

**软件工程报告**

|  |  |
| --- | --- |
| 综合设计题目： | 航班信息管理系统 |
| 姓 名： | 潘翔 |
| 学 院： | 计算机科学与技术学院 |
| 专 业： | 物联网工程 |
| 班 级： | IOT1601 |
| 学 号： | U20161489 |
| 指导教师： |  |

|  |  |
| --- | --- |
| 分数 |  |
| 教师签名 |  |

2018 年 9月 22日

**目 录**

[1 选题 1](#_Toc389907554)

[1.1 教材 1](#_Toc725122700)

[1.2 作业选题背景知识 1](#_Toc1184256243)

[1.3 辅助工具软件 2](#_Toc319784458)

[1.3.1 需求分析 2](#_Toc1823636146)

[1.3.2 工程管理 2](#_Toc673399164)

[1.3.3 软件建模 2](#_Toc801260711)

[1.3.4 版本管理 2](#_Toc835415840)

[1.3.5 自动化测试工具 2](#_Toc1679206848)

[1.4 编程语言和集成调试环境 2](#_Toc1952502629)

[1.5 想法 3](#_Toc973660073)

[2 问题定义及可行性分析 4](#_Toc1946094494)

[2.1 界面操作形式 4](#_Toc1594853499)

[2.2 预期达到功能 4](#_Toc1241992665)

[2.3 使用者角色描述(用户画像) 4](#_Toc1401996478)

[2.4 系统环境描述 4](#_Toc1384931176)

[2.4.1 硬件环境 4](#_Toc1882305442)

[2.4.2 软件环境 4](#_Toc1935387150)

[2.5 系统的性能需求 4](#_Toc317926960)

[2.6 可行性分析 4](#_Toc1405439118)

[3 需求分析 5](#_Toc929176384)

[3.1 功能定义 5](#_Toc1835071620)

[3.2 角色定义 5](#_Toc409306943)

[3.3 设计系统的逻辑关系 5](#_Toc877428085)

[3.4 系统接口 5](#_Toc57889459)

[3.5 系统性能 5](#_Toc1438692310)

[4 概要设计与详细设计 6](#_Toc465062835)

[4.1 系统功能模块划分 6](#_Toc1454575913)

[4.2 系统架构方案 6](#_Toc396168907)

[4.3 系统操作流程 6](#_Toc1891610112)

[4.3.1 系统总体操作流程图 6](#_Toc1515001730)

[4.3.2 系统各部分操作流程图 6](#_Toc786076462)

[4.4 前端实现方案 6](#_Toc469249164)

[4.5 后端实现方案 6](#_Toc551774325)

[4.5.1 系统操作表格图 6](#_Toc1105860920)

[4.5.2 E-R图 6](#_Toc145401662)

[4.5.3 数据流图 6](#_Toc1225173490)

[4.5.4 数据库逻辑结构设计 6](#_Toc1907121632)

[4.5.6 数据表格设计 6](#_Toc980817503)

[4.5.7 数据库事务管理 6](#_Toc756896690)

[4.6 设计模式 6](#_Toc1712140613)

[4.7 注释及命名规范 6](#_Toc1954477576)

[5 测试报告 7](#_Toc555507536)

[5.1 测试环境 7](#_Toc1159510465)

[5.2 功能测试 7](#_Toc1048986593)

[5.3 压力测试 7](#_Toc1957504014)

[6 项目管理 8](#_Toc396957993)

[6.1 项目计划 8](#_Toc783808387)

[6.2 项目实际执行状况 8](#_Toc1745407516)

[6.3 版本迭代情况 8](#_Toc714884953)

[6.3.1 代码迭代情况 8](#_Toc41763857)

[6.3.2 文档迭代情况 8](#_Toc527100252)

[6.4 资源使用情况 8](#_Toc402472925)

[6.5 项目交付情况 8](#_Toc451070800)

[7 课程总结 9](#_Toc1404528338)

[参考文献 10](#_Toc460362384)

[附录 11](#_Toc1889763110)

[文件说明 11](#_Toc1869591173)

# 

# 1 选题

## 教材

教材： 《软件工程(第4版)》

课外阅读：《设计模式：可复用面向对象软件的基础》

## 作业选题背景知识

1. 选题题目：航班信息管理系统
2. 所需知识：
   1. 不同航空公司的航班编号格式
   2. 航班预订流程
   3. 航班退票流程
   4. 航班选座流程
   5. 联程航空的查询及购买
   6. 航班管理系统不同的模块
   7. 航空公司对于用户信息的分析处理
3. 了解途径
   1. 自身航班工具的使用经理
   2. 不同APP和网站的体验使用
   3. 已有航班管理系统的调研
   4. 航班管理系统的相关设计文档和论文
   5. 对于能够接触到的航班工作人员的调查访问
4. 需求分析
   1. 进行用户行为约束， 对于普通用户， 某些操作一旦进行无法更改：
      1. 航班选座
      2. 订单确认
   2. 用户行为的时间约束：
      1. 旅客在飞机起飞前一天凭取票通知交款取票
      2. 用户检票时间约束
   3. 系统查询设计， 完成满足指定要求的查询如航班查询： 进行航班的查询操作
      1. 模糊查询： 利用时间， 地点， 价格等条件查询航班
      2. 精确查询： 利用航班号查询
   4. 航班预订： 进行航班预订的相关操作
      1. 座位选取
      2. 订单生成和确认
      3. 与订单管理模块， 账单管理模块交互
   5. 订单管理： 进行订单的相关操作
      1. 管理订单状态， 已经完成的订单和约束订单无法操作
      2. 更新订单状态
   6. 账单管理： 生成系统的账单

## 辅助工具软件

### 需求分析

思维导图：XMind 8

### 工程管理

MS Project

### 软件建模

MS Visio：UML，数据流图，算法流程图等

### 版本管理

如果是团队集成开发小型采用ant而大型团队使用[cruisecontrol](https://sourceforge.net/projects/cruisecontrol/)，对于单人开发的项目，通常git足够使用

1. 本地：git version 2.18.0
2. 在线：GitHub

### 自动化测试工具

对于QT/C++来说，常见的自动化测试工具为以下，因为不涉及B/S架构，故无须大规模访问的web压力测试，使用数据库，需要对数据库的压力进行一定程度的测试，但由于数据库底层实现的安全机制，故数据安全性和一致性有一定的保障

1. QTest
2. Google Test

## 编程语言和集成调试环境

编程语言： C++

操作系统： ArchLinux x64

IDE: Qt Creator

编译器： 8.1.1 20180531 (GCC)

数据库： 10.1.33-MariaDB MariaDB Server

因 Qt 框架的跨平台特性， 可以发行各平台的发行版本

## 想法

在学习过程中，设计模式是软件工程中很重要的一个部分，尝试使用合适的设计模式对软件进行设计，可以使代码结构清晰，易于维护，并在今后的软件编写过程中尽可能合理使用设计模式和软件架构，进行软件开发和工程管理。

通常之前使用的都是较为简单的工程管理软件，而对于所开发的项目4~6人左右已经足够，但对于软件工程的学习，尝试使用不同的软件工程建模和管理工具。

# 2 问题定义及可行性分析

## 2.1 界面操作形式

## 2.2 预期达到功能

## 2.3 使用者角色描述(用户画像)

## 2.4 系统环境描述

### 2.4.1 硬件环境

### 2.4.2 软件环境

## 2.5 系统的性能需求

## 2.6 可行性分析

阿萨德阿萨德

阿萨德阿萨德  
//技术上来分析该系统的可行性

# 3 需求分析

根据用户的需求，给出系统分析的结果，明确给出系统实现后的功能定义，角色定义等，设计系统的逻辑关系，系统接口以及达到系统性能的措施及解决办法等。采用结构化分析方法时要有系统的详细的数据流图和核心数据的数据字典，必要时进一步画出系统的E-R图和状态转换图。采用面向对象分析方法时画出系统详细的用例模型和对象模型，必要时进一步写出系统的动态脚本或画出系统的事件跟踪图。

## 3.1 功能定义

## 3.2 角色定义

## 3.3 设计系统的逻辑关系

## 3.4 系统接口

## 3.5 系统性能

# 4 概要设计与详细设计

以数据为中心的思想，分析系统的数据结构设计，系统的命名规格定义等内容。

设计出系统的功能模块图或系统结构图。

结构化设计时采用过程模型工具（例如流程图、盒图或PAD图等）设计核心模块内容或算法描述。采用面向对象方法时给出核心类的核心服务的详细内容或算法描述，设计出类之间的关联，指明继承关系。

采用数据库存储数据时给数据库表的设计及表间关联，采用文件存储数据时给出文件的设计结构。

给出系统核心的数据结构。

## 4.1 系统功能模块划分

## 4.2 系统架构方案

## 4.3 系统操作流程

### 4.3.1 系统总体操作流程图

### 4.3.2 系统各部分操作流程图

## 4.4 前端实现方案

## 4.5 后端实现方案

### 4.5.1 系统操作表格图

### 4.5.2 E-R图

### 4.5.3 数据流图

### 4.5.4 数据库逻辑结构设计

### 4.5.6 数据表格设计

### 4.5.7 数据库事务管理

## 4.6 设计模式

## 4.7 注释及命名规范

# 5 测试报告

## 5.1 测试环境

## 5.2 功能测试

黑盒测试

## 5.3 压力测试

白盒测试

# 6 项目管理

自己选择一个模型，估算出系统的工作量（功能点或者源代码行数，应当有依据和过程），自己假定获得了项目的历史生产率数据（FP/PM或L/PM，元/FP,元/L等），估算出项目需要的人数，持续工期和总体成本。

拟定安排的项目人员，同时描述系统的硬件资源和软件资源。

利用甘特图或者工程网络描述项目的进度计划。

## 6.1 项目计划

## 6.2 项目实际执行状况

## 6.3 版本迭代情况

### 6.3.1 代码迭代情况

### 6.3.2 文档迭代情况

## 6.4 资源使用情况

## 6.5 项目交付情况

# 7 课程总结

在学习过程中，设计模式是软件工程中很重要的一个部分，尝试使用合适的设计模式对软件进行设计，可以使代码结构清晰，易于维护，并在今后的软件编写过程中尽可能合理使用设计模式和软件架构，进行软件开发和工程管理。

通常之前使用的都是较为简单的工程管理软件，而对于所开发的项目4~6人左右已经足够，但对于软件工程的学习，尝试使用不同的软件工程建模和管理工具。

# 参考文献

[1] QT官方文档

[2] CSDN[基于Qt的图表库](http://blog.csdn.net/liang19890820/article/details/54600307)

http://blog.csdn.net/liang19890820/article/details/54600307

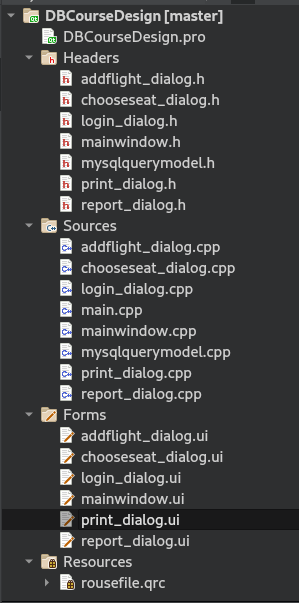
# 附录

## 文件说明

采用驼峰命名法，所有的类名为当前类的功能。

* .h文件为头文件用于函数和信号槽的声明
* .cpp文件用于函数的实现和相关信号槽的绑定
* .ui文件为界面文件

整体架构



整体文件架构图

* addflight\_dialog 添加航班相关事务的处理和界面绘制
* chooseseat\_dialog 添加航班相关事务的处理和界面绘制
* login\_dialog 登录处理和相关权限管理及系统初始化
* Mainwindow 主体窗口绘制和相关响应
* Mysqlquerymodel mysql相关操作的封装和继承重载
* print\_dialog 机票打印和行程单确认相关确认
* report\_dialog 统计报表相关事务的处理和界面绘制
* rousefile.qrc 界面的样式文件及图标文件