

# “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： 王哲威

学号： 2017011520

班级： 自 76

日期： 2018/8/4

# 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 .....	3
2 系统结构设计.....	4
3 系统详细设计.....	4
3.1 类结构设计 .....	4
3.2 界面结构设计 .....	5
3.3 关键设计思路 .....	5
4 项目总结.....	5
5 相关问题的说明.....	5

# 1 系统功能设计

这是进行复杂软件开发的第一步，即需求分析。此部分需要详细描述清楚系统将要实现的功能。

## 1.1 总体功能描述

描述软件系统总体功能

.....

用户类型	功能名称	功能描述
顾客	查询城市中酒店	给定城市查询其中酒店（并排序）
	查询酒店中房间	给定酒店列出其中房间
	预订房间	预订+付款
	订单操作	申请退款/评价
	用户基本操作	注册/登录/修改密码
酒店	管理房间	管理自己的房间
	管理退款	审核退款请求
管理员	审核酒店	核实新注册的酒店

## 1.2 功能流程描述

用户：注册时 Server 通过相连的手机发送验证码。验证身份后创建用户。用户登录后获得相关功能模块访问授权。

搜索时对 Server 回传信息进行排序。

预定时对 Server 从 `its.tsinghua.edu.cn` 拉取的二维码进行渲染，同时定时访问 `its` 页面确认支付状况，完毕则通知 Server,由 Server 访问 `its` 进行确认。

酒店注册时需要输入大量信息。其中除了少数描述性信息允许修改外，大多数信息为确定的。

登录后获得相关授权。

修改信息时根据 Server 记录的当前用户 `id` 修改对应记录。

修改房间时检查当前用户 `id` 与房间的父节点是否匹配。

管理员登录时通过 `windows` 检验机制判断密码是否可以登录。

## 2 系统结构设计

Sqlite 数据库		
数据处理（与缓存）模块		
Server 功能模块		
Server 通信模块		
Client 通信模块	Client 通信模块	Client 通信模块
Customer 功能模块	Hotel 功能模块	Admin 功能模块

## 3 系统详细设计

### 3.1 类结构设计

Server 包括以下类：

DBManager	数据的访问与缓存。
Password	密码的加盐等操作。
Serverlet	基本的 Tcp 服务器。
Session	对用户请求的应答，也负责处理管理员的请求以及登陆前的各类请求。
User	顾客和宾馆共有的基类。处理所有登陆后的请求。
Customer	顾客类
Hotel	宾馆类

Client 包含以下类：

Welcome	欢迎页面（同时进行网络的连接）
Client	负责所有网络通信
Password	同上
Logindialog	登录/注册页面
Chpwd	修改密码
MainWindow	统筹其它页面/顾客主页面（搜索酒店）
Hotel_Detail	顾客查看酒店的信息以及其中的房间
Book	预订
UserTrades	查看以及处理用户订单
Hotel2	宾馆首页
Rooms	宾馆管理其房间
HotelTrades	宾馆处理退款请求

## 3.2 界面结构设计

顾客/酒店客户端：欢迎页->登录页->主页

顾客主页（酒店搜索）中可以跳转到“密码修改”/“酒店详情”。后者可以跳转到“预定+付款”

酒店主页（基本信息编辑）可以跳转到“密码修改”，“房间信息修改”和“退款申请管理”页面。

管理员：登录页->主页（酒店审核）。

## 3.3 关键设计思路

利用 **Server** 提供最基础的功能以及权限管控。**Client** 进行排序等进一步操作。

使用较为频繁而数量相对较少的信息（酒店的信息与房间信息）在内存中缓存，写入时同时更新内存与数据库。

其它信息仅在数据库中。

每个操作为独立操作。除了身份验证相关操作需要在服务器端寄存数据，其它操作尽可能减少在服务器端使用数据。

## 4 项目总结

最初因重复 `close` 一个 `socket` 发生过系统稳定性的问题。修正后稳定性增强。

`QNetworkRequest` 的使用似乎会破坏 `QTcpServer` 的稳定性。后来改为通过 `QProcess` 带参数调用自身，将二者隔离在不同进程中。

长连接涉及拆包问题。使用 `Json` 作为传输格式自动解决之。

登录时密码有被监听的风险。解决方案为服务器受到请求时给出随机 `salt`，客户端将 `salt` 与 `hash(密码)` 异或后 `hash` 提交，用于验证。

付款二维码是否支付的检查相对比较消耗性能，所以被放在客户端进行，当客户端检查到“已支付”或“订单失效”时向服务器汇报，服务器收到信号后再访问 `api` 接口检查二维码是否支付。

付款码在固定时间后如果没有支付则失效。因此我们将它们放置在一个队列中，用一个定时器轮询检查。每次从队列头找到所有时间已达到的订单。

`Qt Webkit` 不支持 `MinGW`, 使用 `IE` 控件作为替代。

似乎对中文兼容性较差。尚未解决。

## 5 相关问题的说明

编译环境 Qt5.11.0 MinGW

其中需要搭配安卓端时应当使用 adb 命令将手机的 tcp:6000 端口绑定到服务器的 tcp:6000 端口

安卓短信发送器的编写过程中参考了：

[1]<https://examples.javacodegeeks.com/android/core/socket-core/android-socket-example/>

[2]<https://www.jianshu.com/p/a858a9abff70>

程序主体的编写过程中参考了：

[1]<http://doc.qt.io>

[2]<https://docs.microsoft.com/>

授权协议：

根据 Qt 的使用条款，应用 GPL 协议开源。

项目地址：<https://github.com/6ziv/Hotel.git>

问题反馈：<mailto://root@6ziv.com>