也没有用，退出 Qt 后 rebuild 也没有用，而重启后就可以运行了。我至今不知道为什么出现这种情况，可能是由于 Qt 或编译器的版本太旧了。由于时间紧张就没有卸载，重新安装，但养成了写几行代码就 build 一下的习惯，这也有益于未来做其他大型工程。

9. 代码风格

本项目具有良好的代码风格，文件名、类名、变量名非常清晰明了，使用驼峰式或帕斯卡式风格命名变量，做到见名知意，程序具有良好可读性。

10. 不足之处

本项目最初计划使用数据库保存每次操作产生的数据，因为时间不够就没有再学习 QSQLITE 的使用方法，非常遗憾。

本项目中的时间依靠平台管理员手动调整，希望在以后能够连接网络，自动获取时间，也可以考虑改用 Qt 自带的 QDate 类。

另外，此系统的功能还有很多可以扩展的地方，如平台管理员对酒店进行更为细致严格的审核，如平台管理员删除不合格酒店等。

注：测例在用户手册和 README.txt 中都有写明。