**计算机学院**

**课程实验报告**

**课程名称　 敏捷软件开发技术**

**实验名称　 包车在线订票系统**

**专 业**  **软件工程**

**班 级 1402班**

**嘉应学院计算机学院**

**2017 年 12 月**

**团队成员名单**

|  |  |
| --- | --- |
| **成员姓名** | **成员学号** |
| **缪伟宁** | **141110089** |
| **黄展国** | **141110092** |
| **卢如杰** | **141110096** |
| **蔡志雄** | **141110070** |
| **蔡耀威** | **141110071** |
| **余秀庚** | **141110082** |
| **林炽利** | **141110108** |
| **陈杨龙** | **141110113** |

目录

[第一章 方案概述 1](#_Toc503220850)

[第二章 可行性分析 2](#_Toc503220851)

[2.1目的 2](#_Toc503220852)

[2.2背景 2](#_Toc503220853)

[2.3可行性研究的前提 2](#_Toc503220854)

[2.3.1项目要求 2](#_Toc503220855)

[2.3.2项目预期目标 3](#_Toc503220856)

[2.3.3条件、假定和限制 3](#_Toc503220857)

[2.3.4评价尺度 3](#_Toc503220858)

[2.4对现有系统的分析 4](#_Toc503220859)

[2.5包车在线订票系统 4](#_Toc503220860)

[2.5.1包车在线订票系统的说明 4](#_Toc503220861)

[2.5.2处理流程和数据流程 4](#_Toc503220862)

[2.5.3包车在线订票系统使用的影响 6](#_Toc503220863)

[2.5.4技术条件方面的可行性 7](#_Toc503220864)

[2.5.5投资及效益分析 7](#_Toc503220865)

[2.5.6结论 8](#_Toc503220866)

[第三章 项目开发计划 9](#_Toc503220867)

[3.1目的 9](#_Toc503220868)

[3.2背景 9](#_Toc503220869)

[3.3项目概述 9](#_Toc503220870)

[3.3.1项目目标 9](#_Toc503220871)

[3.3.2项目工作内容 9](#_Toc503220872)

[3.3.3应交付成果 10](#_Toc503220873)

[3.3.4项目开发环境 10](#_Toc503220874)

[3.3.5项目验收方式与依据 10](#_Toc503220875)

[3.4项目团队组织 10](#_Toc503220876)

[3.4.1组织结构和人员分工 10](#_Toc503220877)

[3.4.2协作与沟通 11](#_Toc503220878)

[3.4.3项目团队外部沟通与协作模式 11](#_Toc503220879)

[3.5实施计划 11](#_Toc503220880)

[3.5.1风险评估及对策 11](#_Toc503220881)

[3.5.2工作流程 11](#_Toc503220882)

[3.5.3详细工作计划 11](#_Toc503220883)

[3.6总体进度计划 12](#_Toc503220884)

[3.6.1项目启动 12](#_Toc503220885)

[3.6.2系统分析（提供可行性分析报告） 12](#_Toc503220886)

[3.6.3系统设计（完成系统总体及各个模块设计与开发） 12](#_Toc503220887)

[3.6.4系统测试（完成系统个单元及总体测试） 12](#_Toc503220888)

[3.6.5系统实施（完成系统安装，并培训用户） 12](#_Toc503220889)

[第四章 用户故事及需求分析 13](#_Toc503220890)

[4.1用户故事 13](#_Toc503220891)

[4.2需求分析 14](#_Toc503220892)

[4.2.1功能模块 14](#_Toc503220893)

[4.2.2登录 15](#_Toc503220894)

[第五章 开发约定 17](#_Toc503220895)

[第六章 产品backlog 18](#_Toc503220896)

[第七章 系统设计 19](#_Toc503220897)

[7.1目的 19](#_Toc503220898)

[7.2总体设计 19](#_Toc503220899)

[7.2.1需求规定 19](#_Toc503220900)

[7.2.2运行环境 19](#_Toc503220901)

[7.2.3基本设计概念和处理流程 19](#_Toc503220902)

[7.2.4结构 19](#_Toc503220903)

[7.2.5人工处理过程 20](#_Toc503220904)

[7.2.6系统数据结构设计 20](#_Toc503220905)

[7.2.7系统出错设计 21](#_Toc503220906)

[7.2.8程序系统的组织结构 21](#_Toc503220907)

[7.2.9程序设计说明 22](#_Toc503220908)

[7.3详细设计 25](#_Toc503220909)

[7.3.1订单状态 26](#_Toc503220910)

[7.3.2数据库设计实现 26](#_Toc503220911)

[7.3.3前端实现 28](#_Toc503220912)

[7.3.4后端实现 29](#_Toc503220913)

[第八章 Sprint评审 33](#_Toc503220914)

[8.1目标 33](#_Toc503220915)

[8.2会议准备 33](#_Toc503220916)

[8.3会议进程 33](#_Toc503220917)

[8.4会议结果 33](#_Toc503220918)

[第九章 系统测试 34](#_Toc503220919)

[9.1测试计划 34](#_Toc503220920)

[9.2测试用例 38](#_Toc503220921)

[第十章 总结 47](#_Toc503220922)

[10.1 项目小组总结 47](#_Toc503220923)

[10.1.1 目的 47](#_Toc503220924)

[10.1.2 项目背景 47](#_Toc503220925)

[10.1.3 开发结果 47](#_Toc503220926)

[10.1.4 评价 48](#_Toc503220927)

[10.1.5 总结 48](#_Toc503220928)

[10.2 项目看板 49](#_Toc503220929)

[10.3 燃尽图 49](#_Toc503220930)

# 第一章 方案概述

本方案以小组讨论出项目开发需求出发，意在构建一个满足需求并保证安全性的 Web2.0 网站。

出于快速上线的要求，技术团队采用 PHP+MySQL+HTML 进行迅捷开发，使用了国内最流行的基于 MVC 分层机制的轻量级 PHP 开发框架 ThinkPHP 3.1.2。

同时，技术团队实现基于角色的权限控制、重过滤表单输入、用户单客户端在线、即时更改在线管理员权限数据等机制保证网站运营管理的安全性。

为了加快用户访问速度、减轻服务器压力，技术团队采用负载均衡、Nginx 替代 Apache、 脚本缓存、页面静态化等技术手段对网站进行优化处理。

第二章 可行性分析

## 2.1目的

目的是对“包车在线订票系统”的产品设计背景和实施的可行性进行描述。本报告的预期读者是项目的使用单位和项目实施的决策者，以及项目的开发者。

## 2.2背景

学校里学生的期末回家订票是很烦琐的工作，为了提高学校包车人员的工作效率和学生的订票效率，我们准备开发一套包车在线订票系统。现在需要对这个项目的实施进行可行性研究分析。

## 2.3可行性研究的前提

在进行可行性研究之前，先了解以下的前提要求。

### 2.3.1项目要求

1. 功能需求

“包车在线订票系统”的实施，需要实现如下的主要功能。

|  |
| --- |
| **功能** |
| 前台和后台分开管理 |
| 前台展示线路车次列表 |
| 顾客提交订单，订单查询 |
| 后台管理员登录 |
| 系统管理员对人员增删改 |
| 订单管理 |
| 线路车次的增删改 |

表2-1

1. 性能需求
   1. 时间特性需求

在网络连接正常的情况下，响应时间为秒级。

* 1. 灵活性

当需求发生某些变化时，学生管理应用软件操作方式、数据结构、运行环境基本不会发生变化，变化只是将对应的数据库文件内的记录改变，或改变过滤条件。

* 1. 可用性

软件应该尽可能的一目了然，使一般用户能够使用

* 1. 可维护性

当用户使用本套订票系统时，遇到了软件本身的逻辑错误，软件维护人员可以迅速排除错误，解决问题。

* 1. 可兼容性

本系统要在windows XP，Vista，Win7等操作系统都可以直接运行。

1. 项目输出

本套包车在线订票系统需要将所有要浏览的数据显示在屏幕上，以便使用户能够浏览到数据库中的数据或用户想要浏览范围内的数据。

1. 项目输入

用户通过软件输入必要的信息，然后保存到数据库，所输入的信息是经过需求分析限定的内容，同时也是数据库中每个字段存储的内容。

1. 实现方式需求

本套包车在线订票系统软件采用B/S方式实现，顾客可使用浏览器通过互联网进行对订单的提交、查询，管理员可使用浏览器通过互联网进行登录、用户管理、线路管理、订单管理。

1. 在安全与保密方面的要求

本套包车在线订票系统所涉及的数据存放于My SQL数据库中，在程序中应尽可能的使用存储过程的方法，以免使某人反编译软件或入侵到服务器后对数据库的结构进行修改，在程序中应该设置不同权限的账户和密码，以保证数据不容易被错改、破坏，而且要经常对数据库进行备份操作，使得数据一旦受到破坏或是出错能够保证及时的恢复数据，将损失降到最低。

1. 完成期限

本系统从立项到上线需要在一个月内完成。

### 2.3.2项目预期目标

本套包车在线订票系统完成之后，可以实现如下的目标：

1. 减少对包车管理工作的人力与费用；
2. 提高顾客订票的速度；
3. 提高包车信息管理的精度；
4. 促进包车订票工作信息化管理。

### 2.3.3条件、假定和限制

限制必要的输入条件，以排除由于用户的误操作造成不必要的错误。

### 2.3.4评价尺度

* 1. 系统按计划完成
  2. 各项功能满足需求
  3. 产品界面友好，用户容易上手
  4. 项目实施的费用在控制范围之内
  5. 项目需要的各类文档齐全

## 2.4对现有系统的分析

当前对校园包车管理还是通过人工进行，没有使用计算机管理。这造成人力和物力的浪费，还降低执行效率。开发这套包车在线订票系统，就可以把它纳入包车的信息化管理，包车订票可以同时联网操作，可以极大的提高执行效率，而且提高管理的准确性。

## 2.5包车在线订票系统

### 2.5.1包车在线订票系统的说明

本套“包车在线订票系统”软件的开发，是设计出一套顾客在线订票和后台管理的网络管理软件。软件主要是通过网页的形式展示给用户，管理员可以通过后台管理系统进行用户增删改，权限管理和状态管理，线路增删改、状态管理，二级代理可以管理线路车次增删改以及每个车次的订单管理，然后将操作通过互联网保存到服务器端的MySQL数据库中。前台实时跟新数据信息。

### 2.5.2处理流程和数据流程

网站分前台后台

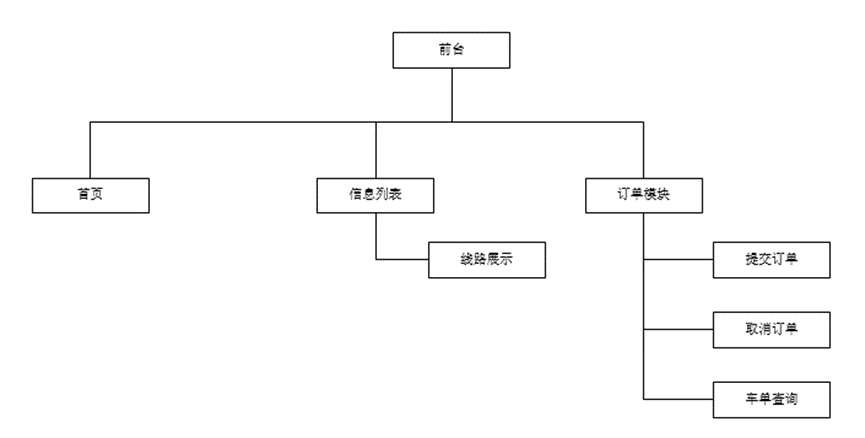


图2-1 网站前台

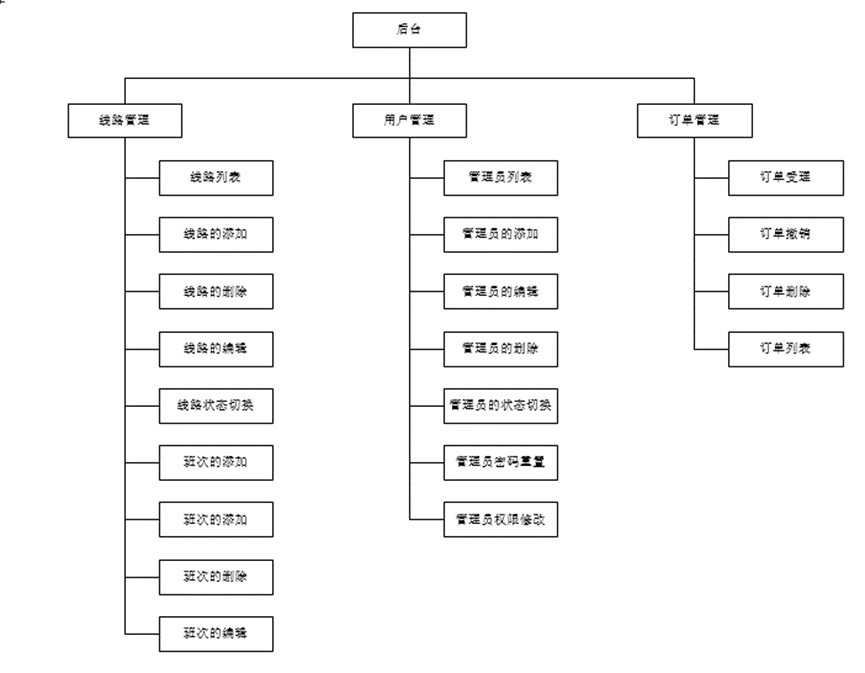


图2-2 网站后台

前台订票数据流程



图2-3 前台订票数据流程

管理员登录



图2-4 管理员登陆流程

### 2.5.3包车在线订票系统使用的影响

1. 对设备的影响

客户端的运行环境：Windows XP或以上的操作系统，IE6以上的浏览器（或其他浏览器）。

服务器的运行环境：操作系统为Windows2000 Server或以上环境。

数据库：MySQL。

应用服务器：Apache

1. 对软件的影响

无

1. 对用户单位机构的影响

建立和运行包车在线订票系统，需要服务器和客户端的局域网可以正常运行；对系统进行操作的管理员和顾客，需要熟悉计算机的基础应用，简单的网络管理和数据库操作知识。

1. 对系统运行过程的影响
   * + - 1. 用户必须按“包车在线订票系统使用说明书”进行规范操作；
         2. 承载数据库和软件的服务器必须性能可靠；
         3. 服务器的数据库有备份机制，以便数据库出现故障，可以进行数据恢复；
         4. 包车在线订票系统软件采用B/S方式实现，所以网络运行环境要稳定可靠。
2. 对开发的影响
3. 包车在线订票系统需要使用PHP开发；
4. 数据库为MySQL；
5. 应该服务器是基于Apache
6. 在开发过程中搭建模拟环境进行调试和测试；
7. 系统运行性能要可靠，数据库要做安全处理。
8. 对地点和设施的影响

安装数据库和应用服务器的机器要放置在云；用户终端可以是单个联网的PC机。

1. 对经费开支的影响
2. 项目需求和设计的研究费用；
3. 开发人员费用支出；
4. 测试人员费用支出；
5. 项目检查和技术管理费用；
6. 项目维护和升级费用；

### 2.5.4技术条件方面的可行性

1. 在当前的限制条件下，包车在线订票系统的功能目标可以达到；
2. 利用现有PHP技术，可以实现所需的功能；
3. 公司的开发人员精通PHP技术及MySQL数据库，而且编写代码质量很高，完全可以胜任这个项目的开发；
4. 在规定的期限内，本系统可以开发完成。

### 2.5.5投资及效益分析

1. 支出
2. 基本投资

安全与保密设备费用；

计算机操作系统和应用软件的费用；

数据库管理软件。

1. 其他一次性支出

项目需求和设计的研究费用；

开发人员费用支出；

测试人员费用支出；

项目检查和技术管理费用；

项目维护和升级费用；

1. 非一次性支出

设备的租金和维护费用；

软件维护及升级费用；

1. 收益
2. 一次性收益

包车在线订票系统的应用可以缩减管理工作的人员数量，节省开支；

提升包车管理和订票、订单管理的质量和效率；

1. 非一次性收益

包车在线订票系统只需要一次性的投入，就可以重复利用。每年都可以节省费用支出。

1. 不可定量的收益

包车在线订票系统的应用可以改变以往的人工管理模式，把包车管理工作进一步推进信息化管理的校园道路。

1. 收益／投资比

包车在线订票系统整个生命期的收益／投资比值。

1. 投资回收周期

包车在线订票系统上线后收益的累计数开始超过支出的累计数的时间。

### 2.5.6结论

通过上述项目可行性分析可以得知，包车在线订票系统的开发，在技术上成熟，经济上合算，同时可以提升包车管理工作的效率和质量，所以可以立即开始进行。

# 第三章 项目开发计划

## 3.1目的

编写这份项目计划的目的。使项目成员和相关人员了解项目开发计划书的作用、以及希望达到的效果。计划的作用是“项目成员以及项目相关人员之间的共识与约定，项目生命周期所有活动的行动基础，便于项目团队根据本计划开展和检查项目工作。”

## 3.2背景

* 1. 项目的名称：包车在线订票系统。
  2. 项目的委托单位：嘉应大学。
  3. 项目的用户（单位）：大学生。
  4. 项目的主要承担部门：开发部。
  5. 项目建设背景：“包车在线订票系统”软件主要是通过网页的形式展示给用户，订票者可以在订票系统上订购所需的车票，管理者可以在后台发布路程和车票，然后将数据通过互联网存储在服务器端的MySQL数据库中。

## 3.3项目概述

### 3.3.1项目目标

包车订票系统以计算机为管理工具，通过对用户订票的信息管理，把管理人员从繁琐的数据计算处理中解脱出来，使其有更多的精力从事更多的工作。

包车订票系统是为了实现订票者能不出门，在家就完成整个购票流程，提升产品的用户体验，订票者通过互联网访问主页，这主页中可以进行班次的查询，车票查询和购票信息的查询，管理者在后台发布订票的相关信息。

### 3.3.2项目工作内容

产品范围界定：对包车订票系统的订票进行处理。

工作范围界定：

* 添加管理用户信息
* 删除管理用户信息
* 修改管理用户信息
* 添加路线
* 删除路线
* 修改路线
* 修改登陆密码
* 角色管理

### 3.3.3应交付成果

1. 需完成的软件

源程序和数据库导出的sql文件。

1. 需提交用户的文档

需求规格说明书、帮助手册等。

### 3.3.4项目开发环境

操作系统 win7

编译环境 Thinkphp

服务器 Apache

集成开发环境 PHP

数据库 MySQL

### 3.3.5项目验收方式与依据

该项目采用交付后验收。

项目验收依据主要有标书、合同、相关标准、项目文档（最主要是需求规格说明书）。

## 3.4项目团队组织

### 3.4.1组织结构和人员分工

项目的成员以及工作分配：

|  |  |
| --- | --- |
| **负责人** | **工作内容** |
| 余秀庚、卢如杰 | 页面设计、数据库部署 |
| 蔡耀威、黄展国 | 可行性分析、需求分析 |
| 林炽利、陈杨龙 | 测试计划，网络搭建 |
| 缪伟宁、蔡志雄 | 后台设计、模块设计、项目部署 |

表3-1 成员工作分配

### 3.4.2协作与沟通

一天一小会，三天一大会。经常协商讨论该项目的进度，主要讨论做什么，怎么做，怎么做的更好。代码就是在不断的重构之中。

我们的沟通方式主要是在会议室之中，针对每个人的想法和意见选出最好的解决方案。如此重复，直至项目的完成。

### 3.4.3项目团队外部沟通与协作模式

项目团队：

与外部的客户的沟通。主要通过面对面的交谈和微信交流明确客户的需求，再针对客户需求进行可行性分析，再与开发人员进行交流讨论。

## 3.5实施计划

### 3.5.1风险评估及对策

1. 工程／规模／进度上的风险

规模大，规模估算不精确甚至误差很大；就规模而言，用户要求交付期、费用很紧；预料外的工作(测试未完时的现场对应等)；

1. 技术上的风险

使用新的开发技术、新设备等，或是新的应用组合，没有经验；是新的行业或业务，没有经验；性能上的要求很严；

1. 用户体制上的问题

用户管理不严，恐怕功能决定、验收不能顺利地完成（或者出现了延迟）；或者恐怕功能会多次变更；与用户分担开发，恐怕工程会拖延（或者出现了延迟）；用户或其它相关单位承担的工作有可能延误；

1. 其它。

### 3.5.2工作流程

该项目采用如敏捷工作流程来进行。

### 3.5.3详细工作计划

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 任务名称 | 工期 | 开始时间 | 完成时间 | 前置任务 |
| **准备阶段** | **4 个工作日** | **2017年12月1日** | **2017年12月4日** |  |
| 调查与可行性分析 | 2 个工作日 | 2017年12月1日 | 2017年12月2日 |  |
| 小组会议 | 1 个工作日 | 2017年12月3日 | 2017年12月3日 | 2 |
| 需求分析与成本估算 | 1 个工作日 | 2017年12月4日 | 2017年12月4日 | 3 |
| **设计阶段** | **5 个工作日** | **2017年12月5日** | **2017年12月9日** | **1** |
| 建立业务模型 | 3 个工作日 | 2017年12月5日 | 2017年12月7日 |  |
| 系统设计 | 3 个工作日 | 2017年12月5日 | 2017年12月7日 |  |
| 数据库设计 | 2 个工作日 | 2017年12月8日 | 2017年12月9日 | 7,6 |
| **实现阶段** | **15 个工作日** | **2017年12月10日** | **2017年12月27日** | **5** |
| 数据库实现 | 2 个工作日 | 2017年12月10日 | 2017年12月11日 |  |
| 前端实现 | 5 个工作日 | 2017年12月12日 | 2017年12月16日 | 10 |
| 后端实现 | 13 个工作日 | 2017年12月12日 | 2017年12月27日 | 10 |
| **测试阶段** | **3 个工作日?** | **2017年12月28日** | **2017年12月30日** | **9** |
| 单元测试 | 2 个工作日 | 2017年12月28日 | 2017年12月29日 |  |
| 功能测试 | 1 个工作日? | 2017年12月30日 | 2017年12月30日 | 14 |
| 完善优化 | 7 个工作日 | 2017年12月31日 | 2018年1月6日 | 13 |
| 部署运行 | 1 个工作日? | 2018年1月7日 | 2018年1月7日 | 16 |
| 小组会议总结 | 1 个工作日? | 2018年1月8日 | 2018年1月8日 | 17 |

## 表3-2 详细工作计划表

## 3.6总体进度计划

### 3.6.1项目启动

### 3.6.2系统分析（提供可行性分析报告）

### 3.6.3系统设计（完成系统总体及各个模块设计与开发）

### 3.6.4系统测试（完成系统个单元及总体测试）

### 3.6.5系统实施（完成系统安装，并培训用户）

# 第四章 用户故事及需求分析

4.1用户故事

1. 作为系统管理员，我希望该系统能够分前后台，前台可观简洁，以便顾客使用系统
2. 我希望前台可以展示后台添加的线路出车班次，以便顾客选择。
3. 我希望前台顾客使用无需注册，选择线路班次填写订单相关信息就能提交订单，减少操作。
4. 作为系统管理员，我希望可以登录系统，以便使用系统
5. 我希望能够验证登录信息，以防非本人操作系统确保安全性。
6. 我希望如非系统管理员可以直接返回前台站点，以便顾客能快速的查看信息进行订票。
7. 作为系统管理员，我希望能够增删改系统用户，以便管理系统用户
8. 我希望可以增加系统用户，以便线路代理可以使用系统。
9. 我希望可以删除用户信息，以便系统被非法利用。
10. 我希望可以修改用户信息，以便管理用户的代理线路和权限。
11. 我希望可以修改用户账号状态，以便以后使用，避免反复添加。
12. 作为系统管理员，我希望拥有其他用户的所以权限，以便监控系统的工作状态
13. 作为系统管理员，我希望能够增删改线路和管理线路状态，以便对前台信息进行快速管理维护
14. 我希望可以添加线路信息，以便对包车业务的扩展。
15. 我希望能够删除线路信息，以免无用线路信息对顾客造成干扰。
16. 我希望能够修改线路信息，以免错误信息误导顾客。
17. 我希望能够更改线路状态，希望以后可以使用不用重复添加。
18. 作为线路二级代理，我希望能够维护线路的车次，以便顾客的订票车次选择
19. 我希望能够添加车次信息，以便客户能对该车次更好的了解。
20. 我希望能够删除车次信息，避免无用车次供顾客选择。
21. 我希望能够修改车次信息，能够给顾客更好的了解。
22. 我希望能够对车次的订单进行管理，以便能够查看该车次的订单情况。
23. 作为线路二级代理，我希望能够在后台能快速访问前台界面，以便查看车次情况。
24. 作为顾客，我希望能够查询车票信息，以便查看订单情况

4.2需求分析

### 4.2.1功能模块

网站分为前台、后台两个部分。前台供顾客选路线、提交、查询、撤销订单，

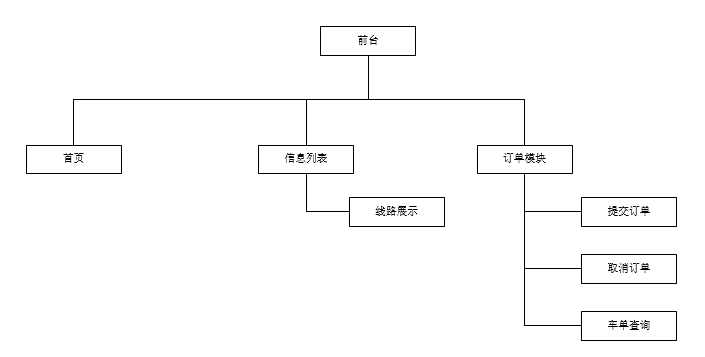


图4-1 前台功能模块

后台供管理人员更新线路和车次信息、处理用户订单及对网站运行做调整，

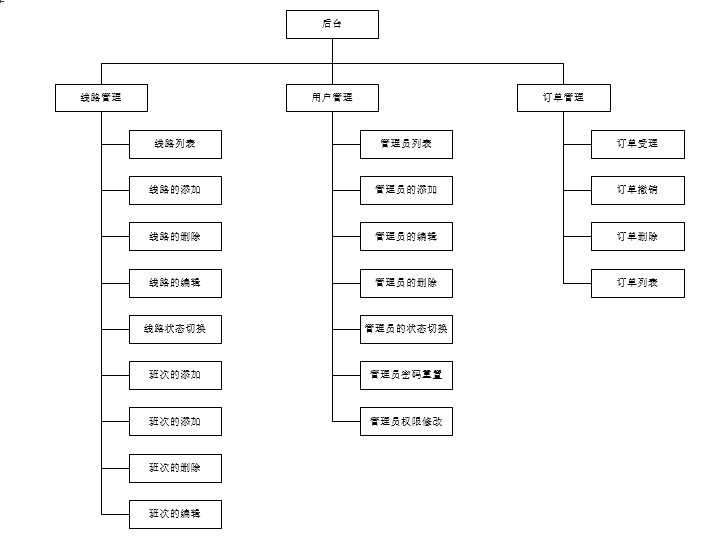


图4-2 后台功能模块

### 4.2.2登录

网站仅有后台管理员登录，前台不需注册顾客即可填写信息提交订单，并根据单号查询、取消订单。 管理员权限通过基于角色的权限控制机制个性化定制不同角色的权限。



图4-3 管理员登陆流程

# 第五章 开发约定

* ThinkPHP 控制层、模型层类文件都是以.class.php 为后缀，使用驼峰法命名，并且首字母大写，例如 DbMysql.class.php；
* 确保文件的命名和调用大小写一致；
* 类名和文件名一致（包括上面说的大小写一致），例如 UserAction 类的文件命名是 UserAction.class.php， InfoModel 类的文件名是 InfoModel.class.php；
* 函数、配置文件等其他类库文件之外的一般是以.php 为后缀；
* 函数的命名使用小写字母和下划线的方式，例如 get\_client\_ip；
* 方 法 的 命 名 使 用 驼 峰 法 ， 并 且 首 字 母 小 写 或 者 使 用 下 划 线 “\_” ， 例 如getUserName，\_parseType，通常下划线开头的方法属于私有方法；
* 属性的命名使用驼峰法，并且首字母小写或者使用下划线“\_”，例如 tableName、\_instance，通常下划线开头的属性属于私有属性；
* 以双下划线“\_\_”打头的函数或方法作为魔法方法，例如 \_\_call 和 \_\_autoload；
* 常量以大写字母和下划线命名，例如 HAS\_ONE 和 MANY\_TO\_MANY；
* 配置参数以大写字母和下划线命名，例如 HTML\_CACHE\_ON；
* 语言变量以大写字母和下划线命名，例如 MY\_LANG，以下划线打头的语言变量通常用于系统语言变量，例如 \_CLASS\_NOT\_EXIST\_；
* 变量以小写字母和下划线命名，如 example\_var；
* 模板文件默认是以.html 为后缀；
* 数据表和字段采用小写加下划线方式命名，并注意字段名不要以下划线开头，例如think\_user 表和 user\_name 字段。

第六章 产品backlog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Name | Important | EST | How to demo |
| 1 | 顾客订票 | 10 | 4 | 顾客进入网站，线路车次列表，选择车次，填写订单信息，确认提交。 |
| 2 | 订单查询 | 10 | 3 | 顾客根据提交订单是产生的订单号，进行查询，查看订单信息 |
| 3 | 系统管理员登录 | 30 | 15 | 管理员进入后台管理登录页面，登录账户，对账号进行验证 |
| 4 | 系统管理员对系统用户进行管理 | 20 | 10 | 系统管理员可以增删改系统用户，账户状态管理，权限管理 |
| 5 | 系统管理员对线路的管理 | 20 | 10 | 系统管理员可以增删改线路，给线路分配代理人，给对应线路进行车次的增删改 |
| 6 | 系统管理员对订单的管理 | 20 | 10 | 系统管理员能够对所有的线路和车次的订单进行受理，取消和删除 |
| 7 | 线路二级代理对订单的管理 | 10 | 8 | 线路二级代理能够对自己代理的线路的车次的订单进行受理，取消和删除 |

表6-1 产品backlog

# 第七章 系统设计

7.1目的

该设计技术说明是为了方便开发人员和用户使用学籍管理系统软件，阐明如何使用学籍管理系统，向用户解释该软件的作用或在必要时作为参考。

预期的读者：开发小组的开发人员、用户。

## 7.2总体设计

### 7.2.1需求规定

本套包车在线订票系统软件采用B/S方式实现，顾客可使用浏览器通过互联网进行订票，和车票查询，系统管理员可使用浏览器通过互联网进行线路、人员、订单的管理。

### 7.2.2运行环境

客户端的运行环境：Windows XP以上的操作系统、IE6以上的浏览器（或其他浏览器）。

服务器数据库端的运行环境：操作系统为Windows2000 Server或以上环境，数据库为MySQL。

应用服务器：Apache。

### 7.2.3基本设计概念和处理流程

本套“包车在线订票系统”软件的开发，是设计出一套顾客在线订票和后台管理的网络管理软件。软件主要是通过网页的形式展示给用户，管理员可以通过后台管理系统进行用户增删改，权限管理和状态管理，线路增删改、状态管理，二级代理可以管理线路车次增删改以及每个车次的订单管理，然后将操作通过互联网保存到服务器端的MySQL数据库中。前台实时跟新数据信息。

### 7.2.4结构

本套系统软件采用Browser/Server方式实现。

### 7.2.5人工处理过程

1. 数据库人工处理

数据库不进行人工处理，一切行为通过客户端维护；

1. 客户端的人工处理
2. 顾客订票：顾客根据情况选择线路车次，购买车票填写相关信息如：姓名、电话、车票数、上下车地点等；
3. 查询车票：顾客根据购票回馈的订单号进行查询；
4. 管理员添加用户：管理员添加系统用户的相关信息
5. 管理员删除用户：管理员删除系统用户的相关信息
6. 管理员修改用户：管理员修改系统用户的相关信息
7. 管理员修改用户权限：管理员修改系统权限，不同权限具有不同功能
8. 管理员分配线路代理：管理员通过下拉框给线路分配代理人。
9. 代理人管理线路：代理人能对线路进行班次增删改。
10. 代理人管理订单：代理人对班次的订单进行受理、取消和删除操作
11. 系统管理员拥有代理人的所以功能和权限。

### 7.2.6系统数据结构设计

1. 逻辑结构设计要点

按照需求分析设计数据库中的字段，建立一个逻辑上的数据库的结构。

1. 物理结构设计要点

在数据库软件(My SQL)中建立数据库，并要保证数据库最低要符合第二范式。

1. 数据结构与程序的关系
2. 静态数值需求
   * + 1. 支持并行操作的用户。
       2. 处理多条记录数据。
       3. 表或文件最小为2048字节,最大无限制。
3. 精度需求

在进行提取数据库数据时，要求数据记录定位准确，在向数据库中添加数据时，要求输入数据准确。主要的精度适应系统要求，不接受违规操作。

1. 时间特性需求
2. 响应时间应在人的感觉和视觉事件范围内；
3. 更新处理时间，随着应用软件的版本升级，以及网络的定期维护更新。
4. 灵活性

当需求发生某些变化时，管理应用软件操作方式、数据结构、运行环境基本不会发生变化，变化只是将对应的数据库文件内的记录改变，或将过滤条件改变即可。

1. 数据管理能力需求

本应用软件可管理多条记录，本应用软件基本约用1,300千字节空间，所有文件均放置在数据库中调用，查询数据、文件、记录时，通过库文件名直接进行操作或通过存储过程来完成操作。

### 7.2.7系统出错设计

1. 出错信息

在设计本套包车在线订票系统软件时，设计人员应尽可能的考虑到所有的出错情况，并做出相应的恢复信息。无法预料的错误信息，应返回给用户一个特定的信息提示。

1. 补错措施

对于出错概率较大的地方，设计人员应结合用户需求做一些必要的限制，减少出错的可能。

1. 系统维护设计

本套包车在线订票系统软件的维护设计要有专门人员来做，这些人员应对本套学籍管理系统软件的程序代码结构与流程有深入的了解。

### 7.2.8程序系统的组织结构

模块层次图

前台：

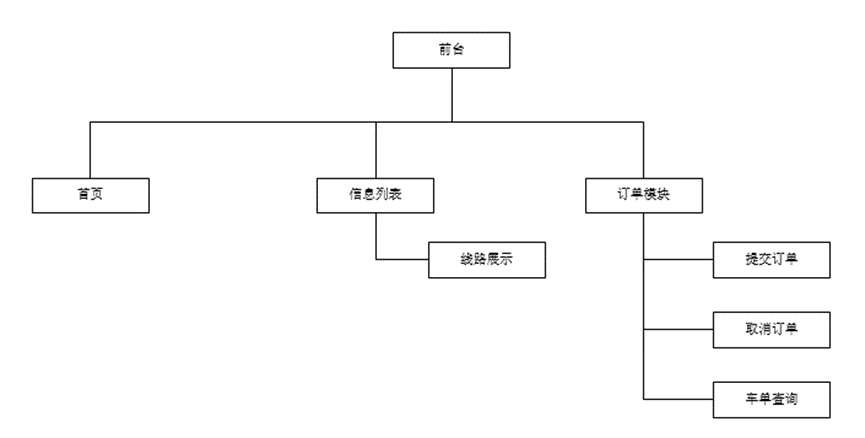


图7-1 前台模块层次

后台：

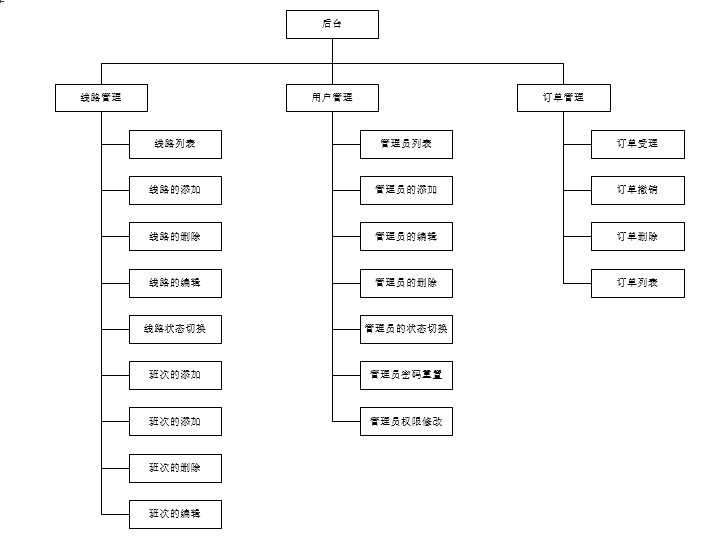


图7-2 后台模块层次

### 7.2.9程序设计说明

1. 程序描述
2. 客户端程序

本套学籍管理包车在线订票系统软件的客户端应用程序以网页的形式编写，包括静态和动态的网页，存放在服务器中，客户使用浏览器通过互连网络对网页进行访问，并完成客户端可以完成的功能。

1. 后台数据库

本套包车在线订票系统软件的后台数据库使用My SQL来搭建后台数据库服务器，用来存放所有的数据。

1. 功能描述

a) 顾客订票：顾客根据情况选择线路车次，购买车票填写相关信息如：姓名、电话、车票数、上下车地点等；

b) 查询车票：顾客根据购票回馈的订单号进行查询；

c) 管理员添加用户：管理员添加系统用户的相关信息

d) 管理员删除用户：管理员删除系统用户的相关信息

e) 管理员修改用户：管理员修改系统用户的相关信息

f) 管理员修改用户权限：管理员修改系统权限，不同权限具有不同功能

g) 管理员分配线路代理：管理员通过下拉框给线路分配代理人。

h) 代理人管理线路：代理人能对线路进行班次增删改。

i) 代理人管理订单：代理人对班次的订单进行受理、取消和删除操作

j) 系统管理员拥有代理人的所以功能和权限。

1. 性能描述

a) 时间特性需求

在网络连接正常的情况下，响应时间为秒级。

b) 灵活性

当需求发生某些变化时，学生管理应用软件操作方式、数据结构、运行环境基本不会发生变化，变化只是将对应的数据库文件内的记录改变，或改变过滤条件。

c) 可用性

软件应该尽可能的一目了然，使一般用户能够使用

d) 可维护性

当用户使用本套订票系统时，遇到了软件本身的逻辑错误，软件维护人员可以迅速排除错误，解决问题。

e) 可兼容性

本系统要在windows XP，Vista，Win7等操作系统都可以直接运行。

1. 项目输入

用户通过软件输入必要的信息，然后保存到数据库，所输入的信息是经过需求分析限定的内容，同时也是数据库中每个字段存储的内容。

1. 项目输入

用户通过软件输入必要的信息，然后保存到数据库，所输入的信息是经过需求分析限定的内容，同时也是数据库中每个字段存储的内容。

1. 算法
2. 将用户输入的数据，按字段保存到数据库中。
3. 将数据库中的数据，按字段提取到用户界面中。
4. 必要的去除重复项的算法。
5. 按条件修改、删除数据中的数据。
6. 保持表间数据的一致性。
7. 流程逻辑

登录：



图7-3 登陆流程

订票：



图7-4 订票流程

1. 注释设计

尽可能的将软件中插入注释语句，使语句容易阅读。制作网页的时候应该制作两分，一份是标有注释语句的网页，用来给维护人员、测试人员和开发人员了解开发过程所用，另一份是不带有注释语句的网页，用来最后实际应用当中，这样可以充分的利用有限的带宽，降低客户的计算机打开网页的时间，提高客户端的浏览速度。

1. 限制条件

限制必要的条件，以排除由于用户的误操作造成不必要的错误。

1. 测试计划

在开发工程师编写代码时，测试人员便要开始制定测试计划，其中要包括白盒和黑盒的具体测试项目，及其必要的测试数据和出错的信息。每次测试的结果要写报告，并就发现和怀疑的问题与开发工程师联系。测试的结果要让开发工程师明白。

7.3详细设计

### 7.3.1订单状态

订单状态是由数据表oms\_order的status字段标识的，每种标识对应的订单状态

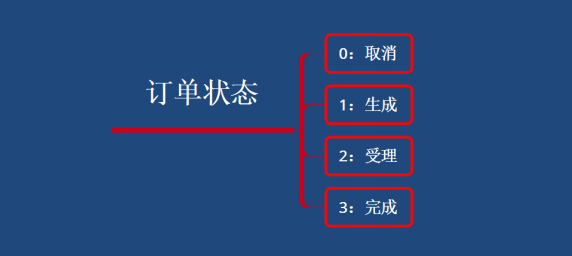


图7-5 订单状态

已完成的订单无法再取消，已生成的订单不可直接设置未完成，已受理的订单不能退回生 成状态，已取消的订单无法完成。

### 7.3.2数据库设计实现

该系统使用5张数据表完成

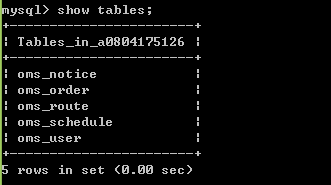
****

图7-6 数据表

oms\_user表结构

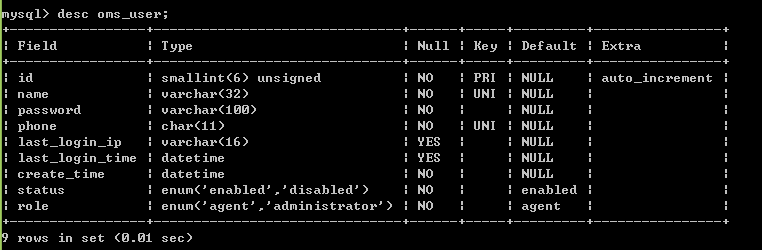
****

图7-7 oms\_user表结构

oms\_order表结构

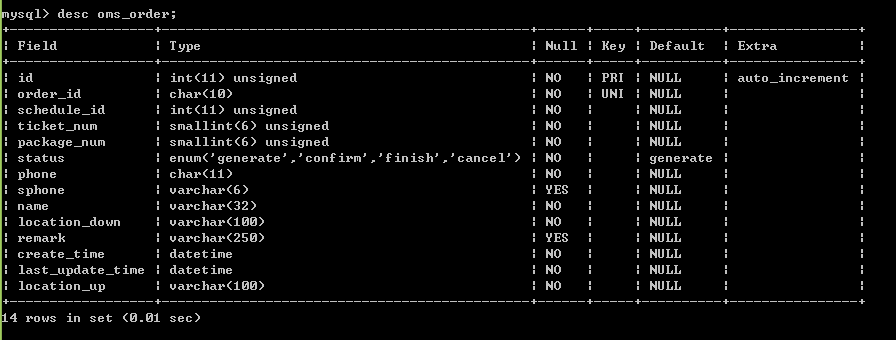
****

图7-8 oms\_order表结构

oms\_route表结构

****

图7-9 oms\_route表结构

oms\_schedule表结构

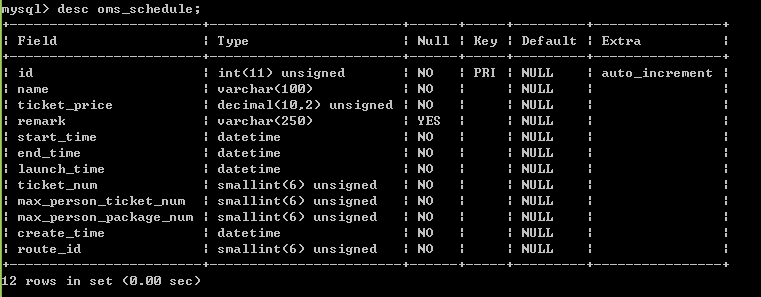
****

图7-10 oms\_schedule表结构

oms\_notice表结构

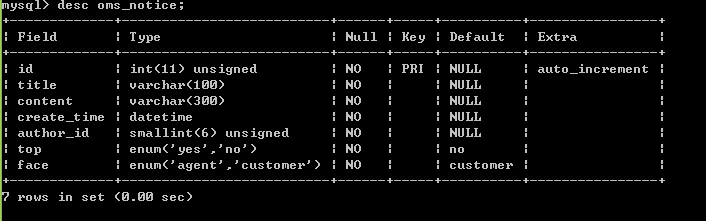
****

图7-11 oms\_notice表结构

### 7.3.3前端实现

根据产品backlog实现前台如下功能

1. 线路车次选择
2. 订单提交

根据流程图进行设计



图7-12 流程

1. 根据订单号查询订单情况

首页效果如下：



图7-13 首页

订单页面效果如下：

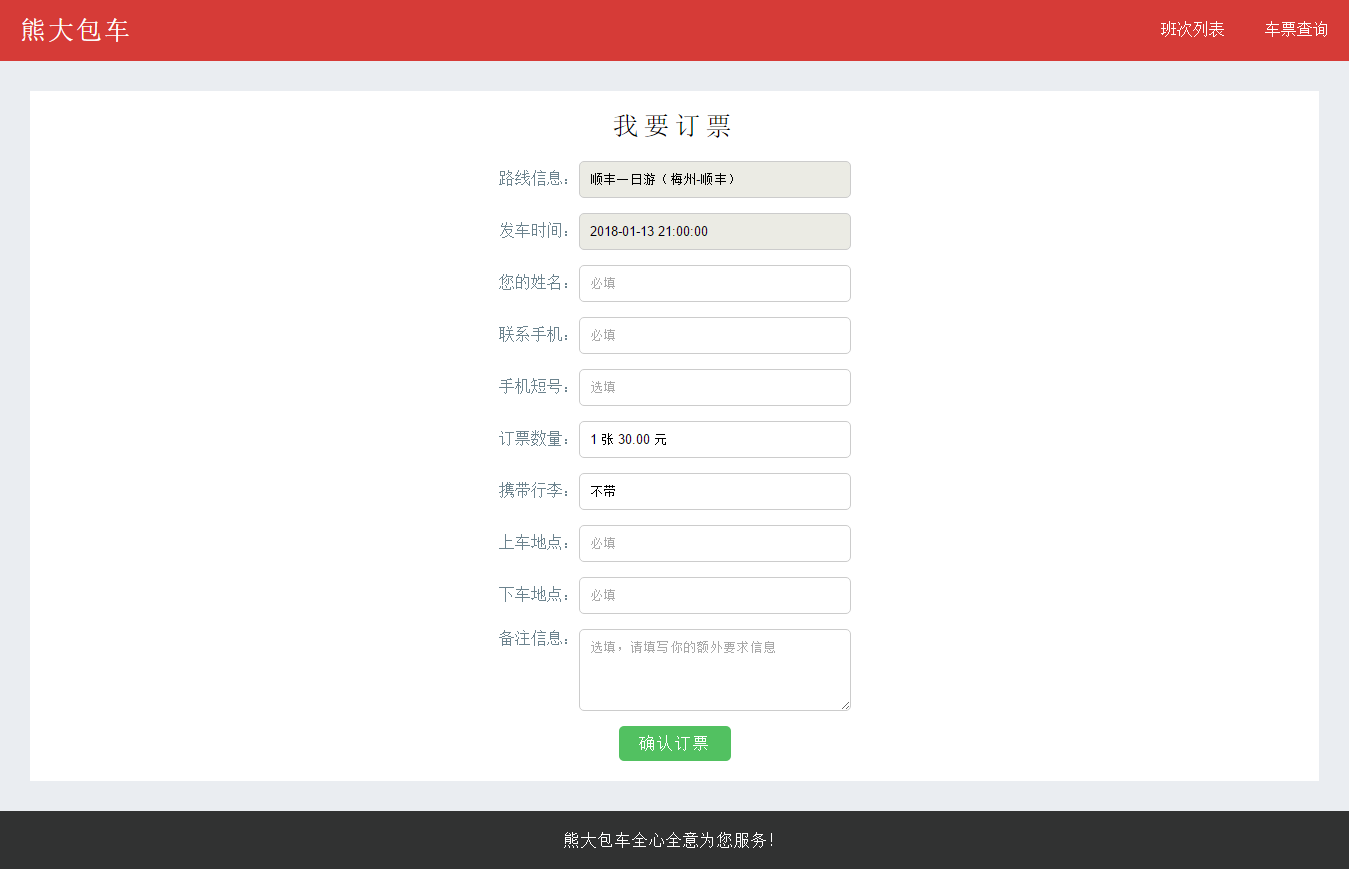


图7-14 订单页面

订单查询效果如下：



图7-15 订单查询页面

### 7.3.4后端实现

1. 防注入机制

遵循“不信任用户的任何输入”原则，对 PHP 的$\_GET、$\_POST 及输入数据进行以下处理： ″ 对 HTML 实体字符转换； ″ 对配置文件中指定的敏感字符进行删除； ″ 调用自定义函数对数据进行过滤。 所有数据库操作语句不可直接套入变量执行，必须通过 ThinkPHP 的数组过滤查询方式。

1. 根据产品backlog实现如下功能
2. 管理员登录实现

在管理员登录时，往数据库和 SESSION 中记录本次的登录时间，当 SESSION 和数据库中记 录的登录时间不一致的时，强迫不一致的用户下线并发送密码泄漏的告警信息。

1. 用户管理实现

在管理员修改相应角色的权限数据后，在线的管理员帐号应立即更新权限数据。

1. 线路管理实现

管理员修改相应线路数据后，前台数据和改线路代理人应立即更新数据。

1. 订单管理实现

管理员和线路代理人都有权限实现订单的受理，取消删除

登录页面效果如下：



图7-16 登陆页面

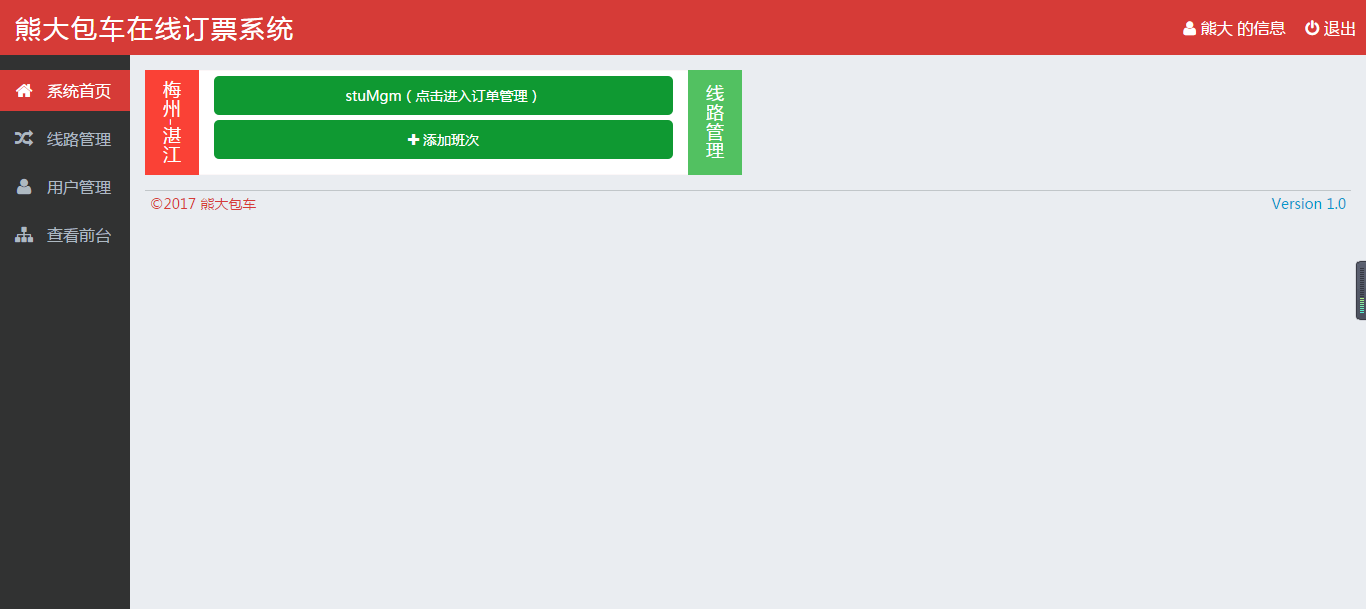
登录系统管理员效果如下：

图7-17 后台页面

线路管理效果如下：

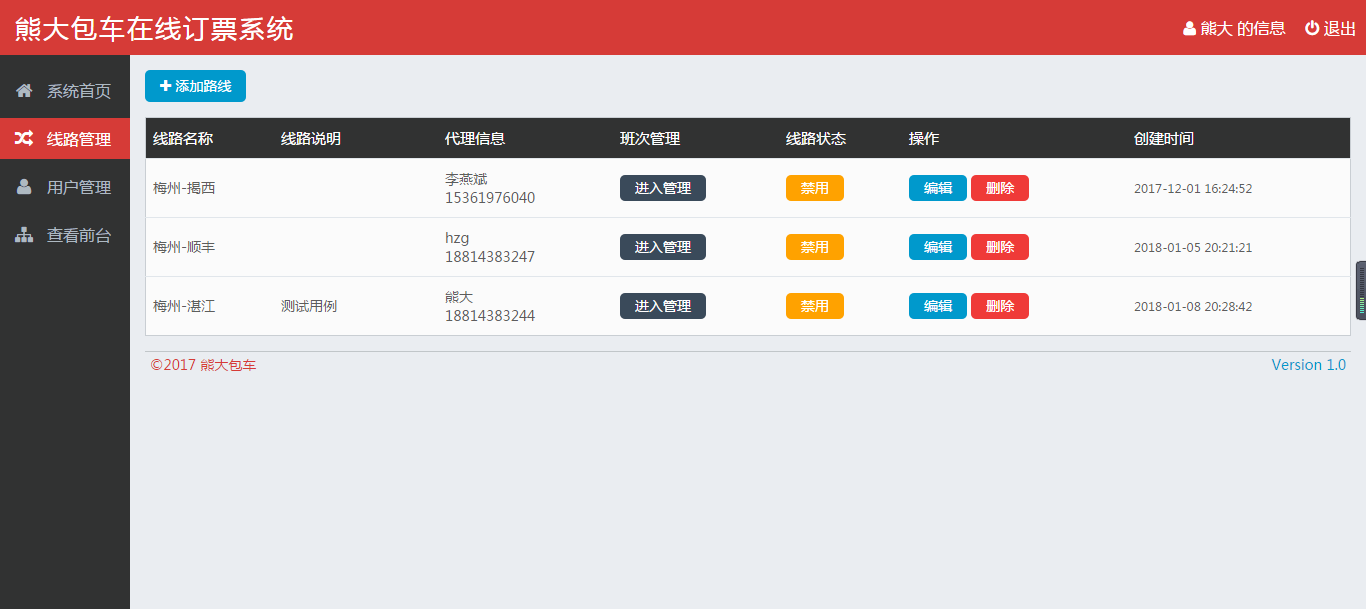


图7-18 线路管理页面

用户管理效果如下：

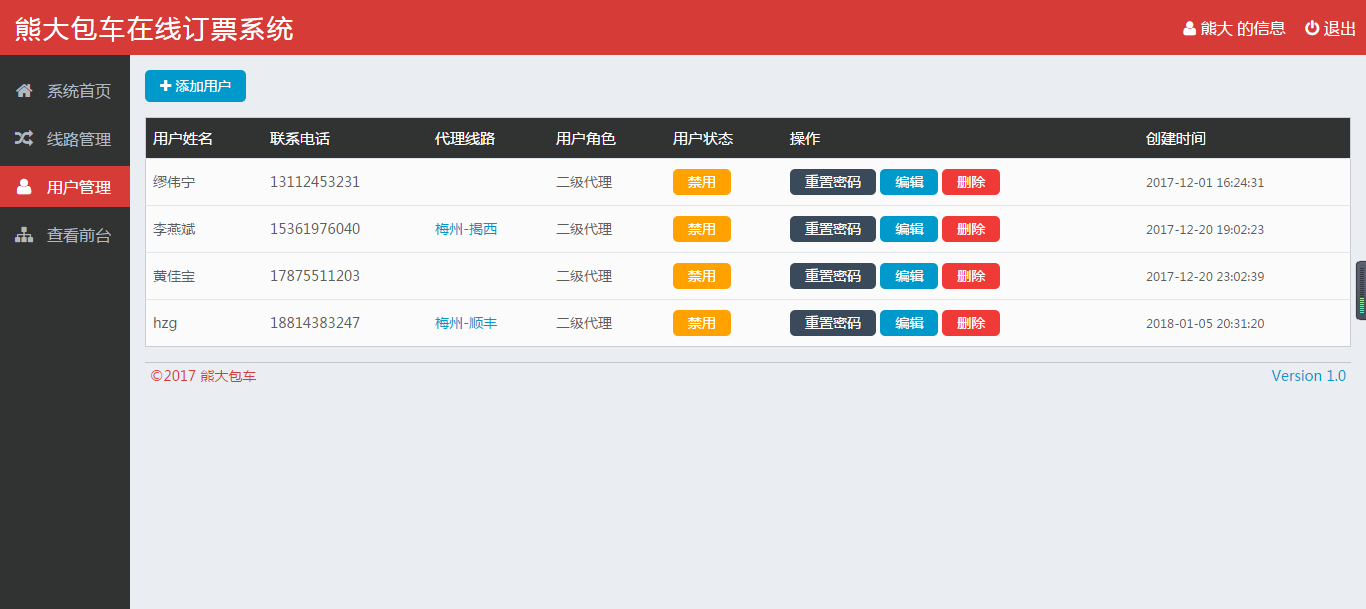


图7-19 用户管理页面

线路班次管理效果如下：



图7-20 线路班次管理页面

车次订单管理如下：

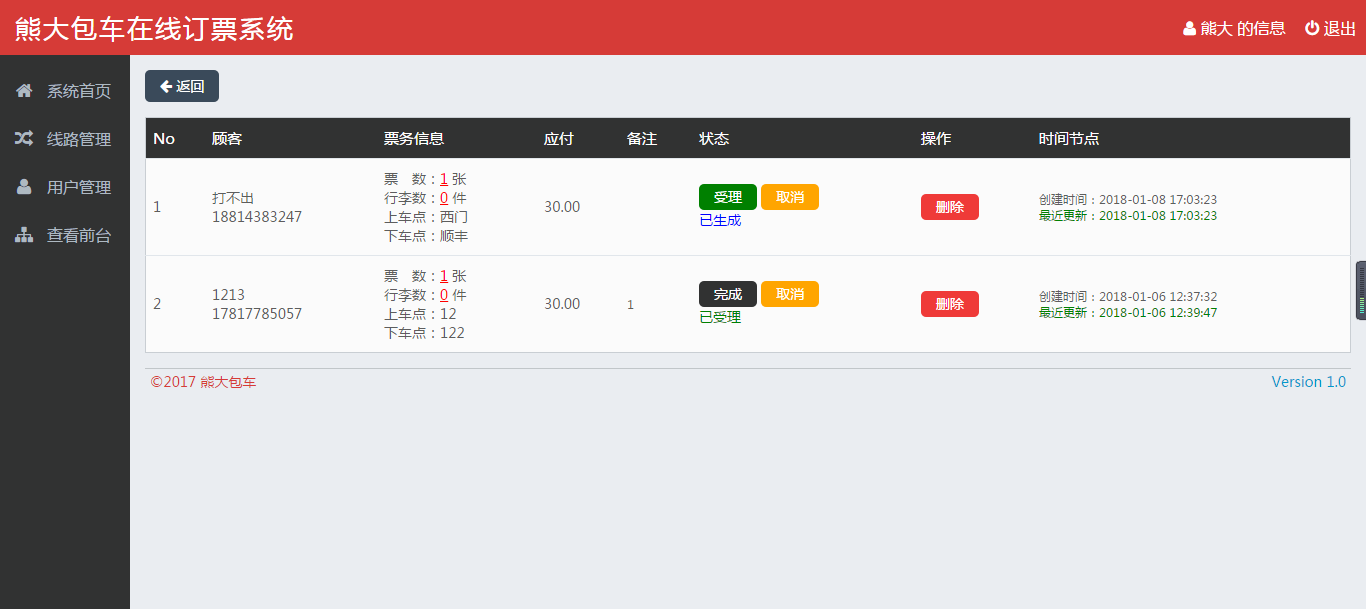


图7-21 车次订单页面

线路分配代理效果如下：

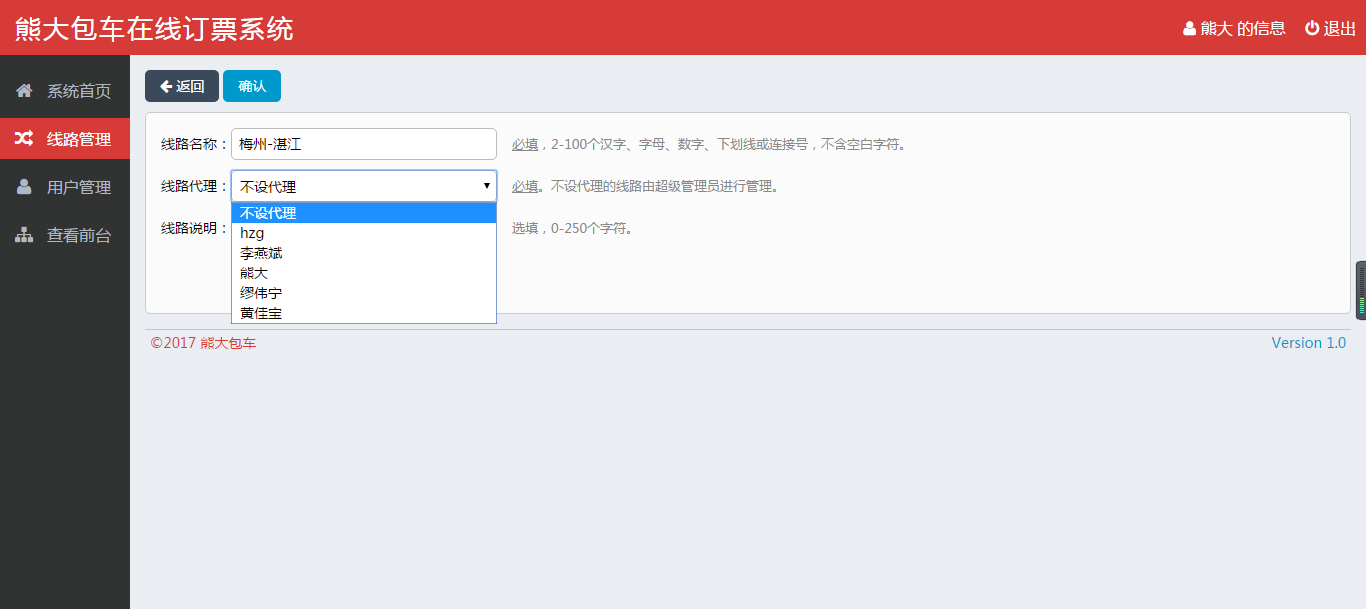


图7-22 线路分配代理页面

第八章 Sprint评审

8.1目标

根据团队这次产品backlog，评审相关的Backlog中的问题，检查是否已达到Sprint的目标。

8.2会议准备

1、对于每个人来说Sprint目标都是公开的。

2、对每个人来说既定产品Backlog是公开的，可获取的。

3、小组准备好工作站和设备等等，用以展示产品的新功能。

8.3会议进程

1. 团队按Backlog中的问题，逐个地介绍这次Sprint的结果，和演示新功能。
2. 产品负责人给出评价和反馈。
3. 如果产品负责人想要改变功能：添加一个新问题到产品Backlog中。
4. 如果对功能有一个新的想法：添加一个新问题到产品Backlog中。
5. 如果小组报告项目遇到阻碍现在还没有解决：把该障碍加入到障碍Backlog。
6. 每个Sprint结束都要有demo。
7. 只有完成的用户故事才演示，Demo要演示可以工作的功能。
8. Demo过程中要收集利益相关者的一些反馈。
9. Demo开始的时候要介绍一下原始的Sprint计划。
10. 以用户故事是否能成功交付来评价任务完成情况。

**8.4会议结果**

　　为 Sprint 计划会议2的进行准备好既定产品 Backlog

第九章 系统测试

## 9.1测试计划

9.1.1准备测试

|  |  |  |  |
| --- | --- | --- | --- |
| **包车在线订票系统功能** | **相关模块** | **回归测试范围** | **测试人员** |
| 顾客订票首页 | N/A (new) | N/A |  |
| 订单查询 | N/A (new) | N/A |  |
| 管理员登录 | N/A (new) | N/A |  |
| 管理员人员管理 | N/A (new) | N/A |  |
| 管理员线路管理 | N/A (new) | N/A |  |
| 管理员车次管理 | N/A (new) | N/A |  |
| 管理员订单管理 | N/A (new) | N/A |  |
| 二级代理车次管理 | N/A (new) | N/A |  |
| 二级代理订单管理 | N/A (new) | N/A |  |

表9-1 测试

9.1.2不准备测试

本次测试将不考虑关系数据库（My SQL）的安装和功能。我们假定数据库已安装并处于可操作的状态。我们也假定数据库表结构是准确的，包含需求规格说明书中定义的规定类型和宽度的字段。

9.1.3测试进度

|  |  |  |  |
| --- | --- | --- | --- |
| 测试活动 | 计划开始日期 | 实际开始日期 | 结束日期 |
| 制定测试计划 | 2017年12月28日 | 2017年12月28日 | 2017年12月30日 |
| 单元测试1天 | 2017年12月28日 | 2017年12月28日 | 2017年12月29日 |
| 功能测试1天 | 2017年12月30日 | 2017年12月30日 | 2017年12月30日 |

表9-2 测试进度

9.1.4测试资源

列出了在此项目的人员配备方面所作的各种假定。

注：可适当地删除或添加角色项。

|  |  |  |
| --- | --- | --- |
| 角色 | 所推荐的最少资源（所分配的专职角色数量） | 具体职责或注释 |
| 制定测试计划 | 2 | 制定和维护测试计划，设计测试用例及测试过程，生成测试分析报告 |
| 单元测试 | 2 | 执行集成测试和系统测试，记录测试结果 |
| 功能测试 | 1 | 设计测试需要的驱动程序和稳定桩 |
| 测试报告 | 2 | 编写测试驱动程序和稳定桩，执行单元测试 |

表9-3 测试资源

9.1.5测试环境

|  |
| --- |
| 软件环境（相关软件、操作系统等） |
| 操作系统：Windows 2000 Professional以上版本； |
| 应用服务器和Web服务器：Apache |
| 数据库系统：MY SQL |
| 客户端软件：MS Internet Explorer或Netscape Navigator等浏览器和office软件。 |
| 硬件环境（网络、设备等） |
| 对兼做应用服务器、Web服务器和数据库服务器的机器配置要求较高：256M以上内存，PIII500MHz以上CPU，10G以上可用硬盘空间； |
| 客户机只要能使用浏览器和Office软件即可。 |
| 网络条件和设备：网路连接卡或调制解调器。 |

表9-4 测试环境

9.1.6集成测试

集成测试，也叫组装测试或联合测试。在单元测试的基础上，将所有模块按照设计要求组装成为子系统或系统，进行集成测试。集成测试的主要目的是检测系统能否达到业务需求，检测系统对业务处理是否存在逻辑不严谨及错误，检测需求是否存在不合理的标准及要求。此阶段测试基于功能完成的测试。

集成测试表

|  |  |
| --- | --- |
| 测试目标 | 检测需求中业务流程，数据流的正确性 |
| 测试范围 | 需求中明确的业务流程，或组合不同功能模块而形成一个大的功能。 |
| 技术 | 利用有效的和无效的数据来执行各个用例、用例流或功能，以核实以下内容：  在使用有效数据时得到预期的结果。  在使用无效数据时显示相应的错误消息或警告消息。  各业务规则都得到了正确的应用。 |
| 开始标准 | 在完成某个集成测试时必须达到标准 |
| 完成标准 | 所计划的测试已全部执行。  所发现的缺陷已全部解决。 |
| 测试重点和优先级 | 测试重点指在测试过程中需着重测试的地方，优先级可以根据需求及严重来定 |
| 需考虑的特殊事项 | 确定或说明那些将对功能测试的实施和执行造成影响的事项或因素（内部的或外部的） |

表9-5 集成测试

9.1.7功能测试

对测试对象的功能测试应侧重于所有可直接追踪到业务功能和业务规则的测试需求。此类测试基于黑盒技术，该技术通过图形用户界面（GUI）与应用程序进行交互，并对交互的输出或结果进行分析，以此来核实应用程序及其内部进程。以下表为各种应用程序列出了推荐使用的测试概要。

功能测试表

|  |  |
| --- | --- |
| 测试目标 | 确保测试的功能正常，其中包括导航，数据输入，处理和检索等功能。 |
| 测试范围 |  |
| 技术 | 利用有效的和无效的数据来执行各个用例、用例流或功能，以核实以下内容：  在使用有效数据时得到预期的结果。  在使用无效数据时显示相应的错误消息或警告消息。  各业务规则都得到了正确的应用。 |
| 开始标准 |  |
| 完成标准 |  |
| 测试重点和优先级 |  |
| 需考虑的特殊事项 | 确定或说明那些将对功能测试的实施和执行造成影响的事项或因素（内部的或外部的） |

表9-6 功能测试

9.1.8用户界面测试

用MS Internet Explorer和Firefox对图形用户界面进行测试。在两种浏览器上对所有功能进行测试。

界面测试表

|  |  |
| --- | --- |
| 测试目标 | 通过测试进行的浏览可正确反映业务的功能和需求，这种浏览包括窗口与窗口之间、字段与字段之间的浏览，以及各种访问方法（Tab键、鼠标移动、和快捷键）的使用  窗口的对象和特征（例如，菜单、大小、位置、状态和中心）都符合标准。 |
| 测试范围 |  |
| 技术 | 为每个窗口创建或修改测试，以核实各个应用程序窗口和对象都可正确地进行浏览，并处于正常的对象状态。 |
| 开始标准 |  |
| 完成标准 | 成功地核实出各个窗口都与基准版本保持一致，或符合可接受标准 |
| 测试重点和优先级 |  |
| 需考虑的特殊事项 |  |

表9-7 集成测试

9.1.9性能评测

性能评测是一种性能测试，它对响应时间、事务处理速率和其他与时间相关的需求进行评测和评估。性能评测的目标是核实性能需求是否都已满足。

性能测试表

|  |  |
| --- | --- |
| 测试目标 | 核实所指定的事务或业务功能在以下情况的性能行为：  正常的预期工作量  预期的最繁重工作量 |
| 测试范围 |  |
| 技术 | 使用为功能或业务周期测试制定的测试过程。  通过修改数据文件来增加事务数量，或通过修改脚本来增加每项事务的迭代数量。  脚本应该在一台计算机上运行（最好是以单个用户、单个事务为基准），并在多个客户机（虚拟的或实际的客户机，请参见下面的“需要考虑的特殊事项”）上重复。 |
| 开始标准 |  |
| 完成标准 | 单个事务或单个用户：在每个事务所预期时间范围内成功地完成测试脚本，没有发生任何故障。  多个事务或多个用户：在可接受的时间范围内成功地完成测试脚本，没有发生任何故障。 |
| 测试重点和优先级 |  |
| 需考虑的特殊事项 | 综合的性能测试还包括在服务器上添加后台工作量。  可采用多种方法来执行此操作，其中包括：  性能测试应该在专用的计算机上或在专用的环境内执行，以便实现完全的控制和精确的评测。  性能测试所用的数据库应该是实际大小或相同缩放比例的数据库。 |

表9-8 性能评测

9.1.10容量测试

容量测试使测试对象处理大量的数据，以确定是否达到了将使软件发生故障的极限。容量测试还将确定测试对象在给定时间内能够持续处理的最大负载或工作量。本系统要为生成一份报表而处理一组数据库记录，那么容量测试就需要使用一个大型的测试数据库。检验该软件是否正常运行并生成正确的报表

容量测试表

|  |  |
| --- | --- |
| 测试目标 | 核实测试对象在以下高容量条件下能否正常运行：  连接或模拟了最大（实际或实际允许）数量的客户机，所有客户机在长时间内执行相同的、且情况（性能）最坏的业务功能。  已达到最大的数据库大小（实际的或按比例缩放的），而且同时执行多个查询或报表事务。 |
| 测试范围 |  |
| 技术 | 使用为性能评测或负载测试制定的测试。  应该使用多台客户机来运行相同的测试或互补的测试，以便在长时间内产生最繁重的事务量或最差的事务组合  创建最大的数据库大小（实际的、按比例缩放的、或填充了代表性数据的数据库），并使用多台客户机在长时间内同时运行查询和报表事务。 |
| 开始标准 |  |
| 完成标准 | 所计划的测试已全部执行，而且达到或超出指定的系统限制时没有出现任何软件故障。 |
| 测试重点和优先级 |  |
| 需考虑的特殊事项 | 对于上述的高容量条件，哪个时间段是可以接受的时间？ |

表9-9 容量测试

9.1.11安全性测试

侧重于安全性的两个关键方面：

⦁ 应用程序级别的安全性，包括对数据或业务功能的访问。

⦁ 系统级别的安全性，包括对系统的登录。

应用程序级别的安全性可确保：在预期的安全性情况下，用户只能访问特定的功能模块。

系统级别的安全性可确保：只有具备系统访问权限的用户才能访问应用程序。

安全测试用例。

安全测试表

|  |  |
| --- | --- |
| 测试目标 | 应用程序级别的安全性：核实用户只能访问其所属用户类型已被授权访问的那些功能或数据。  系统级别的安全性：核实只有具备系统和应用程序访问权限的用户才能访问系统和应用程序。 |
| 测试范围 |  |
| 技术 | 应用程序级别的安全性：确定并列出各用户类型及其被授权访问的功能或数据。  为各用户类型创建测试，并通过创建各用户类型所特有的事务来核实其权限。  修改用户类型并为相同的用户重新运行测试。对于每种用户类型，确保正确地提供或拒绝了这些附加的功能或数据。 |
| 开始标准 |  |
| 完成标准 | 各种已知的用户 类型都可访问相应的功能或数据，而且所有事务都按照预期的方式运行，并在先前的应用程序功能测试中运行了所有的事务。 |
| 测试重点和优先级 |  |
| 需考虑的特殊事项 | 必须与相应的网络或系统管理员对系统访问权限进行检查和讨论。由于此测试可能是网络管理员或者系统管理员的职能，可能不需要执行此测试。 |

表9-10 安全测试

## 9.2测试用例

测试用例必须包括详细的测试步骤、明确的期望结果，不能使用模棱两可的语言。

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **名称** | **描述** | **测试所需时间** |
| 1 | 管理员登录 | 前提条件：正常进入登录界面  步骤1：正确的管理员账号密码  期望结果：管理员能进入系统，操作系统  步骤2：不正确的管理员账号或密码  期望结果：提示账号或密码不正确、账号不存在或被禁用 | 2分钟 |
| 2 | 管理员人员管理 | 前提条件：管理员登录。  步骤：选择用户管理进行增删改操作  期望结果：能够增删改人员 ，控制用户账号状态 | 3 |
| 3 | 管理员线路的管理 | 前提条件：管理员正确登录。  步骤：进行线路的增删改 期望结果：能够完成正常的增删改 | 5分钟 |
| 4 | 给线路增删改车次 | 前提条件：用户登录、线路已存在  步骤： 分别在线路进行增删改操作 期望结果：进行增删改后，前台页面能及时的显示车次信息 | 5分钟 |
| 5 | 车次的时效 | 前提条件：管理员登录、线路已存在、车次存在  步骤：等待车次的时间  期望结果：时间过了，前台页面不在显示该车次 | 5分钟 |
| 6 | 订单的提交 | 前提条件：顾客正常访问网站  步骤：选择展示的线路车次，填写信息  期望结果：顾客能正常提交订单，并返回订单信息 | 5分钟 |
| 7 | 管理员对订单管理 | 前提条件：管理员登录、订单的提交  步骤：选择车次进入管理订单界面  期望结果：能够对订单受理，受理后通过短信提醒以及能取消和删除订单 | 5分钟 |
| 8 | 管理员分配线路代理 | 前提条件：管理员登录、线路存在、已有用户  步骤：选择线路分配改修路的代理人  期望结果： 线路代理人一个下拉列表展示已存的用户，进行分配 | 5分钟 |
| 9 | 代理人登录 | 前提条件：正常进入登录页面，管理员设定账号可用  步骤1：正确的账号和密码  期望结果：进入代理人管理页面  前提条件2：正常进入登录页面，管理员设定账号禁用  步骤2：正确的账号和密码  期望结果：提示账号被禁用  前提条件3：正常进入登录页面  步骤3：不正确的账号和密码  期望结果：提示账号或密码不正确、账号不存在或禁用 | 5分钟 |
| 10 | 代理人管理车次 | 前提条件：代理人登录、顾客提交订单  步骤：进入自己代理的线路，对线路车次进行增删改  期望结果：进行增删改后，前台页面能及时的更新信息 | 5分钟 |
| 11 | 代理人管理车次 | 前提条件：代理人登录、订单的提交  步骤：选择车次进入管理订单界面  期望结果：能够对订单受理，受理后通过短信提醒以及能取消和删除订单 | 5分钟 |

表9-11 测试用例

测试用例对应的测试结果

|  |  |
| --- | --- |
| **用列表Id** | **测试效果** |
| **1** |  |
| **2** |  |
| **3** |  |
| **4** |  |
| **5** |  |
| **6** |  |
| **7** |  |
| **8** |  |
| **9** |  |
| **10** |  |
| **11** |  |

# 第十章 总结

## 10.1 项目小组总结

### 10.1.1 目的

主要是总结本软件的开发经验和总结所学到的知识，以及对一个系统的软件设计的总体感悟，并将软件设计过程中遇到的问题加以阐述和说明。

对象：开发人员

### 10.1.2 项目背景

* 1. 项目的名称：包车在线订票系统。
  2. 项目的委托单位：嘉应大学。
  3. 项目的用户（单位）：大学生。
  4. 项目的主要承担部门：开发部。
  5. 项目建设背景：“包车在线订票系统”软件主要是通过网页的形式展示给用户，订票者可以在订票系统上订购所需的车票，管理者可以在后台发布路程和车票，然后将数据通过互联网存储在服务器端的MySQL数据库中。

### 10.1.3 开发结果

1. 产品

名称：包车在线订票系统

形式：B/S系统

文档：包车在线订票文档

1. 主要功能

|  |
| --- |
| **功能** |
| 前台和后台分开管理 |
| 前台展示线路车次列表 |
| 顾客提交订单，订单查询 |
| 后台管理员登录 |
| 系统管理员对人员增删改 |
| 订单管理 |
| 线路车次的增删改 |

表10-1 功能

1. 性能
2. 时间特性需求

在网络连接正常的情况下，响应时间为秒级。

1. 灵活性

当需求发生某些变化时，学生管理应用软件操作方式、数据结构、运行环境基本不会发生变化，变化只是将对应的数据库文件内的记录改变，或改变过滤条件。

1. 可用性

软件应该尽可能的一目了然，使一般用户能够使用

1. 可维护性

当用户使用本套订票系统时，遇到了软件本身的逻辑错误，软件维护人员可以迅速排除错误，解决问题。

1. 可兼容性

本系统要在windows XP，Vista，Win7等操作系统都可以直接运行。

### 10.1.4 评价

整个软件系统比较稳定，进行过比较严密的测试。

可用性：此系统具有很好的实用效果，适合于广大包车人。

可维护性：此系统比较稳定。维护是软件设计周期的最后阶段。

可转移/转换性：此软件运用PHP语言B/S开发，都可以使用浏览器互联网进行访问。。

### 10.1.5 总结

通过这次的项目开发，培养了我们的克服困难、坚持不懈的毅力以及团队合作精神。开发的这款软件，从设计到开发都经过了细致摸索和推敲和实地调查，做到了作品的原创性。这是一款独立研发且具有成品性质的软件，是我们大家共同努力的结果项目开发中，大家的能力，诸如大家的合作，个人的协作能力，策划能力，以及时间观念都有一定的提高。希望软件的设计能给大家耳目一新的感觉，丰富多彩的视听效果，希望成为广受用户的欢迎。 通过本次的团队合作项目得到了许多经验和教训：一个成功的设计应该是以用户为出发点，始终在考虑“用户需要什么”，我们制作的系统是乘客使用的，而不是自用的。一味从自我考虑，只做符合自己的软件，你会发现它的需求是如此的不足，功能有很大的缺失，最后会发现做出来的软件连你自己都失望。

软件一定要有自己的亮点，不要落入平庸。设计上一定要有重点，突出自己的特色和主要的功能。细节决定一切。

## 10.2 项目看板



图9-1 项目看板

## 10.3 燃尽图

小组成员分工合作，按照项目计划以及产品backlog逐步完成工作，每天记录工作量，完成统计项目燃尽图如下：

图10-2 燃尽图