**托付物流管理系统设计与实现**

摘要

现代社会中，随着经济全球化的不断推进，信息技术的发展不断成为人类生活中不可或缺的一部分。现阶段，网上购物的热潮以及市场的多元化，使物流行业得到了极大的发展，物流企业将速度决定效率为企业理念，把客户成为自己的上帝，提出以客户为核心的业务流程。所以，性能优良的物流管理系统在当下这个经济高速发展的时代，有着广阔的前景。

本物流管理系统主要完成物流分配管理阶段订单管理、用户管理、车辆管理、线路管理、配送点管理与货物管理。本文的主要工作包括：

设计思路是基于JSP技术与MVC设计模式，系统以Java语言体系结构作为开发基础，以关系型数据库Oracle数据库为数据支持，前端页面设计使用JSP动态页面技术，CSS + HTML + Jquery + Bootstrap相结合的开发体系结构。通过测试满足任务书的要求。

**关键字：**物流管理系统，MVC，数据库

**DESIGN AND IMPLEMENTATION OF TRUSTED LOGISTICS MANAGEMENT SYSTEM**

**ABSTRACT**

In modern society, with the continuous advancement of economic globalization, the development of information technology has become an indispensable part of human life. At this stage, the upsurge of online shopping and the diversification of the market, so that the logistics industry has been a great development, logistics companies will determine the efficiency of efficiency for the corporate philosophy, the customer to become their own God, put the customer as the core business processes. Therefore, the excellent performance of the logistics management system in the current era of rapid economic development, has broad prospects.

The logistics management system mainly completes the order management, user management, vehicle management, line management, distribution point management and cargo management in the logistics distribution management stage. The main work of this paper includes:

The design idea is based on JSP technology and MVC design pattern. The system uses Java language architecture as the foundation of development, relational database Oracle database for data support, front page design using JSP dynamic page technology, CSS + HTML + Jquery combined development system structure. Through the test to meet the requirements of the task book.

**Key words:** Logistics management, MVC, database

目 录

[ABSTRACT 2](#_Toc483686225)

[第一章 绪论 1](#_Toc483686226)

[1.1 引言 1](#_Toc483686227)

[1.2 项目开发及研究意义 1](#_Toc483686228)

[1.3 国内外研究现状及发展趋势 1](#_Toc483686229)

[1.4 本论文研究内容 2](#_Toc483686230)

[1.5 本章小结 3](#_Toc483686231)

[第二章 系统技术与框架分析 4](#_Toc483686232)

[2.1 J2EE开放平台 4](#_Toc483686233)

[2.2 Spring MVC框架 4](#_Toc483686234)

[2.3 Strust2 框架 5](#_Toc483686235)

[2.3 Mybatis 框架 6](#_Toc483686236)

[2.4 Bootstrap框架 6](#_Toc483686237)

[2.5 本章小结 6](#_Toc483686238)

[第三章 需求分析 7](#_Toc483686239)

[3.1 编写目的 7](#_Toc483686240)

[3.2 系统说明 7](#_Toc483686241)

[3.3 可行性分析 7](#_Toc483686242)

[3.4 系统总体需求 8](#_Toc483686243)

[3.5 系统非功能性需求分析 8](#_Toc483686244)

[3.5.1 系统界面需求 8](#_Toc483686245)

[3.5.2 软硬件环境需求 9](#_Toc483686246)

[3.5.3 产品质量需求 9](#_Toc483686247)

[3.6 系统功能性需求分析 9](#_Toc483686248)

[3.6.1 用户管理模块需求 10](#_Toc483686249)

[3.6.2 配送点管理模块需求 11](#_Toc483686250)

[3.6.3 线路管理模块需求 11](#_Toc483686251)

[3.6.4 订单管理模块需求 12](#_Toc483686252)

[3.6.5 车辆管理模块需求 12](#_Toc483686253)

[3.6.6 货物管理模块 13](#_Toc483686254)

[3.7 关键用例描述及业务活动图 13](#_Toc483686255)

[3.7.1 关键用例描述 13](#_Toc483686256)

[3.7.2 关键业务活动图 14](#_Toc483686257)

[3.7本章小结 15](#_Toc483686258)

[第四章 系统概要设计 16](#_Toc483686259)

[4.1 系统架构设计 16](#_Toc483686260)

[4.1.1 B/S体系结构 16](#_Toc483686261)

[4.1.2 系统开发环境 17](#_Toc483686262)

[4.2 系统结构设计 17](#_Toc483686263)

[4.2.1 系统总体设计 17](#_Toc483686264)

[4.2.2 登录注册结构 18](#_Toc483686265)

[4.2.3 系统后台主体结构 19](#_Toc483686266)

[4.2.4 系统实体类设计 20](#_Toc483686267)

[4.3 数据库设计 20](#_Toc483686268)

[4.3.1 数据库环境 20](#_Toc483686269)

[4.3.2 数据库原则设计 21](#_Toc483686270)

[4.3.3 数据库安全设计 21](#_Toc483686271)

[4.3.4 数据库逻辑设计 22](#_Toc483686272)

[4.3.5 数据库物理设计 23](#_Toc483686273)

[4.3.6 数据库维护与优化方案 25](#_Toc483686274)

[4.4本章小结 25](#_Toc483686275)

[第五章 系统详细设计与实现 27](#_Toc483686276)

[5.1 登录注册详细设计与实现 27](#_Toc483686277)

[5.2 后台主体详细设计与实现 29](#_Toc483686278)

[5.2.1 用户管理详细设计与实现 29](#_Toc483686279)

[5.2.2 配送点管理详细设计与实现 30](#_Toc483686280)

[5.2.3 线路管理详细设计与实现 32](#_Toc483686281)

[5.2.4 订单管理详细设计与实现 34](#_Toc483686282)

[5.2.5 车辆管理模块设计与实现 34](#_Toc483686283)

[5.2.6 货物管理模块设计与实现 34](#_Toc483686284)

[5.3 界面布局设计 34](#_Toc483686285)

[5.4 本章小结 34](#_Toc483686286)

[第六章 系统测试 35](#_Toc483686287)

[6.1 软件测试的原则和方法 35](#_Toc483686288)

[6.2 单元测试 35](#_Toc483686289)

[6.3 整体测试 38](#_Toc483686290)

[6.4 本章小结 38](#_Toc483686291)

[第七章 结论 39](#_Toc483686292)

[参考文献 40](#_Toc483686293)

[致谢 41](#_Toc483686294)

# 第一章 绪论

## 1.1 引言

现代社会中，随着经济全球化的不断推进，信息技术的发展已经不断成为人类生活中不可或缺的一部分。现阶段，网上购物的热潮以及市场的多元化，使物流行业得到了极大的发展，物流企业将速度决定效率为企业理念，把客户成为自己的上帝，提出以客户为核心的业务流程。所以，从上来看，性能优良的物流管理系统在当下这个追求快速发展的时代，有着宽阔的发展需求，随着个人消费对经济贡献的不断增大，消费需求将成为物流行业发展的主要推动力。以手机终端消费者为对象，个性化、多样化的物流体验将成为电子商务条件下消费者的核心需求，这其中首当其冲的就是快递服务。快递是指快递公司通过铁路、公路和空运等交通工具，对客户进行快速投递的物流活动，属于门对门服务，根据时间快慢和递送质量区别于其他运输方式。快递的高效运转是建立在完善的网络基础上的，且网络具有相当强的整合能力。现阶段看来开发一种基于web端的物流管理系统，是适应信息化发展的必然趋势。

本物流管理系统就是针对本阶段中小物流企业的实际需求而去开发的一款基于web端的物流管理系统。本系统能够帮助企业实现对物流过程的调度优化和动态监控，并实现对物流的管控，来提高物流速度，提供高效，实用的物流管理系统。

## 1.2 项目开发及研究意义

随着现代企业信息化建设的提高，办公自动化，管理现代化和决策科学化，是现代公司办公系统的标志。并随着信息化的不断发展，计算机的使用早已深入人类社会的各个方面。随着人们对数据库的使用和依赖，更大大促使了信息技术的发展，在物流管理领域，开发基于web的物流管理系统，一方面可以提高办公效率，另一方面可以实现高效的物流管理，这些是物流管理进行现代化，标准化建设的必然趋势。

本文主要的研究内容是设计一套基于JAVA后台技术和数据库技术的现代企业信息系统平台，现阶段，它并不能简单的认为是一套管理软件，更准确的说是提供了强大的应用支撑。在提高管理人员对信息处理的准确性，发挥系统的基本功能的前提下，并逐步提高工作效率，节省了人力成本。本物流管理系统是以JAVA EE为开发平台，以数据库oracle作为系统的后台操作。本系统采用三层模式开发，系统功能全部实现在后台完成，实现前台展示，用户可以通过文档等来实现对系统的操作。

## 1.3 国内外研究现状及发展趋势

在现有的研究中，物流指数已经成为社会发展的重要的数据标准，所谓物流指数就是指

根据各个地区经济发展状况、物流发展基础条件以及物流发展对环境的影响的系统性评价指标，是人们对物流行业发展进行综合诊断和物流行业管理的必要手段，是综合衡量一个地区物流业发展程度的重要指标。[物流](http://baike.baidu.com/item/%E7%89%A9%E6%B5%81)指数的研究发布有利于进一步加强与世界发达国家物流的交流、比较和研究；有利于推进[物流行业](http://baike.baidu.com/item/%E7%89%A9%E6%B5%81%E8%A1%8C%E4%B8%9A)的科学指导和[规范管理](http://baike.baidu.com/item/%E8%A7%84%E8%8C%83%E7%AE%A1%E7%90%86/7011050)；对政府部门宏观决策和物流相关企业微观经营都具有极大的帮助作用。

现今阶段，根据数据显示物流业归属于第三产业，然而物流服务业在其中有着很大的比重，根据数据，物流服务业占2011年第三产业的比重为17.6%，是所有服务行业占比最大的一类。随着中国加入世界贸易组织，并在协议中承诺全面开放第三产业的承诺，物流服务作为第三产业中最重要的一部分，肯定会受到世界巨头（UPS，速贷，联邦快递）冲击，对我国现有的物流企业形成很大的竞争。如今，物流行业从当时的手工下单、手工核查数据的方式，转变成现如今的物流管理平台的信息化方式，进而大量的节省的人力支出，使得配送管理流程实现自动化、一体化。进入2010年后，随着互联网的全球化发展，出现了一种智能运输管理系统，它即是物流系统的一种，也标志这我国未来研究的方向。它的主要功能是采用信息处理、图形处理、通信、控制、电子、互联网的现今技术，使人力、车辆、道路更加协调地结合在一起，用来提高交通运输种的效率问题以及生产效率的综合管理系统。我国从70年代开始对电子信息技术在道路交通和轨道交通领域的发展以及应用，尽力了相应的电子信息技术、互联情报、交通工程、自动化控制的研究机构。尽管如此，由于研究对象的分散以及研究水平受社会环境所限，形成多数研究项目是针对交通运输的某一局部问题而进得的，缺乏一个综全性的、具有战略意义的研究项目恰恰是覆盖这些领域的一项综全性技术，也就是说可以通过智能运输系统将原来这些互不相干的项目有机的联系在一起，使公路交通系统的规划、建设、管理、运营等各方面工作在更高的层次上协调发展，使公路交通发挥出更大的效。

所以随着信息技术的发展，以及互联网全球化的趋势，未来物流行业发展将是朝着信息化、网络化、全球化发展。并且信息技术将是最主要的核心竞争力，[信息技术](https://www.baidu.com/s?wd=%E4%BF%A1%E6%81%AF%E6%8A%80%E6%9C%AF&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3PADvPWmYrHmdPHK9uj0Y0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHn3nHmsP161rjcsnHc1nHT3n0)、网络技术日益广泛用于物流领域，物流与电子商务日益融合。而互联网的出现则促使物流行业发生了革命性的变化，基于互联网的及时准确的信息传递满足了物流系统高度[集约化管理](https://www.baidu.com/s?wd=%E9%9B%86%E7%BA%A6%E5%8C%96%E7%AE%A1%E7%90%86&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3PADvPWmYrHmdPHK9uj0Y0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHn3nHmsP161rjcsnHc1nHT3n0)的信息需求，保证了[物流网络](https://www.baidu.com/s?wd=%E7%89%A9%E6%B5%81%E7%BD%91%E7%BB%9C&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y3PADvPWmYrHmdPHK9uj0Y0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHn3nHmsP161rjcsnHc1nHT3n0)各点和总部之间以及各网点之间信息的充分共享。

## 1.4 本论文研究内容

本文主要研究了如何利用互联网技术建立一套完善稳定的物流管理系统，为用户提供一种简便的物流管理模式，本系统基于B/S架构的Web轻量级开发框架—SSM框架（SpringMVC+Strust2+MyBatis），标准的MVC设计模式。后台开发使用Java语言，数据存储使用Oracle数据库，前端主要使用Bootstrap框架和JQuery库。本文主要研究内容包含以下几个方面：

1. 针对选题背景和现状进行分析，并分析研究的目的和意义。
2. 系统技术框架研究与分析。
3. 对系统进行可行性分析和需求分析，包括功能性需求和非功能性需求。
4. 系统总体设计，包括架构设计、概要设计以及数据库设计。
5. 系统详细设计与实现，包括技术难点和关键问题分析，以及前台子系统和后台子系统的详细设计与实现。
6. 系统运行部署和功能测试、性能测试。
7. 总结整个项目流程，得出结论。

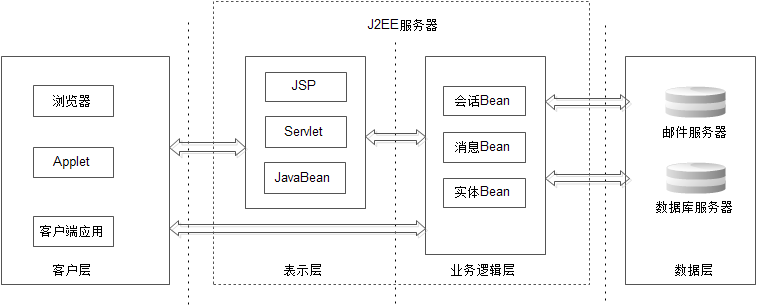
## 1.5 本章小结

论文主要分为七章。其中第一章为绪论，讲述课题背景、研究内容、课题任务等内容。 第二章为系统技术基础，主要介绍系统中所涉及的核心技术。第三章为系统需求分析，主要对系统的功能性需求和非功能性需求进行分析。第四章为系统详细设计，主要是对系统的功能进行详细的说明并进行数据流向分析，设计数据库并分析数据。第五章为系统开发实现，主要说明了系统架构、系统的业务的流程、系统的界面内容等。第六章为系统测试，主要是对系统进行测试，其中包括功能测试和模块测试，第七章为总结，是对课题的总结，并且说明课题的成果，提出课题需要修改之处。

# 第二章 系统技术与框架分析

## 2.1 J2EE开放平台

J2EE（Java 2 Platform Enterprise Edition—java2平台企业版），是使用Java进行企业级开发的一套扩展标准。 J2EE组件被装配在一个J2EE应用中，是具有独立功能的软件单元。J2EE应用典型的四层结构划分为客户层、表示层、业务逻辑层和数据层，如图2-1所示，通过各层之间的数据交互，完成系统的实现。

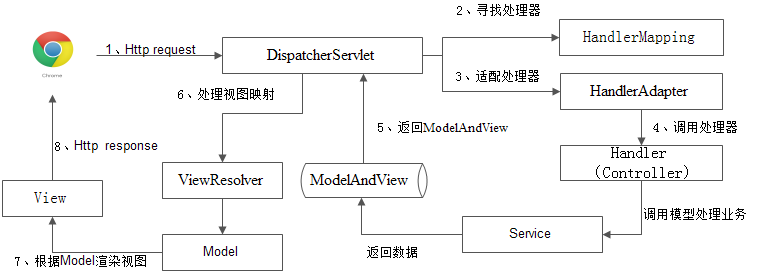


**图2-1 J2EE分层结构图**

## 2.2 Spring MVC框架

Spring 框架提供了构建 Web 应用程序的全功能 MVC 模块。Spring MVC 分离了控制器、模型对象、分派器以及处理程序对象的角色，这种分离让它们更容易进行定制，是一个标准的MVC框架，灵活度高，扩展性强。

SpringMVC框架主要功能流程是首先由DispatcherServlet捕获HTTP请求，然后DispatcherServlet根据相应的配置文件中的配置对请求的URL进行解析，得到资源标识符URI，然后根据URI，调用HandlerMapping获得该Handler配置的所有相关对象（包括Handler对象和Handler对象对应的拦截器），最后以HandlerExecutionChain对象的形式返回，DispatcherServlet 根据Handler，选择一个合适的HandlerAdapter，然后提取Request中的模型数据，填充Handler入参，开始执行Handler（Controller)，Handler执行完成后向DispatcherServlet返回一个ModelAndView对象，其中包含了处理结果的视图和视图中要使用的数据。DispatcherServlet根据得到的ModelAndView中的视图对象，选择一个合适的ViewResolver（视图解析器），然后ViewResolver 结合Model和View来渲染视图，最后视图负责将渲染结果返回给客户端，如图2-2所示。

  
**图2-2 Spring MVC工作流程**

## 2.3 Strust2 框架

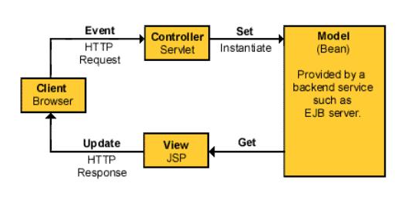
Struts2是Apache软件基金下Jakarta项目的一部分，Struts2是一个基于MVC设计模式的Web应用框架[5]，Struts2 是由传统的Struts1、WebWork两个经典的MVC框架发展起来，无论是从Struts2设计的角度来看，还是从Struts2的实际项目中的易用性来看，Struts2都是一个非常优秀的MVC框架。与传统的Struts1相比，Struts2允许使用普普通通的、传统的Java对象作为Action；Action的execute()方法不再与ServletAPI耦合，因此更容易测试；支持更多的师徒技术；基于AOP思想的拦截器机制，提供了极好的可扩展性；更强大、更易用的输入验证；整合Ajax支持。这些都是Struts2的巨大吸引力。

Struts2是Java开发中使用技术最为重要的框架之一，通常情况下，Struts2的工作步骤为：

* Struts2拦截器可以根据配置文件Struts.xml来确定所需调用的Action
* 服务器根据请求来创建ValuStack栈对象，并进行把Action放到栈顶。
* 服务器调用Action和相应的拦截器。

1. 调用action之前，拦截器可对ValueStact中的action的数据进行操作。
2. 调用action：根据输入计算输出。
3. result把ValueStack中的数据以某种形式向用户输出。
4. 拦截器做收尾工作。

Struts2的工作原理如图2-3所示：



**图2-3 Struts2原理图**

## 2.3 Mybatis 框架

MyBatis 是一个支持定制化SQL、存储过程以及高级映射的优秀的持久层框架，它消除了几乎所有的[JDBC](http://baike.baidu.com/view/25611.htm)代码和参数的手工设置以及[结果集](http://baike.baidu.com/view/691831.htm)的检索。MyBatis 使用简单的 XML或注解用于配置和原始映射，将接口和 Java 的POJOs映射成数据库中的记录[7]。MyBatis框架的工作流程主要分为四步：

1. 加载配置并初始化：Mybatis的配置来源于配置文件和Java代码的注解，将SQL的配置信息加载成一个个MappedStatement对象（包括了传入参数映射配置、执行的SQL语句、结果映射配置），存储在内存中。
2. 接收调用请求：当API接口层收到调用请求时，会接收到传入SQL的ID和传入对象（可以是Map、JavaBean或者基本数据类型），Mybatis根据传入参数中的SQL的ID查找对应的MappedStatement对象，并根据传入参数对象对其进行解析，得到最终要执行的SQL和执行传入参数。
3. SQL执行：根据得到的最终SQL语句和执行传入参数获取数据库连接并执行，得到数据库操作结果。
4. 结果返回：可以按映射配置将最终的处理结果转换HashMap、JavaBean或者基本数据类型，并将最终结果返回。

## 2.4 Bootstrap框架

Bootstrap 是一种基于 HTML、CSS、Javascript的前端框架，它简洁灵活，提供了优雅的HTML和CSS规范，兼容大部分jQuery插件，并包含了丰富的Web组件和[jQuery](http://baike.baidu.com/view/1020297.htm)插件，使开发者可以快速的搭建一个漂亮、功能完备的网站。

## 2.5 本章小结

本章主要介绍了系统运用到的技术与框架，概括性介绍了的Strust2框架、SpringMVC框架工作流程、Mybatis框架工作流程，以及其他前端框架。

# 第三章 需求分析

## **3.1 编写目的**

为了明确所用用户的基本需求，更好的完善和改进对用户的了解，使系统更加量化个明晰开发所需的工作量和工作进度，特此编写需求说明书。本物流管理系统的主要读者是：

（1）客户

（2）需求分析人员

（3）软件开发人员

（4）软件测试人员

（5）项目负责人

## **3.2 系统说明**

托付物流管理系统主要为中小物流公司的实际需求而开发的一套基于web端的物流管理系统。系统的研发和部署能够实现企业对物流过程的优化调度和动态控制，快速的整合企业的物流业务，本物流管理系统在协助工作人员对日常物流管理和人员管理。通过本物流管理系统，物流公司可以实现对车辆、订单、货物、订单线路的动态管理：管理人员能更快地实现对订单的修改和添加；车辆管理人员可以对已有车辆进行查看和调度；财务人员可以通过本系统实现对运输费用的计算。

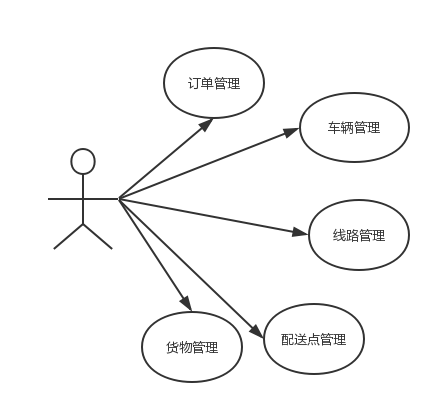
## **3.3 可行性分析**

系统可行性分析的目的是用最小的代价在尽可能短的时间内确定问题是否能够解决，有效地避免对系统的额外投资浪费。托付物流管理系统的可行性主要体现在以下几个方面：

1. 技术可行性：本系统将采用的开发技术包括CSS、Javascript、Java、Spring、Mybatis、Bootstrap，JSP等，在这些技术都是经过实践的检验已成熟稳定，可以查阅大量的技术资料，并且使用的Oracle数据管理系统和MyEclipse开发工具被广泛运用，使得技术上是完全可行的。
2. 经济可行性：本系统的开发环境均使用现阶段开发人员常用的免费开源软件，并使用可视化编程工具以及高效的数据库管理系统，轻量级的后端Spring开发框架和流行的Bootstrap前端框架的使用也快速的完成系统的研发周期，使本系统在成本控制方面做到了完美。
3. 操作可行性：系统使用菜单式，实现用户与Oracle数据库之间的交互Bootstrap前端框架搭建的简洁友好的界面设计，使操作便捷，用户只需要掌握最基本的计算机操作规范和系统流程即可，不需要掌握其它技术知识，所以系统具有操作可行性。
4. 社会可行性：本系统的研发可以简化并提高物流系统工作开展的流程，促进基层员工和企业管理之间的高效融合，并帮助基层员工减少大量工作量，有效提高其工作效率，因此，系统具有很强的社会可行性。

## **3.4 系统总体需求**

托付物流管理系统，面向的用户是基层操作员，主要针对基层操作员对物流订单的基本操作，实现即时的物流订单管理。操作员可以根据订单的具体信息对订单中的，配送点、车辆、货物、路线进行管理，并可以根据以前订单的详情进行订单状态的查询，实现模糊查询，也可以根据订单时间进行查询。系统的业务用例如图3-1所示：



**图3-1 业务用例图**

## **3.5 系统非功能性需求分析**

为了保证系统的稳定安全的运行，系统在开发初期需要考虑系统的易理解性、可扩展性、可测试性、易维护性、可移植性等性能指标。系统的非功能需求答题可以分为系统界面需求、软硬件环境需求和产品质量需求三个方面。

### 3.5.1 系统界面需求

系统界面需求主要包含界面风格、操作易用性、系统界面等，如表3-1所示

**表3-1 系统界面需求**

|  |  |  |  |
| --- | --- | --- | --- |
| **需求名称** |  | **详细要求** |  |
| 界面风格  操作易用性  登录注册界面  系统界面 |  | 色调统一，风格简约  界面设计简单易操作  采用动感设计  采用蓝白设计 |  |

### 3.5.2 软硬件环境需求

软硬件环境需求是对系统开发过程中对软件和硬件的基本需求，具体如表3-2所示。

**表3-2 软硬件环境需求**

|  |  |
| --- | --- |
| **需求名称** | **详细要求** |
| 硬件需求 | 支持软件开发的PC，配置要求：  1)处理器：Inter（R）Core（TM）i5-4200U CPU @ 1.60GHz 2.30GHz  2)内存：4G及以上  3)硬盘：80G以上  4)显卡：支持DirectX 9 128M 及以上 |
| 软件需求 | 1)操作系统：Windows 10(64 bit)+  2)浏览器版本：至少一个IE8及其以上、或其它主流浏览器  3)数据库：Oracle 11g、Oracle sqldeveloper  4)服务器：Apache-tomcat-7.0.47+  5)开发工具:MyEclipse 2014/jdk1.8.0\_91 |

### 3.5.3 产品质量需求

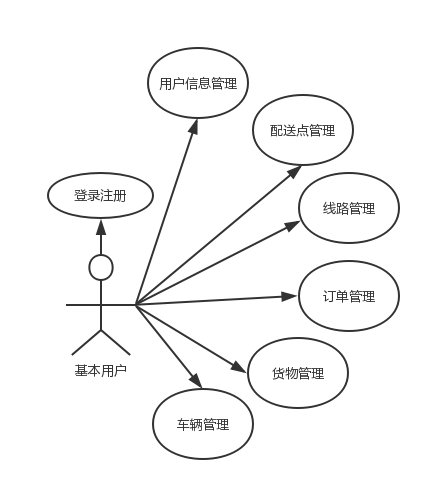
产品质量是项目开发成功与否的关键点，包含正确性、健壮性、可靠性等，具体情况如表3-3所示。

**表3-3 产品质量需求**

|  |  |
| --- | --- |
| **主要质量属性** | **详细要求** |
| 正确性 | 系统能正确实现功能性需求，无逻辑或页面展现错误 |
| 健壮性 | 系统在无外力因素影响情况下，经历压力测试崩溃机率小于5% |
| 可靠性 | 系统具有错误处理机制，信息准确，真实可靠 |
| 性能，效率 | 排除网络干扰，正常情况下系统响应速度控制在5秒之内 |
| 易用性 | 页面简单易操作，有引导界面 |
| 安全性 | 保障系统安全性，不能向第三方泄露用户个人信息 |
| 可扩展性 | 系统可以在目前基础上扩展更多功能 |
| 兼容性 | 浏览器兼容：IE8及其以上、Chrome 20及其它主流浏览器 |
| 可移植性 | 系统子系统模块可以移植、并具有详尽、规范的设计文档 |

## 3.6 系统功能性需求分析

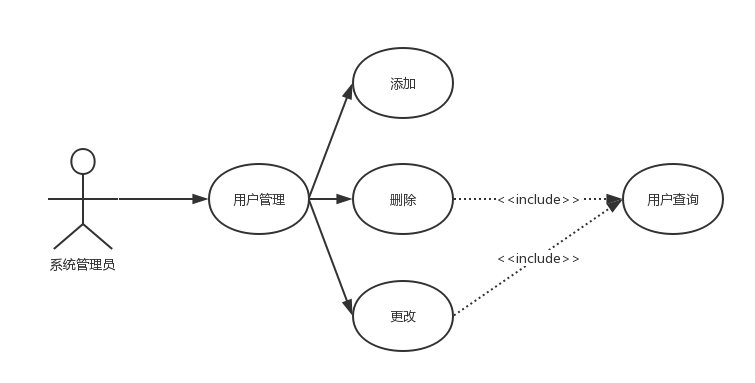
通过对业务流程的分析，托付物流管理系统用户角色主要为基本用户，系统结构分为登录注册和前台系统。登录注册是针对用户的登录和注册的管理，前台系统用户是基本操作员，包含个人信息管理、线路管理、配送点管理、车辆管理、货物管理和订单管理。系统总体用例如图3-2所示。



**图3-2 系统总体用例图**

### 3.6.1 用户管理模块需求

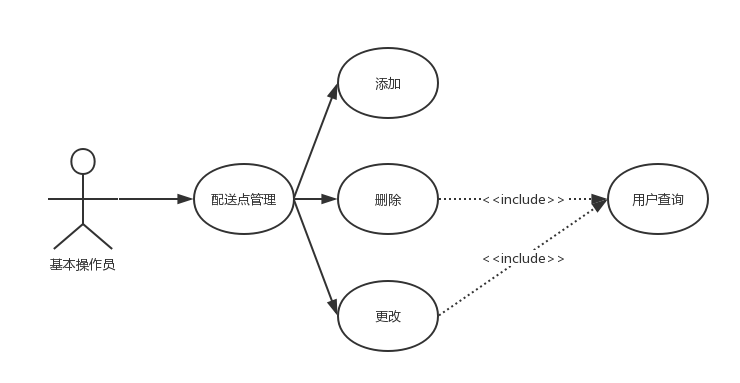
系统用户管理主要有系统管理员和基本操作员，由于基层操作员存在一定的流动性，需要对系统用户进行管理。系统管理员拥有用户管理的权限。用户管理模块用例如图3-3所示。



**图3-3 用户管理模块用例图**

### 3.6.2 配送点管理模块需求

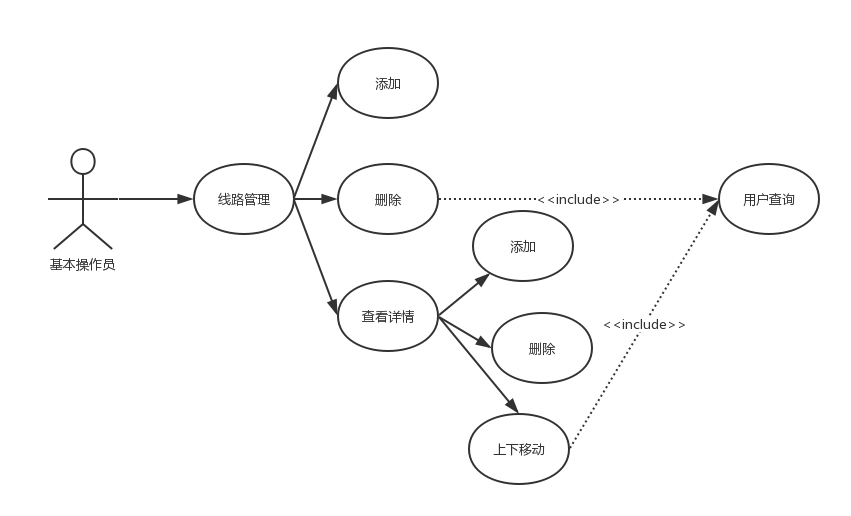
配送点管理主要功能是完成对订单中配送点的设置，有开始配送点，终点配送点，用户管理具有增删改查的权限，配送点管理需求用例图如图3-4所示。



**图3-3 配送点管理用例图**

### 线路管理模块需求

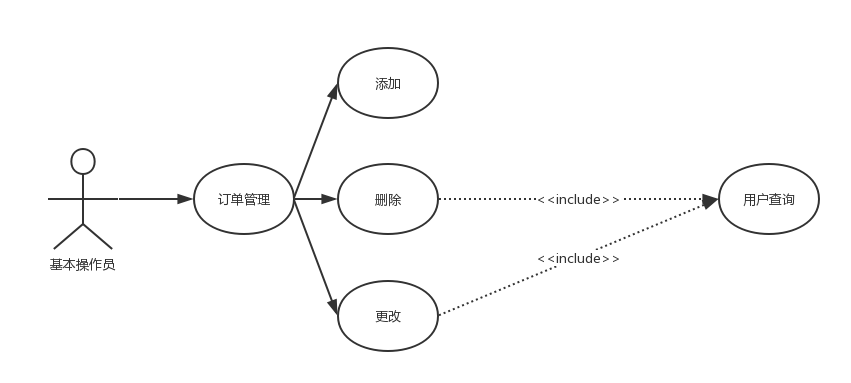
线路管理的主要功能是系统选择添加路线，输入增加路线的名称，然后添加路线经过的配送点，并可以通过添加、移除、上移、下移等操作调整线路顺序。基本用户管理员具有增删改查的权限，线路管理需求用例图如图3-4所示。



**图3-4 线路管理用例需求**

### 订单管理模块需求

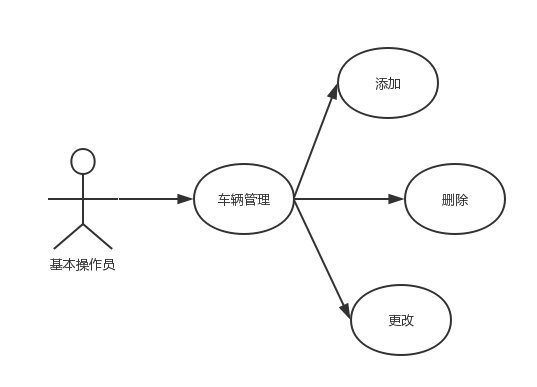
系统订单管理的主要功能就是对基本订单的添加，查询以往订单，按照订单状态查询，订单时间等的模糊查询，并实现修改订单信息，删除订单的那个权限，订单管理的基本用图如图3-5所示。



**图3-5 订单管理用例图**

### 车辆管理模块需求

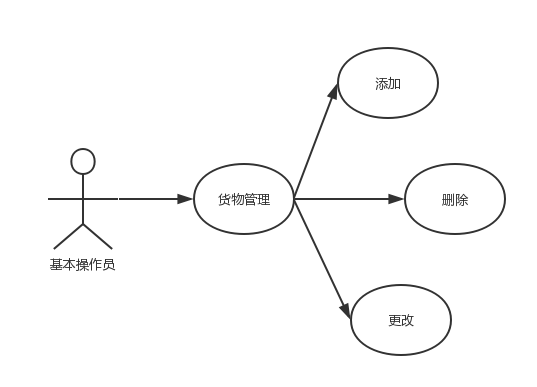
本系统的车辆管理主要功能为公司负责购买所需车辆，以及车辆的维护；在该系统中我们只需要把车辆的增、删、改做完即可。基本操作员对系统车辆管理具有增删改的权限，车辆管理的基本用例图如下图3-6所示。



**图3-6 车辆管理用例图**

### 货物管理模块

本系统货物管理主要功能管理货物种类，并对货物种类进行修改和删除操作，货物管理基本用例图如下图3-7所示。



**图3-7 货物管理用例图**

## **3.7 关键用例描述及业务活动图**

在系统的需求分析阶段制定系统的用例规约，可进一步详细的说明该系统需求，是系统设计阶段的基础，也是测试阶段设计用例的依据。限于论文篇幅，选取关键功能点介绍其用例规约。

### 3.7.1 关键用例描述

系统的核心功能主要在线路管理模块和订单模块，其中关键用例的用例规范描述如

表3-4、表3-5所示。

**表3-4 线路详情用例描述**

|  |
| --- |
| **用例编号/名称 case1/线路详情** |
| **用例概述** 基本用户对线路的详情进行查看并可以的修改、删除  **主要参与者** 基本操作人员  **前置条件** 系统运行正常  **后置条件** 详细信息修改成功，系统保存 |

**续表3-4 线路详情用例描述**

|  |
| --- |
| **用例编号/名称**  **线路详情** |
| **步骤 活动**  **基本事件流**  1 基本用户登录系统  2 登陆成功，查看线路管理，显示列表  3 查看详细线路，并对线路点进行修改  4 返回线路管理列表，查看是否保存  **步骤 活动**  **异常事件流**  1 登录失败  2 系统不保存修改后的线路信息  **其他**  无 |

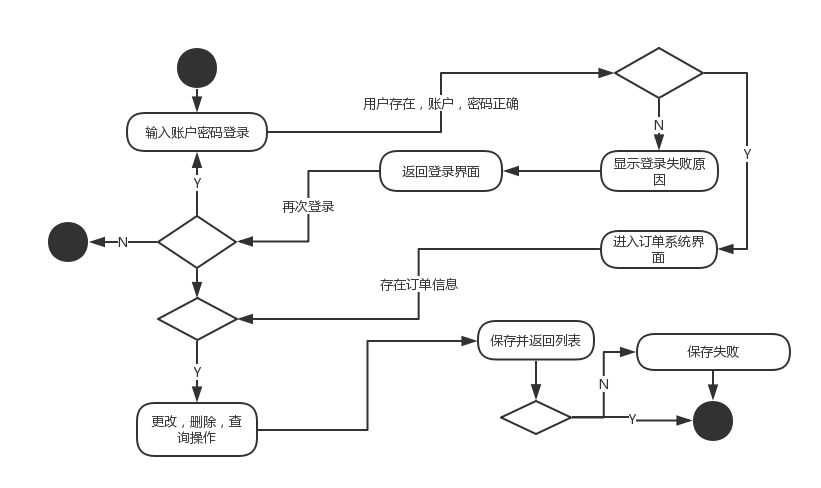
**表3-5 订单管理用例描述**

|  |  |  |
| --- | --- | --- |
| **用例编号/名称** | **Case2 / 订单管理** | |
| **用例概述** | 操作员在系统中添加订单，删除，更改订单等操作。 | |
| **主参与者** | 用户操作员 | |
| **前置条件** | 系统正常运行 | |
| **后置条件** | 回复成功，系统保存回复信息 | |
| **基本事件流** | **步骤** | **活动** |
| 1 | 基本操作员使用账号密码登录系统 |
| 2 | 登录成功，进入系统，系统显示订单信息列表 |
| 3 | 基本操作员在系统查询所需条件的订单信息 |
| 4 | 系统根据操作员的查询条件，查询问题列表信息并显示 |
| 5 | 系统根据结果，保存查询信息 |
| **异常事件流** | **步骤** | **活动** |
| 2a | 登录失败，系统提示登录失败原因，转第一步。 |
| 3a | 操作员的查询为空，不能显示，用例结束 |
| **其他** | 无 | |

### 3.7.2 关键业务活动图

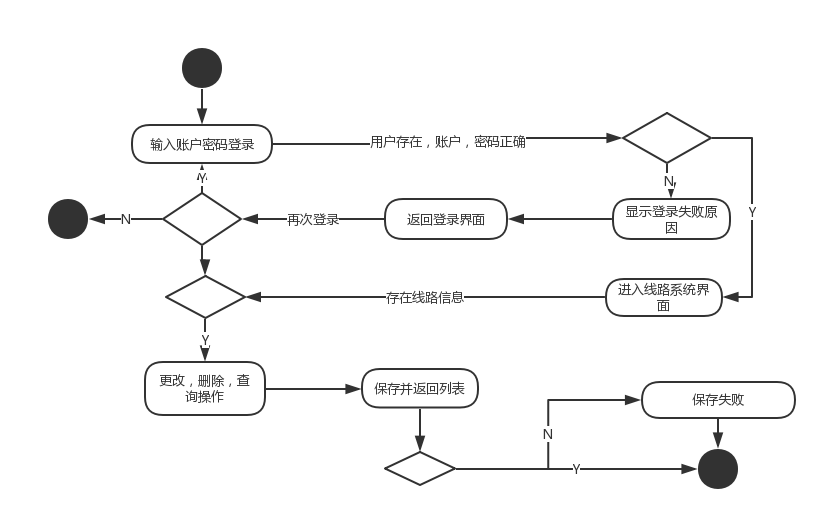
活动图阐明业务用例实现的工作流程，说明业务为向所服务的业务主角提供其所需的价值而必须完成的工作，本系统关键业务在于订单管理和线路管理。

1. 订单管理活动图，如图3-8所示



**图3-8 订单管理活动图**

1. 线路管理活动图，如图3-9所示



**图3-9 线路管理活动图**

## 3.7本章小结

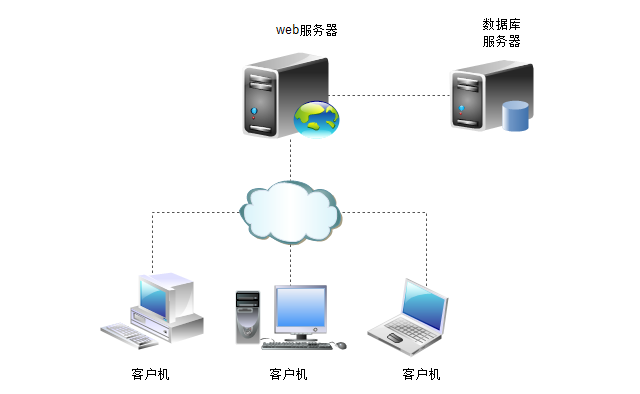
本章主要进行了系统的需求分析，主要从系统说明、总体需求、可行性分析、功能性需求、非功能性需求几个方面进行了详细的分析和阐述，并介绍了物流管理系统关键用例描述和关键业务活动图，为系统设计和开发阶段做好充分的准备。

# 第四章 系统概要设计

## **4.1 系统架构设计**

### 4.1.1 B/S体系结构

通过对托付物流管理系统的需求分析可知，系统具有数据操作量大、用户与数据库交互频繁根据这些特点，所以选择B/S体系结构。

在B/S体系结构系统中，用户通过浏览器向分布在网络上的服务器发出请求，服务器对浏览器的请求进行处理，将用户所需信息返回到浏览器，并根据前端技术进行展示。

**图4-1 B/S体系结构示意图**

与C/S两层体系架构相比，B/S三层体系架构具有较明显的优势性，主要体现在以下几个方面：

（1）部署复杂度

C/S架构客户端不是基于web的，应用逻辑集中在客户端，部署困难，安装耗时，需要每台客户端都安装客户端专用应用程序。

B/S架构客户端基于web，应用逻辑集中在服务端，客户层的功能负荷大大减弱，部署容易，而且很多应用只需安装浏览器即可。

（2）扩展难易度

C/S架构扩展更新困难，每次业务规则发生变化时，都需要不断更新或重新安装客户端代码，并重新部署客户端。

B/S架构易于扩展，每次业务规则发生变化时，只需要在应用服务器重新部署新服务器端组件，无需更新或重新安装客户端。

（3）资源耗费及相应

C/S架构每个客户端都直接连到DB服务器，为每个客户端建立连接耗费服务器资源，不适合过多的用户。

B/S架构通过中间件服务器的负载均衡和缓存数据能力，提高对客户端的响应速度。

（4）安全性

C/S架构安全性低，客户端同DB服务器直连，非法用户容易通过客户端入侵。

B/S架构安全性相对较高，中间的应用层，隔离了客户端程序对数据服务的直接访问，保护了数据安全。

### 4.1.2 系统开发环境

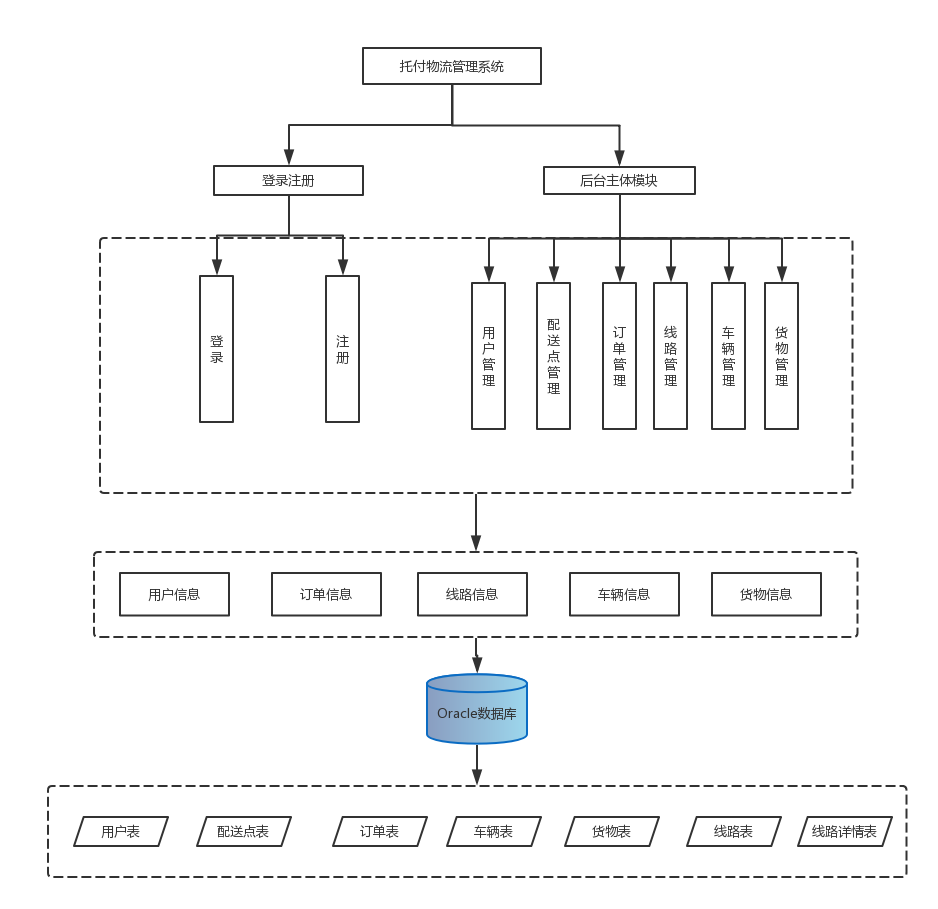
通过对实现系统技术方案的对比论证，如数据库选择Oracle，由于它具有分布式数据库它实现了分布式处理功能等优势，开发工具选择Java语言，由于它具备简单性、面向对象、分布式、解释型、可靠、安全、平台无关、可移植、高性能、多线程、动态性等特性，结合系统的开发技术，得出系统的开发环境参数配置表如表4-1所示。

表4-1 系统开发环境参数表

|  |  |
| --- | --- |
| **环境名称** | **描述** |
| 操作系统/屏幕分辨率 | Windows 10 64位操作系统，基于x64的处理器/1366\*768 |
| 安装内存（RAM）/ 硬盘 | 8.00GB / 500GB |
| 处理器 | Inter（R）Core（TM）i5-4200U CPU @ 1.60GHz 2.30GHz |
| 数据库管理工具/可视化工具  数据库设计工具  开发浏览器  应用服务器  IDE | Oracle 11g/ oracle sqldeveloper 4.0  Power designer  Google Chrome 58.0.30.29.300  Apache-tomcat-7.0.47  MyEclipse 2014 / jdk1.8.0\_91 |

## **4.2 系统结构设计**

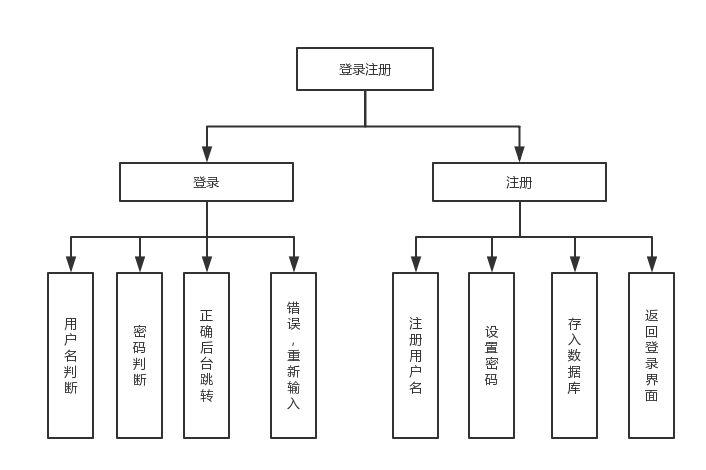
### 4.2.1 系统总体设计

本系统主要有登录注册和前台系统组成。登录注册主要包括对用户的检查和注册功能，前台主体主要是对系统各个模块进行管理，系统总体结构如图4-2所示。

**图4-2 系统总体结构图**

### 4.2.2 登录注册结构

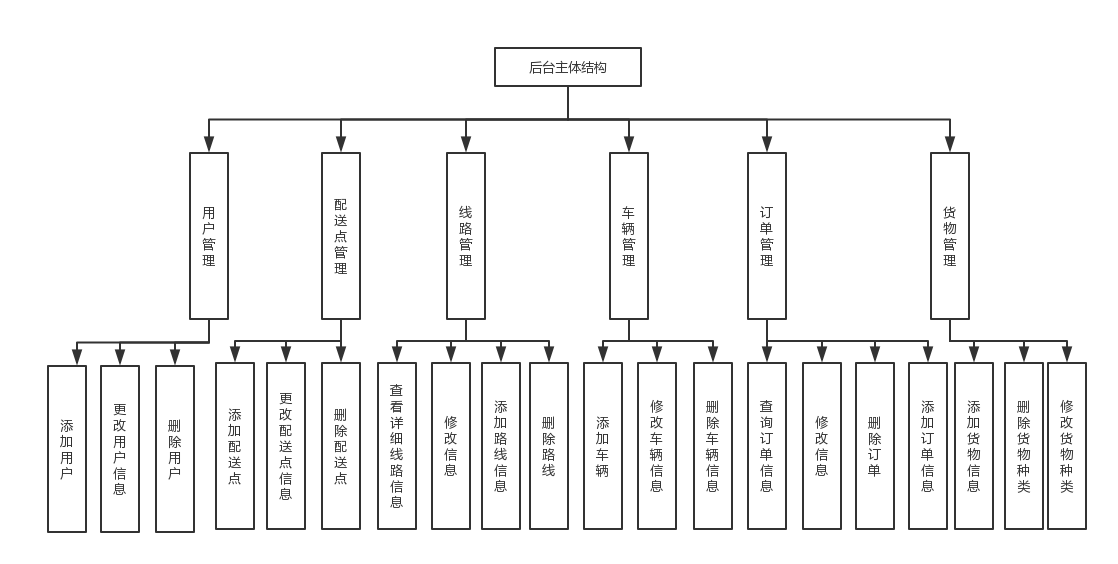
登录注册结构设计主要针对用户登录系统时进行的设置，主要功能包括登录验证、注册系统用户两个功能。这两个模块的主要设计如图4-3所示。



**图4-3 登录注册结构设计**

### 4.2.3 系统后台主体结构

本系统的后台主体设计主要针对操作员对各个模块进行管理，主要的模块有用户管理模块、配送点管理模块、线路管理模块、订单管理模块、车辆管理、货物管理。这六个模块的主要结构如图4-4所示。



**图4-4 后台主体结构设计**

### 4.2.4 系统实体类设计

根据系统需求分析和系统结构设计，本系统共设计有7个实体类，分别为用户实体类、配送点实体类、订单实体类、线路实体类、线路详细实体类、货物实体类、车辆实体类，分别为：Users、Station、Orders、Way、Way\_detil、Goodtype、Vehicle实体类之间的关系如图4-5所示。



**图4-5 实体类关系图**

## 4.3 数据库设计

对于数据库设计遵循数据库设计的三大范式，每张表之间的关联关系的正确性是数据库

效率的关键，因此，数据表关系设计也成为数据库设计的重要环节，本系统中主要有七张表，如何去管理和关联这七张表。使其能更高效的使用，对数据的存储和获取是非常的重要，，数据库建设和设计是最为重要的一环，数据库结构对系统效率影响非常大，数据库结构设计如果合理常规，将能非常大的提高数据存储的效率

### 4.3.1 数据库环境

数据库管理工具：Oracle 11g database

数据库设计与关系工具：Power Designer

数据库可视化工具：Oracle Sqldeveloper

### 4.3.2 数据库原则设计

[数据库设计](http://baike.baidu.com/view/8268.htm)(Database Design)是指对于一个给定的应用环境，构造最优的数据库模式，建立[数据库及其应用](http://baike.baidu.com/view/2233662.htm)系统，使之能够有效地[存储数据](http://baike.baidu.com/view/13997361.htm)，满足各种用户的应用需求，本系统采用关系数据库Oracle，设计遵循如下原则：

（1）规范化

在设计和操作维护数据库时，使用的数据结构正确的数据结构，以便于对数据库进行相应的存取操作，确保数据正确地分布到数据库的表。

（2）数据冗余

数据应该尽可能少地冗余，尽量减少重复数据，从而减少对多个表的冗余更新动作，确保数据的一致性。但为了便于日后维护、分析、拓展，也可适当添加如“操作用户”、“修改时间”这样的一些必要冗余字段。

（3）设计合理的表关联

为了降低表之间的直接耦合度，在多张表之间关系复杂时，可采用第三张映射表来关联维护两张表之间的关系，但当多张表涉及到大数据量的问题时，则应该尽量避免关联。

（4）主键生成策略

主键生成策略大致分为：int自增长类型、手动增长类型、手动维护类型、字符串类型。本系统的数据库设计采用int自增长类型来实现。

（5）基本表的性质

数据库表设计中，为确保数据库事务正确执行，应具备是个要素：

（1）原子性：基本表中的字段是不可再分解的。

（2）原始性：基本表中的记录是原始数据（基础数据）的记录。

（3）演绎性：由基本表与代码表中的数据，可以派生出所有的输出数据。

（4）稳定性：基本表的结构是相对稳定的，表中的记录是要长期保存的。

### 4.3.3 数据库安全设计

数据库安全性保护，就是通过种种防范措施以防止用户越权使用数据库，以防止不合法的使用造成的数据泄密、更改或破坏。数据库系统中一般采用用户标识和鉴别、存取控制、视图以及密码存储等技术进行安全控制。

1. 用户标识和鉴别

用户每次登录数据库时都要输入用户标识，数据库管理系统进行核对后，对于合法的用户获得进入系统最外层的权限。

1. 存取控制

通过用户标识鉴别的用户不一定具有数据库的使用权，需要确定每个角色对数据库表的操作权限并进行授权，确保具有数据库使用权限的用户访问数据库并进行权限范围内的操作，同时令未被授权的用户无法接近数据。

1. 视图机制

进行存取权限的控制，为不同的用户定义不同的视图，通过视图机制把保密数据对无操作权限的用户隐藏起来，从而自动地对提供数据安全保护。

1. 数据加密

数据加密是防止数据在存储和传输中失密的有效手段，对数据库系统用户帐号的密码进行加密处理。

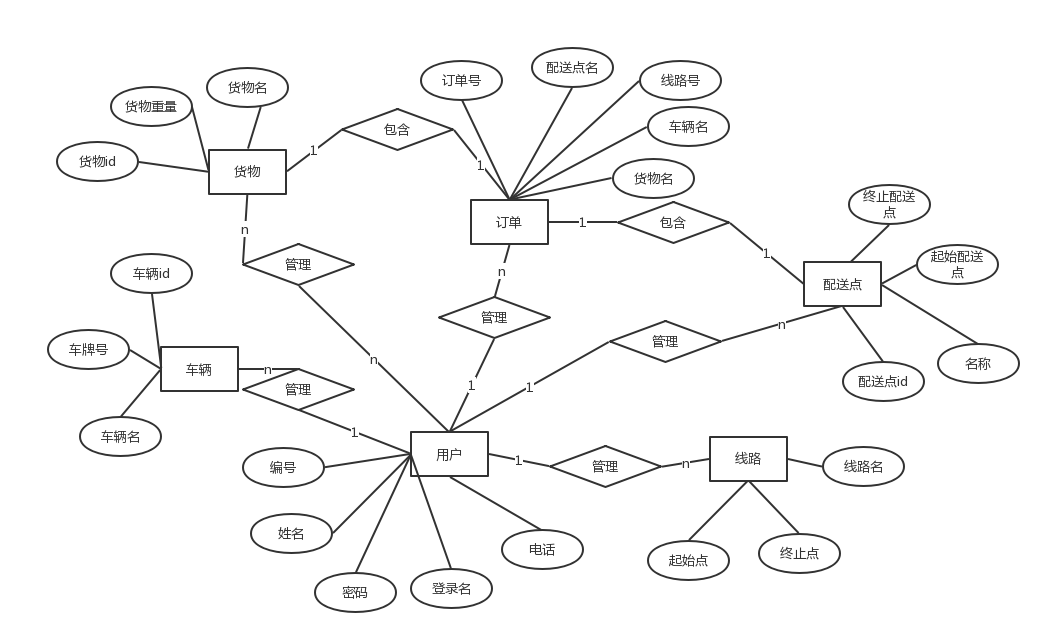
1. 审计方法

用户对数据库的所有操作自动记录下来放入审计日志中，方便一旦发生数据被非法存取时，可以利用审计跟踪的信息，重现导致数据库现有状况的一系列事件。

### 4.3.4 数据库逻辑设计

1.E-R模型

（1）根据系统的需求分析，建立概念模型，E-R关系如图4-6所示。



**图4-6 数据库表E-R图**

2.E-R模型向关系模型的转换

1. E-R模型向关系模型的转换规则（满足第三范式）：
2. 实体类型的转换：将每个实体类型转换成一个关系模式，实体的属性即为关系的属性，实体标识符即为关系的键。
3. 联系类型的转换：实体间的联系是1:1，可以在两个实体类型转换成两个关系模式中的任意一个关系模式的属性中加入另一个关系模式的键和联系类型的属性；如实体间的联系是1:N，则在N端实体类型转换成的关系模式中加入1端实体类型转换成的关系模式的键和联系类型的属性；如实体间的联系是M:N，则将联系类型也转换成关系模式，其属性为两端实体类型的键加上联系类型的属性，而键为两端实体键的组合。
4. 根据E-R模型向关系模型的转换规则，设计如下关系模式（注：主键用实线下划线表示，外键用虚线下划线表示）。

用户：（ID，姓名，联系电话，地址，用户名，密码）；

配送点：（ID，配送点姓名，配送点电话，配送点地址）；

订单：（ID，起始点ID，终止点ID，订单状态，车辆信息，货物数量，货物名称，货物ID，货物重量，货物体积，收货人地址，收货人电话，收货人邮编，寄货人地址，寄货人电话，寄货人鱼变，备注，线路ID，配送点ID，）；

线路：（ID，起始点名称，终止点名称，线路名称）；

线路明细：（ID，线路ID，途经配送点ID，顺序）；

车辆：（ID，车牌，载重，容量）；

货物：（ID，货物种类名称，备注）；

### 数据库物理设计

（1）数据库汇总表

托付物流管理系统共设计7张数据表，数据库表设计的表名及具体表功能的汇总表，如表4-2所示。

表4-2 数据库表汇总表

|  |  |
| --- | --- |
| **表名** | **功能说明** |
| User | 系统用户存储表 |
| Station | 系统配送点存储表 |
| Way | 线路信息存储表 |
| Way\_detil | 线路详细信息存储表 |
| Orders | 订单管理存储表 |
| Goodstype | 货物记录表 |
| Vehicle | 车辆信息存储表 |

（2）用户表，如表4-3所示。

**表4-3 用户表users**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 类型 | 字段大小 | 主键否 | 描述 |
| id  BELONGTO\_IDRROLE\_ID  NAME  MOBILEPHONE  ADDRESS  USERNAME  PASSWORD | int  int  int  varchar  varchar  varchar  varchar  varchar | 11  11  11  50  20  200  50  20 | 是  否  否  否  否  否  否  否 | 用户id  配送点id  角色id  名字  联系电话  住址  用户名  密码 |

（3）配送点表，如表4-4所示。

**表4-4 配送点表station**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 类型 | 字段大小 | 主键否 | 描述 |
| ID  STATIONNAME  STATIONPHONE  STATIONADDRESS | int  varchar  varchar  varchar | 11  50  20  200 | 是  否  否  否 | 配送点id  配送点名称  固定电话  地址 |

（4）线路表，如表4-5所示。

**表4-5 线路表way**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 类型 | 字段大小 | 主键否 | 描述 |
| ID  START\_STATION END\_STATION  NAME | int  varchar  varchar  varchar | 11  50  20  200 | 否  否  否  否 | 线路id  起始点  目的点  线路名称 |

（5）线路详细表，如表4-6所示。

**表4-6 线路详情表way\_detil**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 类型 | 字段大小 | 主键否 | 描述 |
| ID  WAY\_ID STATION\_ID  SORT\_ID | int  int  int  int | 11  11  11  11 | 否  否  否  否 | 主键id  线路id  途经配送点id  顺序 |

1. 货物表，如表4-7所示。

**表4-7 货物表goodstype**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 类型 | 字段大小 | 主键否 | 描述 |
| ID  GOODSTYPENAME REMARK | int  varchar  varchar | 11  50  50 | 是  否  否 | 货物种类id  货物种类名称  备注 |

1. 订单表，如表4-8所示。

**表4-8 订单表orders**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 类型 | 字段大小 | 主键否 | 描述 |
| ID  OUTSETSTATIONID DESTSTATIONID  ORDERSTATUS  VEHICLEID | int  int  int  varchar  int | 11  50  50  200  11 | 是  否  否  否  否 | 订单号  始发配送点id  目的配送点id  订单状态  承运车辆id |

**续表 4-8**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 类型 | 字段大小 | 主键否 | 描述 |
| RECEIVEDATA  GOODSNAME  GOODSNUMBER  GOODSTYPEID  GOODSTOTALWEIGHT  GOODSTOTALVOLUME  TOTALFARE  SENDERNAME  SENDERMOBILEPHONE  SENDERADDRESS  SENDERZIPCODE  ADDRESSEENAME  ADDRESSEEMOBILEPHONE  ADDRESSEEADDRESS  ADDRESSEEZIPCODE  REMARK  WAY\_ID | date  varchar  int  int  int  int  int  varchar  varchar  varchar  varchar  varchar  varchar  varchar  varchar  varchar  varchar | 50  11  11  11  11  11  50  20  200  20  50  20  20  50  20  200  20 | 否  否  否  否  否  否  否  否  否  否  否  否  否  否  否  否  否 | 收寄时间  货物名称  货物数量  货物种类id  货物总重量  货量总体积  费用  寄件人名字  寄件人电话  寄件人地址  寄件人邮编  收件人名字  收件人电话  收件人地址  收件人邮编  备注  线路ID |

1. 车辆表，如表4-9所示。

**表4-9 车辆表**Vehicle

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 类型 | 字段大小 | 主键否 | 描述 |
| ID  VEHICLELICENSE VEHICLELOAD  VEHICLECAPACITY | int  varchar  int  int | 11  50  11  11 | 是  否  否  否 | 车辆id  车牌  载重  容量 |

### 数据库维护与优化方案

（1）数据库优化方案

数据库设计阶段严格遵循数据库设计原则进行规范化设计是优化数据库的首要手段，其次优化表的数据类型，对表格进行反规范化处理，设计合理的数据库表之间的关联，添加必要的冗余字段，减少表的访问次数，使用Mybatis的缓存等方式，可以有效提高数据访问效率。

（2）数据库维护手段

对数据库进行日常的维护，包括备份系统数据、恢复数据库系统、产生用户信息表，并为信息表授权、监视系统运行状况、及时处理系统错误、周期更改用户口令等。措施。

## **4.4本章小结**

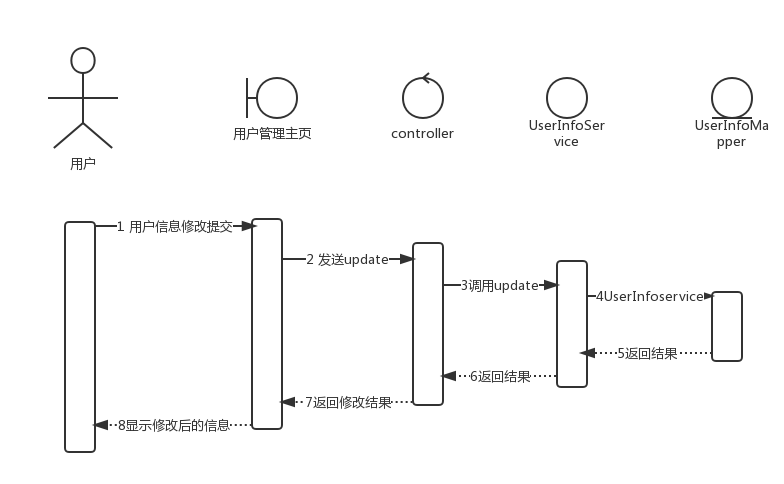
本章主要介绍了系统的概要设计，首先介绍了系统的架构设计，然后介绍了系统的结构设计，包括系统总体结构设计、登录注册结构设计和后台系统结构设计，进而设计出系统的实体关系，最后详细阐述了数据库设计，为系统的详细设计与实现做好充分的准备。

# 第五章 系统详细设计与实现

## 5.1 登录注册详细设计与实现

本系统中登录注册主要实现以下几个模块，登录模块，注册模块，记住密码。

（1）用户登录模块主要功能点为用户登录，其功能实现时序图如图5-1所示。



**图5-1 用户登录时序图**

（2）用户登录设计与实现，如表5-1所示。

**表5-1 登录设计与实现**

|  |  |
| --- | --- |
| **功能描述** | 用户通过输入用户名、密码，登录系统。 |
| **接口** | **public** String login() |
| **核心代码** | **public** String login(){  //根据前台数据进行request请求  HttpServletRequest req = ServletActionContext.*getRequest*();  //UserInfoDao ud = new UserInfoDao();  **if**(userInfoService.login(userInfo)){//判断输入信息是否正确  HttpSession session = req.getSession();  session.setAttribute("user", userInfo);  **return** "main";  }  **this**.addActionMessage("登陆成功");  **return** "login";  } |
| **补充说明** | 如果验证码、用户名、密码其中一个错误则在登录页面进行相关提示。 |

（3）用户登出设计与实现，如表5-2所示。

**表5-2 用户登出设计与实现**

|  |  |
| --- | --- |
| **功能描述** | 用户通过退出，安全退出当前账号，实现用户登出。 |
| **接口与属性** | 用户点击退出 |
| **核心代码** | /\*\*  \* 退出登录请求跳转  \* **@return**  \*/  **public** String logOut(){  HttpServletRequest req = ServletActionContext.*getRequest*();  req.getSession().removeAttribute("user");  **return** "login";  } |
| **补充说明** | 用户点击退出，跳转登录页面。 |

（4）用户注册设计与实现，如表5-3所示。

**表5-3 用户注册设计与实现**

|  |  |
| --- | --- |
| **功能描述** | 用户通过输入用户名、密码、电话、地址等信息 |
| **接口** | **public** String register() |
| **核心代码** | **public** String register(){  userInfoService.add(userInfo);  **return** list();  } |
| **补充说明** | 注册完成，添加用户信息，点击注册跳转到登录界面 |

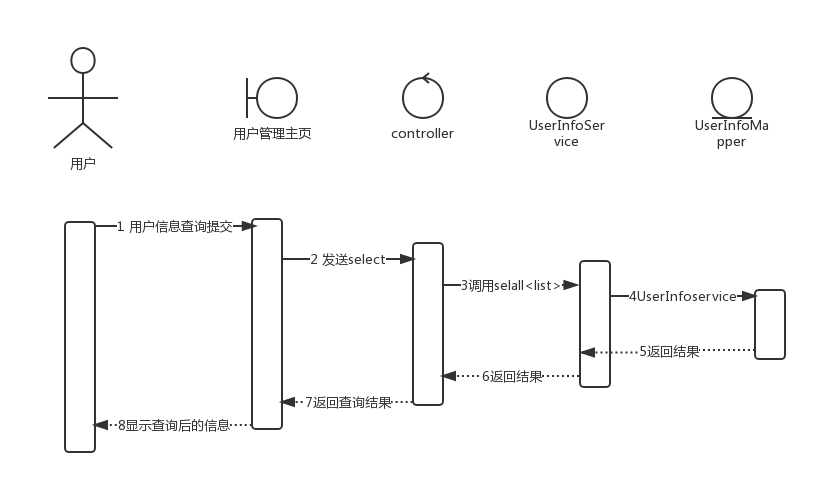
## **5.2 后台主体详细设计与实现**

本系统中后台主体主要实现以下几个模块，用户管理模块、配送点管理模块、线路管理模块、订单管理模块、货物管理模块、车辆管理模块。

### 5.2.1 用户管理详细设计与实现

本系统的用户管理模块主要功能包括用户的修改、删除、添加等功能。

1. 用户管理模块列表显示主要功能是查询，它的时序图如图5-2所示。



**图5-2 用户管理模块时序图**

1. 用户信息列表设计与实现，如表5-4所示。

**表5-4 用户信息列表设计与实现**

|  |  |
| --- | --- |
| **功能描述** | 用户通过数据库查询，打印出用户名、密码、电话、地址等信息 |
| **接口** | **public** String list() |
| **核心代码** | **public** String list(){  **this**.init(userInfoService.selCount());  List<UserInfo>list=userInfoService.selList(**this**.getCurPage());  HttpServletRequest req = ServletActionContext.*getRequest*();  System.*out*.println(list.size());  //  req.setAttribute("list", list);  **return** "list";  } |
| **补充说明** | 点击用户管理查询出用户信息列表 |

1. 用户删除设计与实现，如表5-5所示。

**表5-5 用户删除设计与实现**

|  |  |
| --- | --- |
| **功能描述** | 用户通过点击删除按钮，删除指定的用户 |
| **接口** | **public** String del() |
| **核心代码** | **public** String del(){  userInfoService.del(userInfo);  **return** list();  } |
| **补充说明** | 删除后总条数减少 |

1. 用户修改设计与实现，如表5-6所示

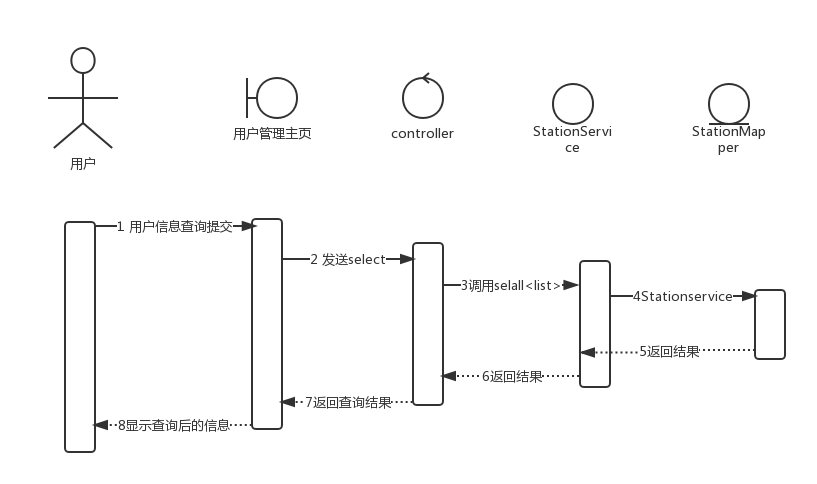
**表5-6 用户修改设计与实现**

|  |  |
| --- | --- |
| **功能描述** | 用户通过点击修改按钮，修改指定的用户 |
| **接口** | **public** String update() |
| **核心代码** | **public** String update(){  userInfoService.update(userInfo);  **return** list();  } |
| **补充说明** | 修改后，返回列表，其中修改的那条信息改变 |

### 配送点管理详细设计与实现

本系统的配送点管理模块配送点管理是由公司完成的，有配送点的浏览、添加、修改、删除等功能。

（1）系统配送点管理的核心功能就是配送点的添加，所以它的时序图如图5-3所示。

****

**图5-3 配送点管理时序图**

（2）配送点管理添加模块设计与实现，如表5-7所示。

**表5-7 配送点添加模块设计与实现**

|  |  |
| --- | --- |
| **功能描述** | 用户通过点击添加按钮，进入添加界面进行添加 |
| **接口** | **public** String add() |
| **核心代码** | **public** String add(){  stationService.add(station);  station=**null**;  **return** list();  } |
| **补充说明** | 添加后，返回列表，则列表中数据总数增加1 |

（3）配送点管理删除模块设计与实现，如表5-8所示。

**表5-8 配送点删除模块设计与实现**

|  |  |
| --- | --- |
| **功能描述** | 用户通过点击删除按钮，指定配送点被删除 |
| **接口** | **public** String del() |
| **核心代码** | **public** String del(){  stationService.del(station);  **return** list();  } |
| **补充说明** | 删除后，返回列表，则列表中数据总数减少1 |

（4）配送点管理修改模块设计与实现，如表5-9所示。

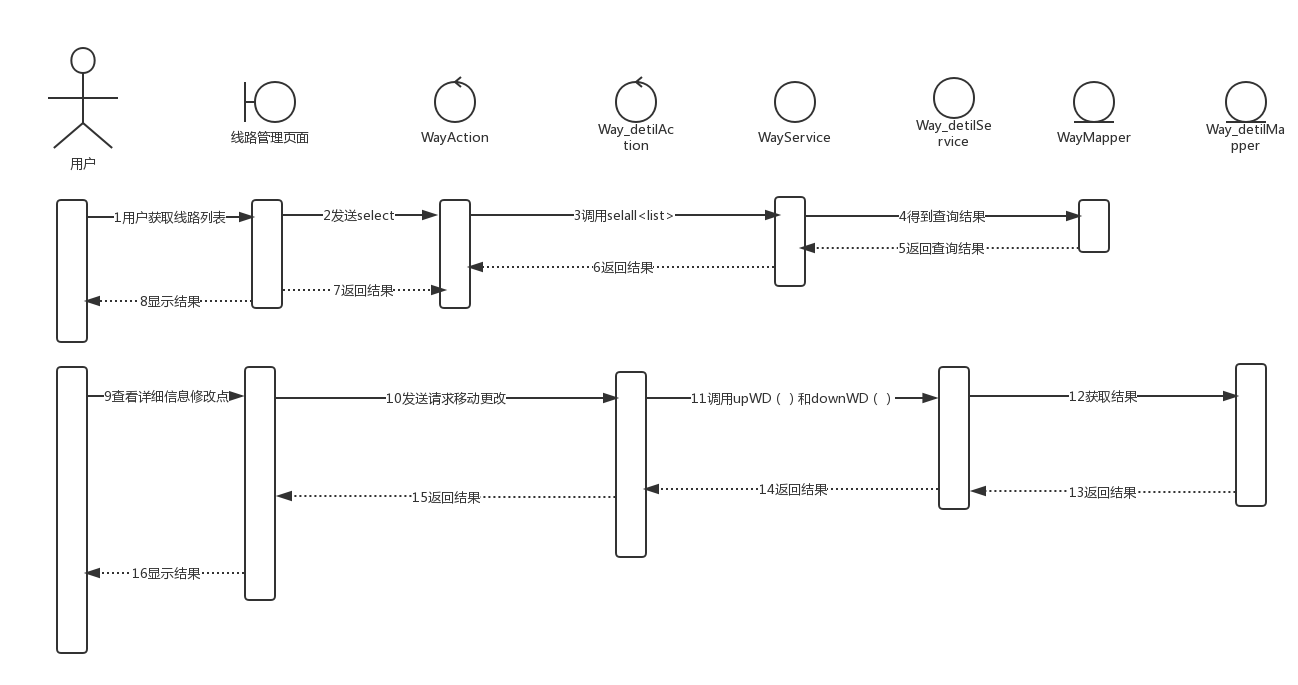
**表5-9 配送点修改模块设计与实现**

|  |  |
| --- | --- |
| **功能描述** | 用户通过点击修改按钮，指定配送点被修改 |
| **接口** | **public** String update() |
| **核心代码** | **public** String update(){  stationService.update(station);  **return** list();  } |
| **补充说明** | 修改后，返回列表，则指定的配送点发生更改 |

### 5.2.3 线路管理详细设计与实现

本系统的线路管理模块是由公司根据市场反应情况添加路线。系统选择添加路线，输入增加路线的名称，然后添加路线经过的配送点，并可以通过添加、移除、上移、下移等操作调整线路顺序。

（1）根据需求分析，线路管理的核心功能是查看线路详情，并进行线路的中配送点的更改，和添加。它的时序图如下图5-4所示。

****

**图5-4 线路管理详细信息时序图**

（2）线路管理上下移动线路点设计与实现，如表5-10所示。

**表5-10 移动线路点模块设计与实现**

|  |  |
| --- | --- |
| **功能描述** | 用户通过点击上下移动按钮，则指定中间点被移动 |
| **接口** | **public** String upWD() **public** String downWD() |
| **核心代码** | **public** String upWD(){  Way\_detail uWD=way\_detailService.selUP(way\_detail);  String sid=uWD.getSortId();  uWD.setSortId(way\_detail.getSortId());  way\_detail.setSortId(sid);  way\_detailService.update(way\_detail);  way\_detailService.update(uWD);  way\_detail = **new** Way\_detail();  uWD=**new** Way\_detail();  **return** "update";  }  **public** String downWD(){  Way\_detail uWD=way\_detailService.selDown(way\_detail);  String sid=uWD.getSortId();  uWD.setSortId(way\_detail.getSortId());  way\_detail.setSortId(sid);  way\_detailService.update(way\_detail);  way\_detailService.update(uWD);  way\_detail = **new** Way\_detail();  uWD=**new** Way\_detail();  **return** "update";  } |
| **补充说明** | 修改后，返回列表，则指定的线路中间点发生更改 |

（3）线路管理删除模块设计与实现，如表5-11所示。

**表5-11 线路管理删除设计与实现**

|  |  |
| --- | --- |
| **功能描述** | 用户通过点击删除按钮，指定线路被删除 |
| **接口** | **public** String del() |
| **核心代码** | **public** String del(){  stationService.del(Way);  **return** list();  } |
| **补充说明** | 删除后，返回列表，则列表中数据总数减少1 |

### 5.2.4 订单管理详细设计与实现

### 5.2.5 车辆管理模块设计与实现

### 5.2.6 货物管理模块设计与实现

## **5.3 界面布局设计**

## **5.4 本章小结**

# 第六章 系统测试

## **6.1 软件测试的原则和方法**

每个软件在开发过程中都会遇到不可预知的问题，在每一个开发周期当中都会残生不可回避的错误和问题，为了能够更好地消除各个方面的问题，避免以后工作中造成不必要的损失，软件在投入使用之前都将必须经过全面和规范的测试。

软件系统测试是针对整个产品系统进行的测试，目的是验证系统是否满足需求分析报告中的定义，并找到与需求规格说明书中不相符且与之矛盾的地方，从而提出更加完善的方案。系统测试发现的问题，必须根据手册进行调试和改正。

## **6.2 单元测试**

功能测试是在软件开发过程中要进行的最低级别的测试活动，在单元测试活动中，测试的独立单元必须在与程序的其他部分相隔离的情况下进行单元测试。单元测试不仅仅是作为编码是否有错误的一种辅助手段在一次性的开发过程中使用，单元测试必须相对来说是可重复性的，无论是在软件编码修改，或者是移植到新的运行环境进行二次开发过程中，都必须进行测试。因此，所有的软件测试行为都必须在整个软件系统的生命周期中进行维护。下面将列出本系统的模块测试及结果。

用户注册测试结果信息如表7-1所示：

**表7-1 注册测试信息表**

|  |  |  |
| --- | --- | --- |
| **测试信息** | **测试数据** | **测试结果** |
| 用户名、密码、确认密码  用户名、密码、确认密码  用户名、密码、确认密码 | 已存在的用户名  用户名正确，密码不一致  用户名、密码、确认密码合法 | 提示“用户名存在”，不能登录  提示“密码不一致”  提示注册成功 |

用户登录测试结果信息如表7-2所示：

**表7-2 用户登录测试信息表**

|  |  |  |
| --- | --- | --- |
| **测试信息** | **测试数据** | **测试结果** |
| 用户名、密码  用户名、密码 | 用户名正确，密码不匹配  用户名正确，密码正确 | 提示“登录失败”，返回登录界面  提示“登录正确” |

用户列表测试模块如表7-3所示：

**表7-3 用户列表测试模块**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能名称** | **输入** | **期望输出** | **测试结果** |
| 查看用户列表 | 选择查询条件，点击  【查询】 | 显示用户列表信息， | 符合期望输出 |
| 修改用户信息 | 选择列表中某用户，点击【修改】，并在修改页输入修改的内容，点击【修改】 | 系统返回到用户列表页面，反之，提示出错信息。 | 符合期望输出 |
| 删除用户 | 选择列表中的某用户，点击【删除】 | 刷新用户列表 | 符合期望输出 |
| 添加用户 | 输入添加用户账号、密码等信息，点击【保存】 | 显示保存结果，刷新用户列表。 | 符合期望输出 |

线路管理测试模块如表7-4所示：

**表7-4 线路管理测试模块**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能名称** | **输入** | **期望输出** | **测试结果** |
| 查看线路列表 | 选择查询条件，点击  【查询】 | 显示线路查询信息， | 符合期望输出 |
| 修改线路信息 | 选择列表中某条线路，点击【修改】，并在修改页输入修改的内容，点击【修改】 | 系统返回到线路信息表，反之，提示出错信息。 | 符合期望输出 |
| 删除线路 | 选择列表中的某条线路，点击【删除】 | 刷新线路列表 | 符合期望输出 |
| 添加线路 | 输入添加线路的开始和结束点，以及线路名，点击【保存】 | 显示保存结果，刷新线路列表。 | 符合期望输出 |
| 上下移动中间线路点 | 点击上移动或者下移动线路点 | 选中的线路点发生移动 | 符合期望输出 |

订单管理测试模块如表7-5所示：

**表7-5 订单管理测试模块**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能名称** | **输入** | **期望输出** | **测试结果** |
| 查看订单列表 | 选择查询条件，点击  【查询】 | 显示订单列表信息， | 符合期望输出 |
| 修改订单信息 | 选择列表中某张订单，点击【修改】，并在修改页输入修改的内容，点击【修改】 | 系统返回到订单列表页面，反之，提示出错信息。 | 符合期望输出 |
| 删除订单 | 选择列表中的某张订单，点击【删除】 | 刷新订单列表 | 符合期望输出 |
| 添加订单 | 输入添加订单的所需信息，点击【保存】 | 显示保存结果，刷新订单列表。 | 符合期望输出 |

配送单管理测试模块如表7-6所示：

**表7-6 配送点管理测试模块**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能名称** | **输入** | **期望输出** | **测试结果** |
| 查看配送点列表 | 选择配送点管理 | 页面刷新，显示配送点列表信息， | 符合期望输出 |
| 修改配送点信息 | 选择列表中个配送点，点击【修改】，并在修改页输入修改的内容，点击【修改】 | 系统返回到配送点列表页面，反之，提示出错信息。 | 符合期望输出 |
| 删除配送点 | 选择列表中的某配送点，点击【删除】 | 刷新配送点页面列表 | 符合期望输出 |
| 添加配送点 | 输入添加的配送点信息，点击【保存】 | 显示保存结果，刷新配送点列表。 | 符合期望输出 |

货物管理测试模块如表7-7所示：

**表7-7 货物管理测试模块**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能名称** | **输入** | **期望输出** | **测试结果** |
| 查看货物列表 | 选择货物管理 | 页面刷新，显示货物列表信息， | 符合期望输出 |
| 修改货物信息 | 选择列表中个货物，点击【修改】，并在修改页输入修改的内容，点击【修改】 | 系统返回到货物列表页面，反之，提示出错信息。 | 符合期望输出 |
| 删除货物 | 选择列表中的某货物，点击【删除】 | 刷新货物页面列表 | 符合期望输出 |
| 添加货物 | 输入添加的货物信息，点击【保存】 | 显示保存结果，刷新货物列表。 | 符合期望输出 |

车辆管理测试模块如表7-8所示：

**表7-8 车辆管理测试模块**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能名称** | **输入** | **期望输出** | **测试结果** |
| 查看车辆列表 | 选择车辆管理 | 页面刷新，显示车辆列表信息， | 符合期望输出 |
| 修改车辆信息 | 选择列表中车辆信息，点击【修改】，并在修改页输入修改的内容，点击【修改】 | 系统返回到车辆列表页面，反之，提示出错信息。 | 符合期望输出 |
| 删除车辆 | 选择列表中的某车辆，点击【删除】 | 刷新车辆页面列表 | 符合期望输出 |
| 添加车辆 | 输入添加的车辆信息，点击【保存】 | 显示保存结果，刷新车辆列表。 | 符合期望输出 |

## **6.3 整体测试**

完成基本的单元测试后，系统测试将真正的进入部署测试环节，主要的步骤有以下几步：

1. 首先，对系统进行部署，添加到web服务器中，本系统使用的是Tomcat7.0.
2. 之后，启动web服务器。
3. 最后，在浏览器中输入访问地址，访问系统，进行系统的各个功能的测试，检验系统是否符合需求分析报告的妖气，经过多轮的实验和反复验证，发现并记录系统中存在的问题和问题并及时修改与完善，最终实现系统的完美交付。

## **6.4 本章小结**

# 第七章 结论

经过从选题到毕业设计代码书写完成，再到毕业论文定稿，每一个环节都在用心去感受，毕业设计的完成也就标志着我们大学生活即将结束，这个管理系统用了差不多两个半月的时间，在这其中的快乐与辛苦可能只有自己能体会，一路过来学会了很多也感触良多。

首先，在一个系统的开发初期，需要一个完善的需求分析，设计数据库和设计每个模块应该具备的功能，然后建立模块之间的整体架构，实现每个模块的基本功能。当基本功能完成之后，接下来就要设计界面，调试每个色素、距离，优化每个细节。再不断的完善一些基本功能，最终完成整个物流系统的开发设计。因为时间短，任务重。只有学会合理分配才能更高效的完成需求。所以也在期间学习到了系统开发流程以及如何更高效的完成一个任务的能力。

其次，就是在具体的程序实现环节，编写代码实现相关功能时，当问题出现时，如何去处理。本次物流系统开发，Java类中学会了如何设置断点调试，因为以前基本不愿意去用设置断点。前端开发中利用JavaScript中的alert输出信息来查找可能的错误，总之，解决问的方法比以前更加丰富了，在一个系统的开发中要学会代码的优化。在错误检查阶段，需要的可能是耐心和细心。

毕业设计对我们毕业生来说不仅仅是一个任务，可能是我们人生中不可或缺的一次经历和磨练，它能使我们获得了只是面的扩充，基本技能的提升，更重要的可能的就是积累了经验。我相信毕业后工作后，这次的毕业设计将会影响我的每次开发和每行代码的书写。

# 参考文献

1. 霍斯特曼.Java核心技术 卷一 基础知识，机械工业出版社，2007
2. Bruce Eckel.Thinking in Java（第四版），机械工业出版社，2007.
3. 刘晓华.JSP应用开发详解，电子工业出版社，2007
4. 何明/何倩颖.Oracle快速Web应用开发-从实践中学习Oracle Application Express，电子工业出版社，2010
5. 孙卫琴.精通Struts:基于MVC的Java Web设计与开发，电子工业出版社， 2004
6. 陈夫真.基于SSM的某高校教室管理信息系统的设计与实现[D].江苏省：苏州大学,2012.
7. 宋雨/赵文清.软件工程，电子出版社，2006
8. 孙卫琴.Tomcat与Java Web开发技术详解[M]. 第2版. 电子工业出版社, 2009
9. 刘腾红，孙细明.信息系统分析与设计[M]. 北京：科学出版社，2003年
10. 单东林/张晓菲.锋利的jQuery[M].魏然人民邮电出版社 2009-6
11. (美)David.Flanagan. JavaScript权威指南(第六版)[J].机械工业出版社 2012.3
12. 张洪伟.JSP网络开发技术与整合应用，清华大学出版社，2006
13. 李海生、郭跃.软件测试技术案例教程[M].清华大学出版社，2012年4月第一版
14. 黄平主编.数据库技术.北京：机械工业出版社，2002.

# 致谢

时光荏苒，岁月如梭。转眼间四年大学时光就要过去，我就要对陪伴我四年的母校说再见了。在有限的大学时光中，我收获良多。不仅掌握了很多的科学文化知识，还懂得了许多做人的准则和规则。首先要感谢我的指导老师——杨雷老师，他对事物的精益求精，勤奋好学，热爱探索的精神深深的激励和感染着我，在实践学习过程中，让我的专业能力不断提高。在他的帮助下，我克服了很多学习中的困难，在我一步步完成毕业设计的几个月里，杨雷老师的严格知道对我的毕业设计有很大的帮助，不至于我迷失方向。我跟杨老师学到的并不仅仅是这个物流管理系统如何去实现，更重要的将是在今后的工作中当我遇到不会的问题，我如何去克服。毕业设计对我来说不再是一份作业。最后我会带着老师们这四年里对我的所有教诲走向我人生的下一个阶段，用我最好的状态去面对未来更大、更强的挑战。

其次我想去感谢我在毕业设计过程中帮助和支持我的老师和朋友们，谢谢你们对我的支持和关心，让我在四年的学习生涯中干道充实和快乐，能够向着自己既定的目标前进。