

西安邮电大学

毕业设计（论文）

题目： 基于负载均衡的商城系统

学院： 自动化学院

专业： 自动化

班级： 自动 1403 班

学生姓名： 谭友钢

学号： 06141087

导师姓名： 程伟 职称： 讲师

起止时间： 2017 年 12 月 5 日至 2018 年 6 月 10 日

毕业设计（论文）声明书

本人所提交的毕业设计（论文）《基于负载均衡的商城系统》是本人在指导教师指导下独立研究、钻研、写作的成果，毕业设计（论文）中所引用他人的文献、数据、图片、资料、代码块等均已明确标注，且在毕业设计（论文）中未抄袭他人学术观点。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式注明并表示感谢。

本人完全明白《西安邮电大学本科毕业设计（论文）管理办法》的各项规定并自愿遵守。

本人深知本声明书的法律责任，违规后果由本人承担。

论文作者签名：

日期： 年 月 日

西安邮电大学本科毕业设计(论文)选题审批表

申 报 人	程伟	职 称	讲师	学 院	自动化		
题目名称	基于负载均衡的商城系统						
题目来源					教学		其它
题目类型	硬件设计		软件设计		论文		艺术作品
题目性质	应用研究				理论研究		
题目简述							
对学生知识与能力要求							
具体任务以及预期目标							
时间进度							
系（教研室）主任 签字	2017 年 12 月 9 日			主管院长 签字	2017 年 12 月 9 日		

西安邮电大学本科毕业设计（论文）开题报告

学生姓名	谭友钢	学号	06141087	专业班级	自动 1403
指导教师	程伟	题目	基于负载均衡的商城系统		

选题目的

近年来，随着人们思想以及生活水平的提高，计算机已成为人们生活中所不可或缺的一部分。随着计算机和网络的普及，越来越多的商家在网上开店，向消费者展示出另一种购物方式，网上购物成为了一种新的购物流念，人们足不出户便可以购买所需的商品。为了迎合市场需求，越来越多的网上商城应运而生。网上交易已成为最时尚，最快捷的购物方式。与传统零售首相比，网上购物具有许多的优点：成本低，方便快捷等，但与此同时也有许多的问题，比如说安全性、稳定性、高效性。鉴于对互联网发展前景以及优缺点的分析，我决定以“基于负载均衡的商城系统”来作为我的毕业设计主题。本课题在实现购物功能的前提下将商城的稳定性以及高效性做出进一步优化。

前期基础

已学课程：《计算机控制技术》、《C 语言程序设计》、《大学计算机基础》《计算机网络》；

掌握的工具：IDEA；

资料积累：《疯狂 java 讲义（第 3 版）》（李刚 电子工业出版社）

《java EE 互联网轻量级框架整合开发》（杨开振 电子工业出版社）

《java 并发编程实战》（Brian Goetz 机械工业出版社）

《实用负载均衡技术》（Peter Membrey 人民邮电出版社）

软硬件条件： 软件条件： IDEA 开发工具、Mysql 数据库、tomcat 服务器、maven 项目构建工具、java 编程语言。

硬件条件：笔记本电脑

要研究和解决的问题

要完成“基于负载均衡的商城系统”的设计，我计划分为以下几个模块，以便能够有条理的完成： 1. 主页：展示网站的基本功能，展示商品以及搜索 2. 登录注册模块：完成用户的登录注册 3. 个人信息管理模块：修改个人信息以及查看个人信息 4. 购物车管理模块：加入商品进入购物车，方便用户继续购买及浏览 5. 管理员管理模块：管理商品信息，订单信息等 6. 订单管理模块：方便用户查看购买的订单以及订单详细信息等。

工作思路和方案

在开始本课题前，复习以前学过的知识，建立起网站开发的基本架构，分模块完成开发。

第一周：复习 java 编程的基础知识，查阅相关资料。

第二周至第三周：整理项目构思，分好模块，并完成开题报告。

第四周至第六周：编写程序，逐步完成各个模块功能。

第七到第九周：对完成的项目进行测试以及 bug 处理。

第十周至第十一周：总结出现的问题以及对问题的解决办法，并开始撰写论文。

第十二周至十四周：完成论文撰写并准备答辩。

指导教师意见

签字：

2018 年 1 月 9 日

目 录

第一章 绪论	1
1.1 课题研究背景.....	1
1.2 国内外现状与发展趋势.....	1
1.3 研究的意义.....	1
第二章 相关技术简介	3
2.1 Java 介绍	3
2.2 JSP 介绍	3
2.2 MySQL 数据库介绍	4
2.2 SSM 框架的简介	5
第三章 需求分析	6
3.1 系统设计目标.....	6
3.2 系统可行性分析.....	6
3.2.1 操作可行性分析.....	6
3.2.2 经济的可行性分析.....	6
3.2.3 技术可行性分析.....	7
3.3 系统功能需求分析.....	7
3.4 系统流程分析.....	8
3.4.1 前台业务流程分析.....	8
3.4.2 后台业务流程分析.....	8
3.4.3 用户模块流程.....	8
3.4.4 管理员模块流程.....	9
3.5 界面需求分析.....	9
第四章 系统设计	11
4.1 模块功能设计.....	11
4.2 数据库的设计.....	12
4.2.1 数据库设计原则.....	12
4.2.2 数据库实体.....	12
4.2.3 创建表数据.....	15
第五章 系统实现	17
5.1 登录页面.....	17
5.2 首页.....	17
5.3 注册.....	19

5.4 查看商品.....	20
5.5 查看订单.....	21
5.6 评价.....	23
5.7 管理个人信息.....	24
5.8 管理员界面.....	25
5.9 管理员用户管理界面.....	26
5.10 管理员订单管理.....	27
5.11 管理员公告管理.....	28
结束语.....	30
致 谢.....	31
参考文献.....	32

摘 要

随着互联网技术和信息技术的发展,以电子数据交换为基础的电子商务得到了越来越广泛的应用。而网上购物也越来越为人们所关注。网上购物给用户提供方便的购买途径,只要简单的网络操作,并具有完善的售后服务。同时,在像当当网这样的地方购买商品,都能实现送货上门,货到付款,使网上购物的安全性得到了保障。随着网络购物安全性和售后服务的加强,网上购物系统的发展对提高人们生活水平方面具有十分重要的意义。

商城系统是针对网络购物的日常管理而设计,操作方便而且界面简洁。本系统基于 SSM 框架,使用 Java 语言,采用 JSP 技术,结合 MYSQL 数据库来创建数据库和程序跟踪与管理信息。本系统的设计过程完全按照软件工程的要求开发的,主要分为以下几个步骤:首先进行可行性研究;第二步进行详细调查,需求分析;第三步进行系统总体结构设计,完成系统所需的数据库设计;第四步进行系统详细设计,最后进行软件编码和系统测试。这个网上购物系统能帮助用户对前台网站进行日常管理和信息发布;并具有占用系统资源少、信息量大、站点维护方便、便于扩充和更新、易于继承和保护历史数据等优点,同时此系统可以克服传统销售中地域、广告宣传、人力资源不足等限制,能很好地适应网上销售需求。

关键词: JSP, Java, Mysql, SSM 框架, 商城系统

Abstract

With the development of Internet technology and information technology, electronic commerce based on electronic data exchange has been more and more widely used. And online shopping is getting more and more attention for people, which provides users with a convenient way to purchase and a perfect after-sales service. At the same time, the purchase of goods in places like Dangdang.com can achieve delivery, cash on delivery, and ensure the security of online shopping. With the strengthening of online shopping security and after-sales service, the development of online shopping system is of great significance to improving people's living standards.

The electronic cloud shopping mall system is designed for the daily management of online shopping. It has many good properties, such as convenient operation and simple interface. Based on the SSM framework, Java language and JSP technology, this system combines with MySQL database in order to create database and program tracking and management information. The design process of this system is completely developed in accordance with the requirements of software engineering. It is mainly divided into the following steps: first, feasibility study; second, detailed investigation and requirement analysis; third, overall system structural design and completion of the system; the fourth step is to carry out detailed system design, and finally perform software coding and system testing. This online shopping system can help users manage the day-to-day management and information distribution of the front-end website. It also has the advantages of occupying less system resources, large amount of information, easy maintenance of the site, easy expansion and updating, and easy inheritance and protection of historical data. At the same time, the system can overcome restrictions on geographical distribution, advertising, and human resources in traditional sales, which can well adapt to online sales needs.

Key words: Java, JSP, Mysql, SSM framework, Electronic Cloud Mall

第一章 绪论

1.1 课题研究背景

自从计算机被研究出来以后，人们的生活改变了许多，计算机的发展速度也是十分快的。机器逐渐取代人工，各大企业也是以电脑工作为主。而电子商务在这种势头下开始很快的发展，网络购物在人们的生活中占有的比重是越来越高，而我国的各大互联网公司发展迅猛，同时对于我国的计算机发展速度的提升也是十分大的。网络购物在方便的同时安全性也十分的高，这种购物方式是十分适合现代快节奏的社会的。

1.2 国内外现状与发展趋势

目前的全世界互联网产业发展迅速，在世界五大互联网公司中美国的公司占了4家，中国占了一家。要知道我国的互联网发展时间是很短的，而美国比我国领先几十年的时间。从这点可以看出来，网络时代已经到来，在国外网上购物是十分普遍的。亚马逊能够跻身于世界互联网前五，与它的发展模式与前瞻性与时代的发展是十分吻合的。研究表明：现在的美中两国等国家十个人中七个人有过网购经历，这是十分可怕的一个数字。要知道现在的互联网巨头中美占了大半。中国的淘宝，京东以及苏宁易购等，美国的亚马逊，苹果等企业承担了中美的大多数网购需求。而马云也由一个创业者在短短十几年时间里成为了中国首富，并且在世界范围内也有很深的影晌。这说明网络购物对于时代的发展潮流是顺势的，表明了国人对于网络购物接受程度也是十分快的，所以我认为这种势头下开发一个商务级别的购物网站是符合时代发展的，并且具有非常好的前景。

1.3 研究的意义

在计算机被研究出来以后，注定了以后的发展是机器占据很大比重的，计算机控制机器来进行生产，工作也是十分正常的。而电子商务这个概念也是近些年才开始出来的，这种观念彻底改变了人们以往只能去超市消费的生活方式，并且让人们们的观念逐渐改变，现代生活节奏的加快，使得人们的逛实体店的时间也慢慢减少，而网络购物在这种条件下的产生是十分符合人们的生活的，使得人们的购物方式也有了很多样的选择。网络购物对于实体店的冲击是巨大的，但是对于眼光前瞻的人们也提供了更多的机会。在最近几年通过淘宝，京东等平台中实现人生目标，走向成功的例子也不少。而且电子商务购物商城的出现也解决了中国的就业岗位丰富性，极大地带动了快递，运输等行业的发展，对于我国的发展重要性是很大的。网上购物的优点与实体店比较也是十分巨大的，一是它并不用占据地段，减少了租金，提高了商家的收入。二是它减少了实体店商家的日常消耗，降低了水电费等日常消耗，且减少了实体销售的环节，使得效率与利润

增高。三是它的危险低，在实体销售中租金高，投入高，使得抽身较困难，但是网络销售这种情况并不多见，它是用户先下单，然后联系厂家生产，再发货，自己的存货并不是很多，投入不是很大，抽身也比较简单。购物方式的多样性对于卖家与买家都是十分有利的，方便人们的生活，这种购物方式是十分符合时代的发展的，它的前景是十分美好的。

第二章 相关技术简介

2.1 Java 介绍

Java 是当今最流行、有很大影响力，受欢迎程度很高的高级编程类语言之一，面向对象是 java 语言的重要特点之一，其主要是是面向底层的地方很少，可以只编译一次，然后多次运行。java 语言能够跨平台操作的主要点和关键点是 JVM（java 虚拟机），因其平台不同，所以它们的 JVM 也是不同的。在 java 中有着字节码，因为它不直接和其他平台之间进行交互，而是跟不同平台中的不同 JVM 进行交互。JVM 可以加载不同的字节码，讲字节码编译成不同机器认识的机器码，用来让不一样的平台来操作执行。

程序员在编写玩完成代码后，将自己的源码通过机器中的编译工具编译成字节码。再通过将字节码给加载到 JVM 中来在机器上操作，如下图 2.1 所示。

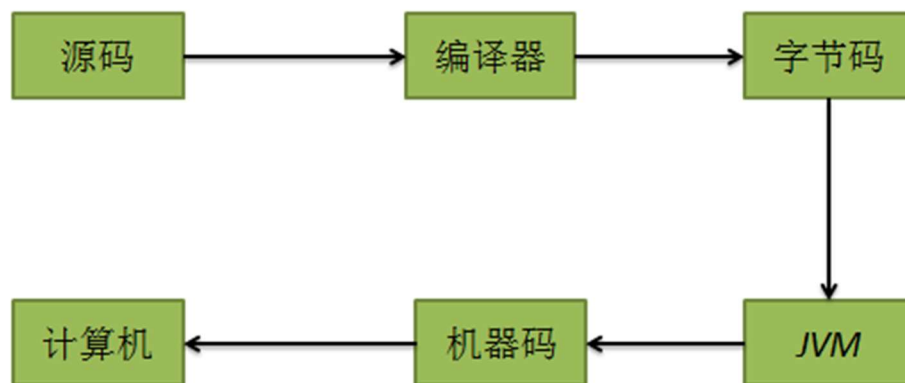


图 2.1 java 程序编译图

面向对象中存在四个主要的特点：抽象、封装、多态和继承。

封装：封装讲现实生活中的事物进行抽象成类。

抽象：抽象是把不同的数据进行抽象化表示。

继承：子类 extend 继承父类之中存在的方法和父类之中的存在的实例变量，叫做继承。

多态：不同类对象对应不同的信息，多对一这种叫做多态。

2.2 JSP 介绍

JSP（Java Server Pages）是一种可以来表示前端页面的语言，它逐渐替代了 html 这种格式，是由 java 的缔造公司 sun 公司研究出来的。这是嵌入 Java 代码的 HTML 文件从本质上来讲。在引入 JSP 之前，在 WEB 文件存在的应用程序中，一切的业务逻辑和 HTML 的响应都是在 Servlet 之中来实现，这种方式存在可读性差，易出错，可维护性差的缺点。

JSP 存在下面的特征：

（1）JSP 可以运行在多个系统以及多个平台，因为 JSP 这种前端技术是基于 Java 语言来编写的，适合于软件中的分发功能。

（2）JSP 是 java 开发人员为页面开发简便而开发出来的，毕竟不是所有的 web 开发人员都熟练掌握脚本语言的编写，java 开发人员为了方便调取，将许多功能封装在 JSP 里面。

（3）JSP 是最为强调的一点是可复用性，许多的 JSP 页面都是可重用的以来性质在页面中。

（4）因为 JSP 技术是由 java 产生的，所以它也具有 java 的跨平台操作的特征，一次加载可以多次运行。

（5）JSP 原则上是为 java 语言更方便而开发出来的，它被编译成 java Servlet。这种技术的存储功能是十分优秀的，当然，它的安全性也是十分高的。

（6）JSP 技术与 java 相像的地方十分多，可移植性是一大特点。

JSP 技术跟 java 语言是相通的，它是为 java 开发简便而产生的，它能很好地将业务中的逻辑和视图的实现进行分离，来让的系统的扩展性增高。

2.3 MySQL 数据库介绍

MySQL 是一款由瑞典 MySQL AB 公司开发的一款关系型数据库，在 2013 年，当时 Oracle 公司为了获得 mysql 的控制权，收购了 sun 公司，mysql 是主流关系型数据库，是各大公司数据库开发所运用的主流数据库，其在 WEB 开发方面是最好用的数据库。

Mysql 相比于 oracle 是开放源码的，而且 mysql 数据库体积小，速度快，很受各大公司的青睐，为减轻成本，各大公司纷纷改用 mysql 数据库，这使得 mysql 的使用率大大的到提升，mysql 最主要的是现在适应各类操作系统，同时可以与 php 完美契合。

MySQL 的特点有：

1.Mysql 的基本组成的语言是 C 和 C++，因此它是具有可转移性质的。

2.Mysql 可以适应各类操作系统，经过多次测试时十分稳定的，并且它还支持多种语言，有 java，python 等主流语言以及 uby 和 Tcl 等语言。

3.mysql 能够被许多公司使用的一大前提是它是支持多线程的。

4.mysql 可以根据项目中的要求进行深度定制满足需要的。

5.mysql 经常在开发中是使用 JDBC 来使得代码跟数据库连接的。

6.mysql 可以自己优化数据库，许多的公司都是自己将数据库优化为最佳，mysql 提供这种管理的工具。

7.Mysql 的数据处理能力是十分快的且优秀的，可以处理成千上万的数据量且能保证速度。

8.Mysql 是开源并且是免费的。

2.4 SSM 框架的简介

自从 Spring MVC 问世以来,互联网开发的主流框架就由 SSH 之后改为了 SSM (Spring + Spring MVC + MyBatis) 框架,这款框架时在当前来说是应用最多的一款企业级别框架,各种企业级的应用系统都可以搭建这款框架。

Spring 是 java 开发中的主流框架,从以前的 Struts2+hibernate+Spring 到现在的 Spring+mybatis+Spring MVC,其中都是跟 Spring 框架有关联的,开发者在设计过程中,是可以选择全用或者部分用的, Spring 框架是按照模块化设计的,跟开发分模块的过程一样,每个模块独立运行且各自完成自己的工作, Spring 主要的特点是 IOC 以来注入与 AOP 面向切面编程, Spring 框架是一款开发很完美的 J2EE 框架。

Spring MVC 是 Spring 后又一让人惊艳的框架,它支持 Spring 的特征且与 Spring 完美契合,它可以在 Spring 框架中插入 MVC 设计模式,它的出现让开发过程变得更加的规范以及更加简便, Spring MVC 使得 DispatcherServlet 前端控制器, HandlerAdapter 处理器适配器, 视图解析器 ViewResolver 分离,这种分离使得它们可以按顺序来执行。

Mybatis 我们以前称它为 iBatis,它也是也是 apache 公司的一款开源性质的框架。它是一个持久层的框架,它使得数据库中的 SQL 可以更快的运行,这点是 hibernate 所不具备的。Mybatis 特点是易学,程序员可直接编写 SQL 语句,可以更加使得 SQL 语句优化性能更高,使得开发效率更高。现在的各大企业功能需求是经常变更或者优化的,这点非常适合 mybatis 框架,可以方便开发人员及时修改 SQL 语句来变更需求,所以这也是 mybatis 框架逐渐取代 hibernate 的原因。

第三章 需求分析

3.1 系统设计目标

需求分析是指通过各种前期调研来确定开发的商城所需要实现的功能以及用户的各种需求，然后再根据这种需求来开发出使得用户满意度高，满足需求的系统，与此同时也要注重系统的可扩展性能。

在进入计算机时代以来，人们的生活能够多样化的一个重大原因是网络，而且在现在人们的购物需求也越来越多，为了让的顾客的购物体验越来越好，使得各个公司要将各种需求能够与互联网联系起来，让的商城系统各种功能更加完善，开发出让顾客所满意的购物商城是商城系统的主要任务。

在设计和实施这个 JSP 购物商城系统管理系统时，应该遵循以下设计目标进行开发：

（1）信息的安全性。在现在的互联网时代，人们在网上支付时安全性是最重要的，只有保证了财产以及信息的安全性，才是最主要的部分。

（2）性能的先进性。根据用户的需求以及运用现在流行的技术，开发出用户满意度高，符合用户需求的商城系统。

（3）功能的扩展性。在商城系统的设计中，我们要在开发中留足够的接口来使得功能具有可扩展性，使得应用可以更新、升级。

（4）功能的有效性。这个是指商城系统设计中要充分调查用户的需求，使得系统可以提高用户满意度，满足用户各种需求。

3.2 系统可行性分析

系统的可行性分析要满足的条件是十分多的，在开发设计之前我们要规划好时间以及所要损耗的人力以及物力，尽量减少开发过程中的突发事件，让的项目能够较为流畅的进行，减少无用功以及不必要的损耗。因此，我们可以来对于这个项目的运行与否分析如下列：

3.2.1 操作可行性分析

操作可行性是指在开发完成后操作方面的可行性，对于基于负载均衡的商城系统来说，会简单操作电脑的人是可以独立完成操作的。只是在浏览器中输入网址然后根据逻辑提示完成各个操作，没有很复杂的操作流程，所以操作可行性是容易的。

3.2.2 经济的可行性分析

经济可行性是指在开发的过程中所产生的经济效益问题，此商城系统在设计以及开发过程中是本人独立开发的，只需要开发用的电脑以及免费软件，所以

耗费的经费是几乎没有的，而且在开发完成各种功能时，还可以产生部分收益，因此在经济的可行性方面是行得通的。

3.2.3 技术可行性分析

（1）硬件可行性分析

在开发所需的硬件方面，是十分简单的，需要硬件配置较为适中的配置就可以，保证系统正常工作，不至于使得系统崩溃，提高开发效率以及访问效率。而低配置的电脑会降低开发效率，使得开发进程十分缓慢，导致网站运行不流畅。所以总的来说，以目前的个人计算机基本配置来说，硬件的问题是完全可以解决的，因此，硬件这些这样分析也是可以的。

（2）软件可行性分析

软件可行性分析是指开发过程中所运用的软件分析，Mysql、Tomcat、myeclipse、jdk 等日常软件都可以在网上下载并配有安装教程，而在 java 开发中的代码复用以及运用 jsp 开发前端页面是完全没有问题的。因此，软件这些这样解析也是可以的。

我们从上面几个方面的可行性进行分析，可以看出此项目基于负载均衡的商城系统在开发可行性方面是没有问题的。

3.3 系统功能需求分析

需求分析是指通过各种前期调研来确定开发的商城所需要实现的功能以及用户的各种需求，然后再根据这种需求来开发出使得用户满意度高，满足需求的系统，与此同时也要注重系统的可扩展性能。

此商城是一个基于 B/S 架构的系统，采用了 mysql+ssm+tomcat+jsp 等技术，实现一个在线购物商城系统。这我开发的商城系统中，分为用户前台这一部分以及管理员后台这一部分，用户前台这部分负责实现的有：

1. 用户第一件事做的就是登陆，但是如果你没有进行注册的话你是不能登陆成功的。在登陆的时候遇到了忘记密码问题还可以进行修改密码。
2. 用户在前台页面登录进去以后也可以在主页里浏览要购买的商品信息并确定是否购买。
3. 用户在购买商品后可以付商品满意度进行评价以及查看评价。

用户后台这部分负责实现的有：

1. 对前台注册的用户的信息进行管理。
2. 实现商品的上架、商品删除等功能。
3. 对用户的评价信息进行管理。
4. 对前台的公告信息进行管理。

3.4 系统流程分析

3.4.1 前台业务流程分析

在商城的前台页面中主要有这几个功能要实现：实现用户的登录注册，查看商城里面商品的详细信息，也可以查看用户购买完商品的支付与否，以及用户的订单信息，用户购买完的评价等部分，用户前台所展现出来功能的流程图的如下图所示：

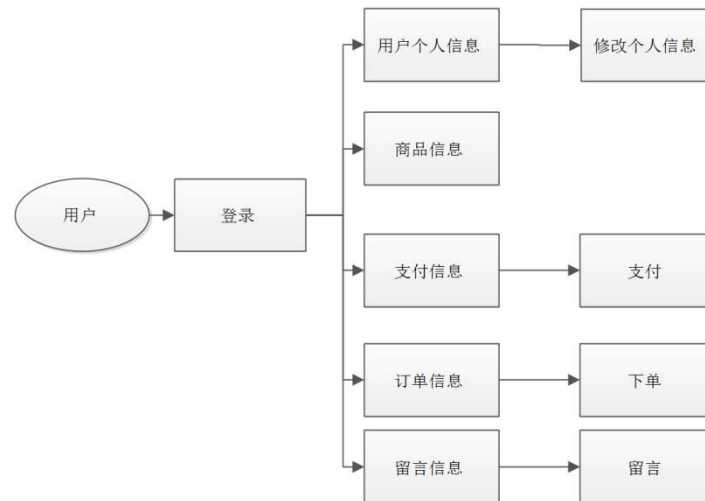


图 3.1 前台业务流程图

3.4.2 后台业务流程分析

在商城的后端管理员页面也由下面几个功能要来实现：管理员查看注册用户的详细信息，管理员查看已购买的订单，管理员登录后台系统，管理员对于商品进行管理。管理员后台要展现出来要点要根据下图所示：

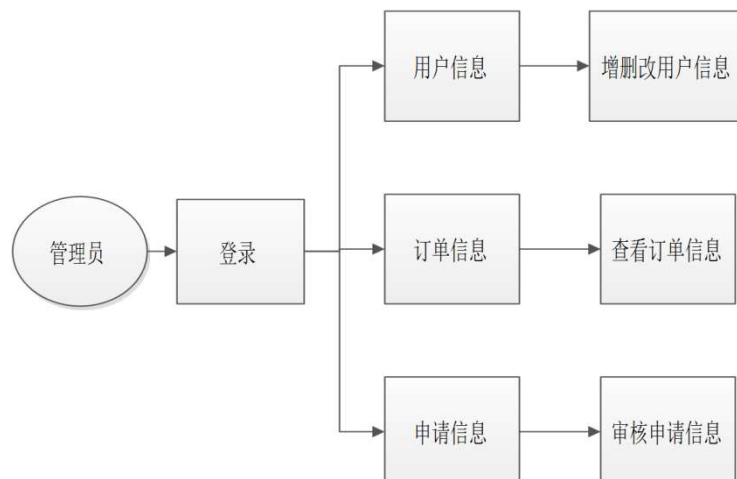


图 3.2 后台业务流程图

3.4.3 用户模块流程

用户在整个商城中的流程图如下所示：

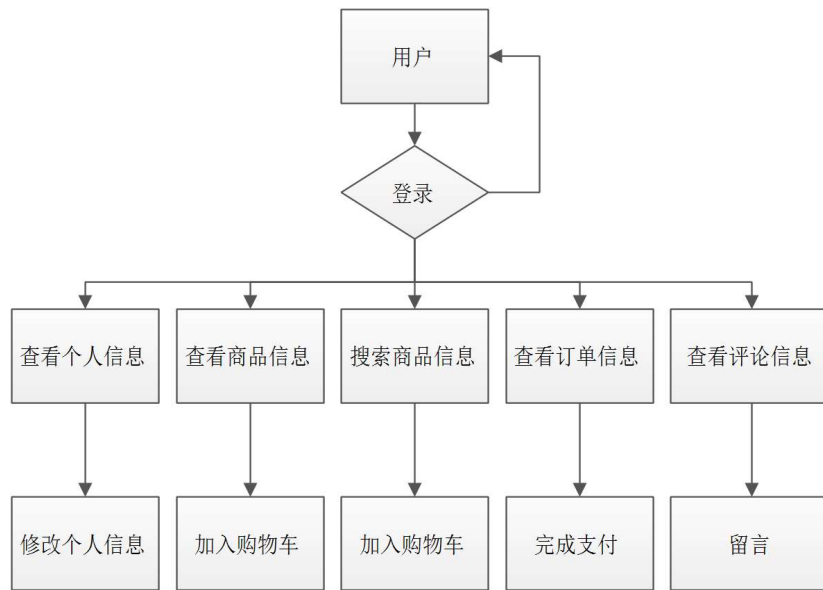


图 3.3 用户模块流程图

3.4.4 管理员模块流程

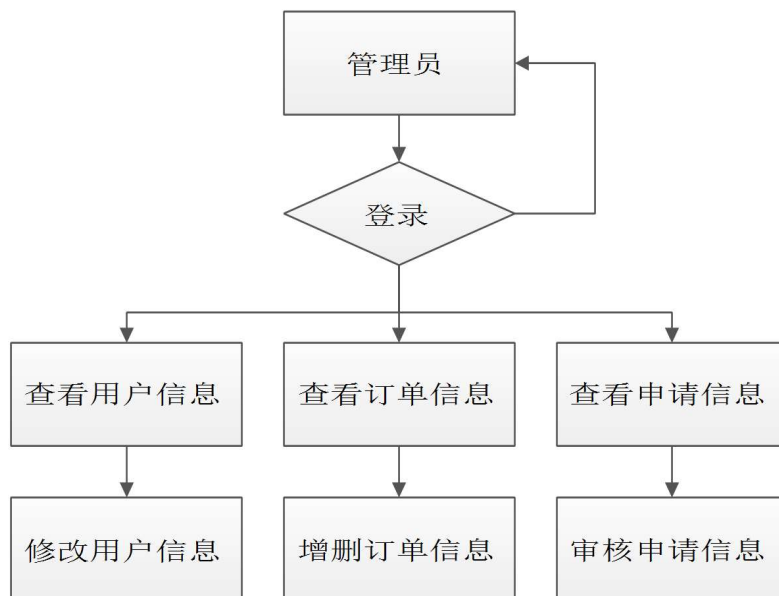


图 3.4 管理员模块流程图

3.5 界面需求分析

在一个购物网站中，界面的好看与否与用户的第一印象是直接挂钩的，一个美观整洁的前台界面可以给人一种心情愉悦的感觉。相反，一个排版错乱的前端页面会让用户第一感觉觉得这个网站是不靠谱的，所以网页的美观与否是十分重要的。此商城采用 JSP 技术来撰写前端页面，使得前后台的交互是十分方便的，

并且也是十分简单的，JSP 技术使得前台排版简单，使得形成的页面布局清晰，美观大方。

1. 输出设计

对于商城系统来说，输出是后台对前台输入信息进行解决与阻截，形成可识别、可用的有效信息，并将这个信息按照一种特定的简洁格排列，来给予后台来分辨并判断，输出设计得职责就是这些。

后台开发的过程与用户浏览并输入的顺序是相反的，并不是前台到后台，二是设计得时候是从后台到前台逆向考虑的，这主要是方便后台开发者进行编写代码以及测试，并且保证前后台信息的准确性。

2. 输入设计

输入设计恰恰是与输出设计相反的，输入设计对于整个系统的影响是巨大的，要知道数据的收集是十分困难的，而且也是易于出错的，一旦出现输入错误，那么逻辑的复杂度也就提高了许多，所以保证输出的准确性是十分重要的。

输入设计要注意以下几点原则：

输入量要准确规定，要有最低限度和最高限度，尽可能简短，因为输入的越少，错误就越少，逻辑也就越简单。

应该尽量使得输入界面简单方便以及输入过程尽量简短，这样也可以使得输入错误率降低。

应在输入数据输入后尽早判断，以便及时作出反应以及处理。

第四章 系统设计

4.1 功能模块设计

依据对前面需求的分析，以及当初的构思，在构思过程中我对商城系统分为两个部分：用户的模块部分和后台的管理模块部分。前台用户的部分要分为用户进行登录商城以及用户没有注册以前要先注册再登录，忘记密码时要修改，浏览商城里面商家的一些商品以及需求或者满意而去购买。此商城的管理员进入后台的第一件事是要登录，如果登进去的话主要有这几个职责：管理前面出现的商品的一些展示信息，还要对前面用户注册完的信息进行管理以及对传到这的订单管理。

1、先看前台用户界面，这块分成前台用户的登录以及未登录时的注册功能，还能查询商品，管理自己的信息以及修改自己的信息。如图 4.1 所示：

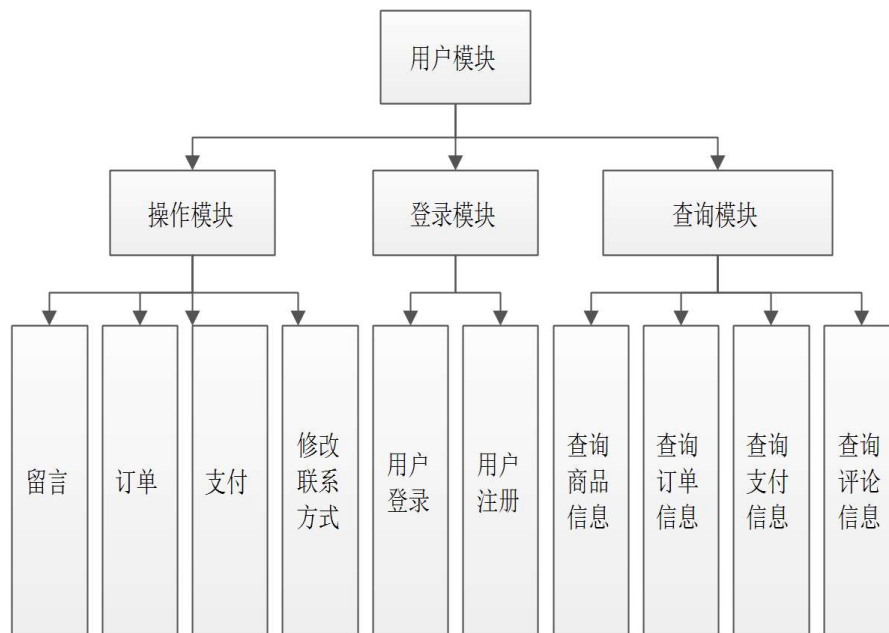


图 4.1 用户功能模块图

2、后台的管理员界面有这么以下几点：对于前台页面上的商品进行管理，包括增加与减少。对前台的用户信息进行修改以及删除，以及对前台顾客购买商品后生成的订单进行管理。此外，还有一个职责是要对前面的页面中间出现的公告进行一些操作管理。如图 4.2 所示：

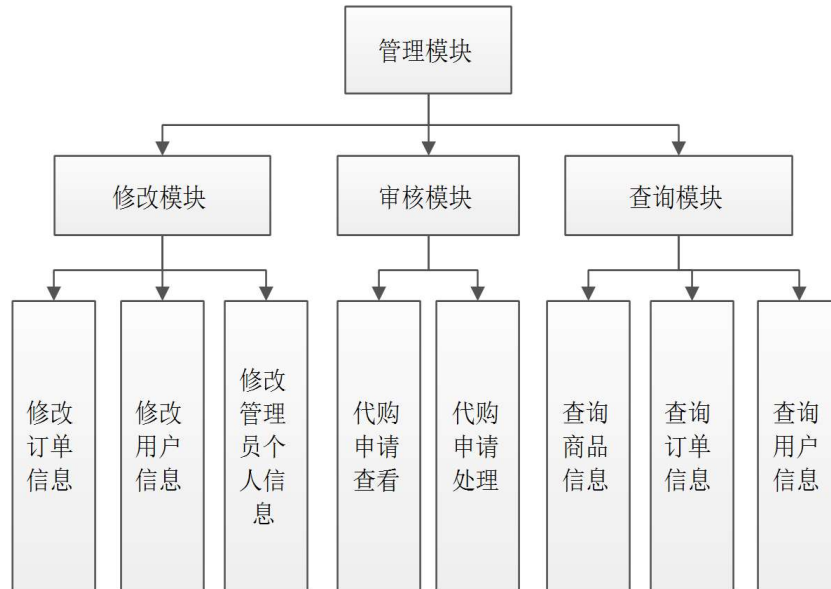


图 4.2 管理功能模块图

4.2 数据库的设计

数据库在一个系统中所占的比重是非常大的，总的来说它负责商城里所有信息的增删改查，数据库设计的是否合理直接影响商城系统性能的优劣性，特别是信息的查找速度，这对于一个商城系统的影响是很大的。

4.2.1 数据库设计原则

在数据库的概念设计中，通常存在四中设计方法：

自顶层往下设计，自底层往顶端设计，还有混合策略设计以及逐步扩张设计。这是数据库常用的四种设计方式。

数据库要先确定模型结构，了解表之间的关联，能不设主键就不要设置，表之间采用中间表进行关联，这对速度的提升是很快的。

4.2.2 数据库实体

数据库实体的设计是要经过需求的不同而不断修改的，力求让的数据库存储信息与用户的需求是一样的，在这种情况下，用 E-R 模型图的设计方法来实现是十分常见的。

我们通过上述分析过程，将需求与概念模型相连接起来，可以建立 E-R 模型图来实现各个需求，此商城系统的 E-R 图如下面的图所示：

1. 管理员的数据实体如下：

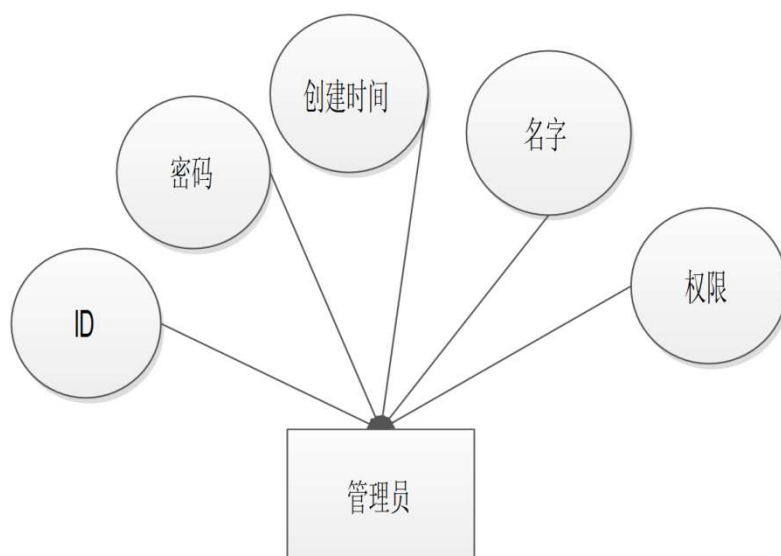


图 4.3 管理员属性实体图

2. 用户的数据实体

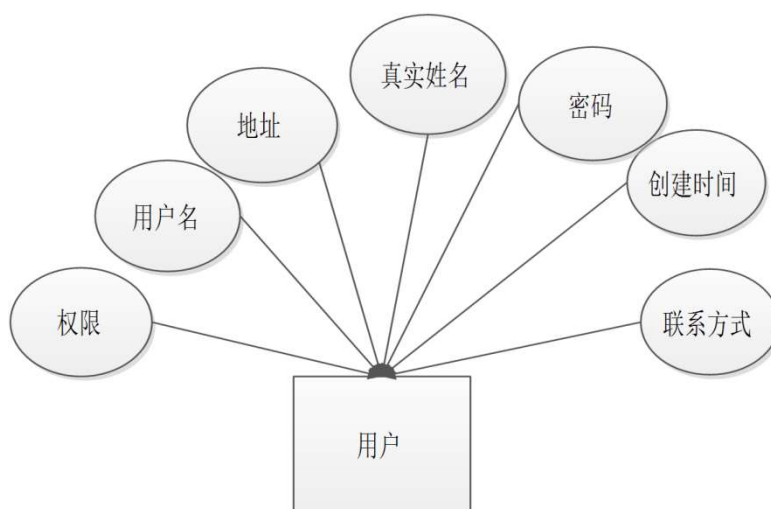


图 4.4 用户属性实体图

3. 商品信息的数据实体

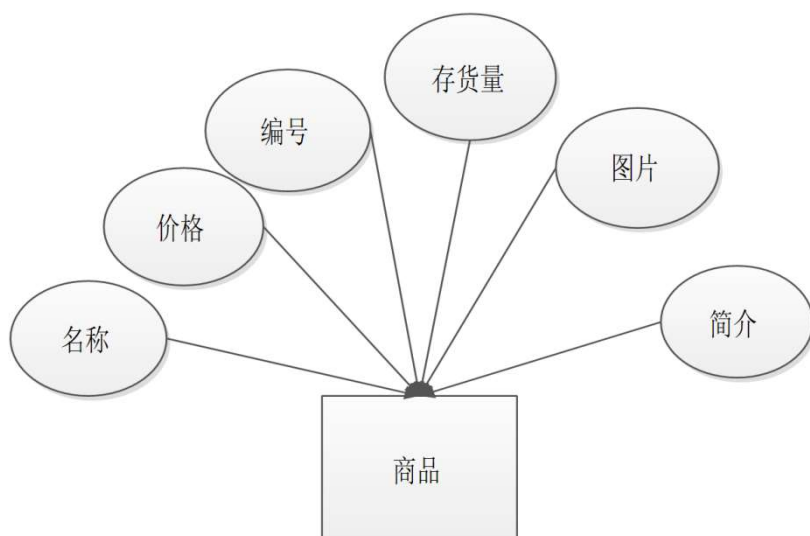


图 4.5 商品信息数据实体图

4. 订单信息的数据实体

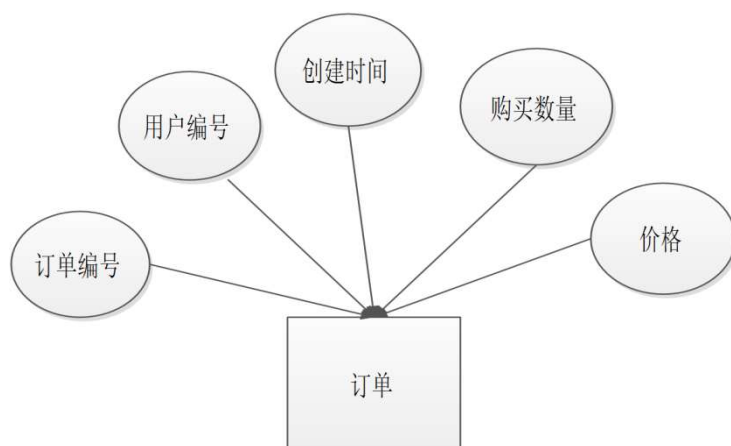


图 4.6 订单信息属性实体图

5. 留言信息的数据实体

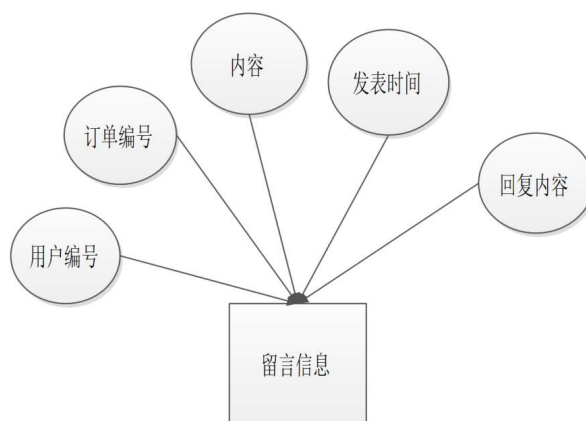


图 4.7 留言信息属性实体图

6. 系统 E-R 图

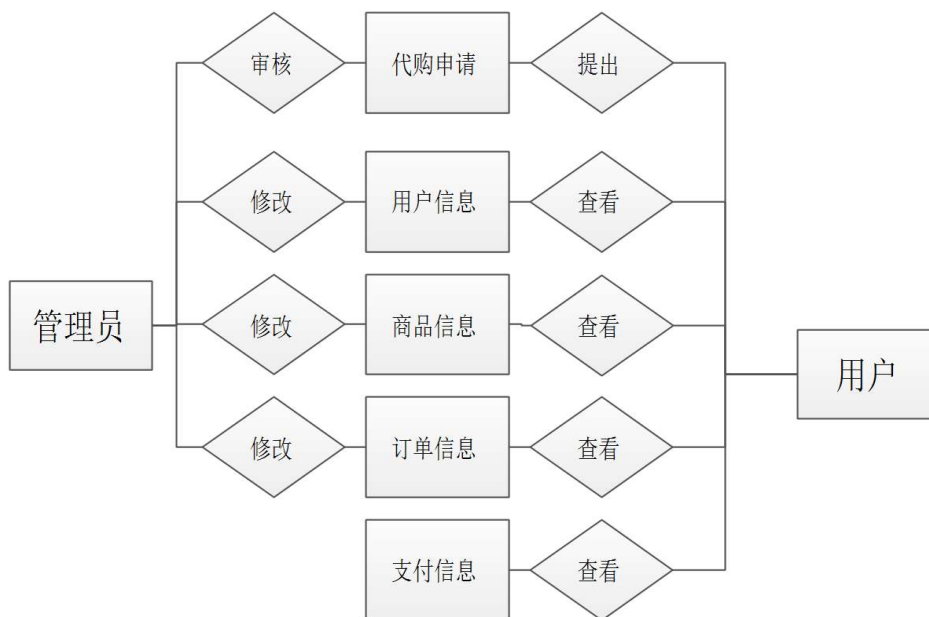


图 4.8 系统信息实体图

4.2.3 创建表数据

表 1 数据用户表

```
CREATE TABLE `user` (
  `userid` int(11) NOT NULL AUTO_INCREMENT,
  `username` varchar(20) DEFAULT NULL,
  `phone` varchar(30) DEFAULT NULL,
  `zhengjianid` varchar(50) DEFAULT NULL COMMENT '证件号',
  `password` varchar(100) DEFAULT NULL,
  PRIMARY KEY (`userid`) USING BTREE
) ENGINE=InnoDB AUTO_INCREMENT=20 DEFAULT CHARSET=utf8 ROW_FORMAT=COMPACT;
```

表 2 管理员信息数据表

```
CREATE TABLE `admin` (
  `adminid` int(11) NOT NULL AUTO_INCREMENT,
  `username` varchar(50) DEFAULT NULL,
  `password` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`adminid`) USING BTREE
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8 ROW_FORMAT=COMPACT;
```

表 3 订单信息数据表

```
CREATE TABLE `orders` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `orderid` varchar(50) DEFAULT NULL,  
  `goodsid` int(11) DEFAULT NULL,  
  `num` int(11) DEFAULT NULL,  
  `ordertime` varchar(100) DEFAULT NULL,  
  `username` varchar(50) DEFAULT NULL,  
  `shopname` varchar(50) DEFAULT NULL,  
  PRIMARY KEY (`id`) USING BTREE  
) ENGINE=InnoDB AUTO_INCREMENT=95 DEFAULT CHARSET=utf8 ROW_FORMAT=COMPACT;
```

表 4 商品信息数据表

```
CREATE TABLE `goods` (  
  `goodsid` int(11) NOT NULL AUTO_INCREMENT,  
  `goodsname` varchar(100) DEFAULT NULL,  
  `goodspic` varchar(50) DEFAULT NULL,  
  `goodsinshop` varchar(50) DEFAULT NULL,  
  `goodsprice` double(10,2) DEFAULT NULL,  
  `goodsintroduce` varchar(500) DEFAULT NULL,  
  `goodstype` varchar(50) DEFAULT NULL,  
  `kucun` int(255) DEFAULT NULL,  
  PRIMARY KEY (`goodsid`) USING BTREE  
) ENGINE=InnoDB AUTO_INCREMENT=29 DEFAULT CHARSET=utf8 ROW_FORMAT=COMPACT;
```

表 5 评价信息数据表

```
CREATE TABLE `pingjia` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `orderid` varchar(50) DEFAULT NULL,  
  `goodsid` int(11) DEFAULT NULL,  
  `pingjiatype` varchar(50) DEFAULT NULL,  
  `pingjiatext` varchar(255) DEFAULT NULL,  
  PRIMARY KEY (`id`) USING BTREE  
) ENGINE=InnoDB AUTO_INCREMENT=27 DEFAULT CHARSET=utf8 ROW_FORMAT=COMPACT;
```

第五章 系统实现

5.1 登录页面

由图 5.1 所示，系统登录界面中，用户填入账户登录信息即可根据不同身份登录不同界面，用户名不能为空且根据用户名、密码以及验证码的正确性判断用户登录信息是否正确。



图 5.1

实现代码如下图：

```
// 登录
@RequestMapping(value = "/login.do")
public ModelAndView loginValidate(HttpServletRequest request,
    HttpServletResponse response) throws IOException {
    HttpSession session = request.getSession(true);
    String username = request.getParameter("username");
    String pwd = request.getParameter("password");
    if (pwd != "" && pwd.equals(userService.validate(username))) {
        // response.setContentType("text/html;charset=utf-8");
        // response.getWriter().write("<script type='text/javascript'> alert('登录成功');</script>");
        // response.getWriter().write("<script> window.location='';window.close();</script>");
        // response.getWriter().flush();
        session.setAttribute("username", username);
        session.setAttribute("password", pwd);
        return new ModelAndView("forward:/index.do");
    } else {
        response.setContentType("text/html;charset=utf-8");
        response.getWriter()
            .write("<script type='text/javascript'> alert('用户名或密码错误，请重新登录');</script>");
        // response.getWriter().write("<script> window.location='views/login/login.jsp' ;window.close();</script>");
        response.getWriter().flush();
        return new ModelAndView("views/login/login.jsp");
    }
}
```

5.2 首页

在首页中，用户登录后可以开始购物和管理自己的信息，并且在左侧可以选择商品分类以及在中间的搜索框可以搜索商品信息。如图 5.2 所示：



图 5.2 首页

实现代码如下图：

```
@Controller
public class goodsController {
    @Autowired
    private GoodsService goodsService;
    @Autowired
    private PingjiaService pingjiaService;

    // 展示商品信息
    @RequestMapping(value = "/goodsPage.do")
    public ModelAndView goodinfo(HttpServletRequest request,
        HttpServletResponse response) throws IOException {
        String goodsid = request.getParameter("goodsid");
        List<Goods> goods = goodsService.goodbyId(Integer.parseInt(goodsid));
        List<Pingjia> goodsPingjia = pingjiaService.findPingjia(Integer.parseInt(goodsid));
        ModelAndView mv = new ModelAndView();
        if (goods != null) {
            mv.addObject("goods", goods);
            mv.addObject("goodsPingjia", goodsPingjia);
            mv.setViewName("views/goods/goodsPage");
        } else {
            mv.setViewName("forward:/index.do");
        }
        return mv;
    }

    //检索商品信息
    @RequestMapping(value = "/searchPage.do")
    public String showGoodsByPage(HttpServletRequest request, Model model) {
        HttpSession session = request.getSession();
        String goodstype = request.getParameter("goodstype");
        String condition = (String) session.getAttribute("condition");
        if (goodstype != null) {
            session.setAttribute("condition", goodstype);
            condition = (String) session.getAttribute("condition");
            List<Goods> allGoods = goodsService.goodbyType(goodstype);
            if (allGoods.size() == 0) {
                model.addAttribute("message", goodstype);
                return "views/goods/searchError";
            }
        }
        this.goodsService.showGoods(request, model, condition);
        return "views/goods/search";
    }
}
```


5.3 注册

新用户进入首页后可以进入注册界面，注册的条件是用户名不少于 6 位且不重复，确认输入的密码与密码必须相同。如图 5.3 所示：



图 5.3

如果用户忘记密码了，还可以通过用户名找回密码，重设密码。如图 5.4 所示：



图 5.4

实现代码如下：

```
// 普通用户注册
@RequestMapping(value = "/register.do")
public String register(HttpServletRequest request,
    HttpServletResponse response) throws IOException {
    User user = new User();
    String username = request.getParameter("username");
    // List<User> n=userService.isUserExist(username);
    // if (n.size()>0) {
    // response.setContentType("text/html;charset=utf-8");
    // response.getWriter().write("<script>alert('用户名已存在');</script>");
    // response.getWriter().write("<script> window.location='../users/addUI.do' ;window.close();</script>");
    // response.getWriter().flush();
    // return "register";
    // }
    String password = request.getParameter("password");
    String phone = request.getParameter("phone");
    // String xuehao = request.getParameter("xuehao");
    String zhengjianid = request.getParameter("zhengjianid");
    // String susheid=request.getParameter("susheid");
    // user.setUserType(0); //普通用户用0表示
    user.setUsername(username);
    user.setPassword(password);
    user.setPhone(phone);
    // user.setXuehao(xuehao);
    user.setZhengjianid(zhengjianid);
    // user.setSusheid(susheid);
    userService.addUser(user);
}
```

```
// 根据用户名找回密码
@RequestMapping(value = "/findPassword.do")
public String findPassword(HttpServletRequest request,
    HttpServletResponse response) throws IOException {
    String username = request.getParameter("username");
    String password = request.getParameter("password");
    List<User> n = userService.checkUsernameExist(username);
    if (n.size() == 0) {
        response.setContentType("text/html;charset=utf-8");
        response.getWriter().write(
            "<script>alert('该用户名不存在，请核对后输入!');</script>");
        // response.getWriter().write("<script> window.location='views/login/findPassword.jsp' ;</script>");
        response.getWriter().flush();
        return "views/login/findPassword";
    } else {
        userService.updatePwd(username, password);
        response.setContentType("text/html;charset=utf-8");
        response.getWriter().write(
            "<script>alert('密码已找回，前往登录界面!!');</script>");
        // response.getWriter().write("<script> window.location='views/login/login.jsp' ;</script>");
        response.getWriter().flush();
        return "views/login/login";
    }
}
```

5.4 查看商品

在主页面显示的属于用户操作的这一部分页面中，用户可以看一些已经有了的商品的属性，还可以在框中搜索需要的东西，以及对这个商品喜欢的话可以进行收藏。如图 5.5 所示：

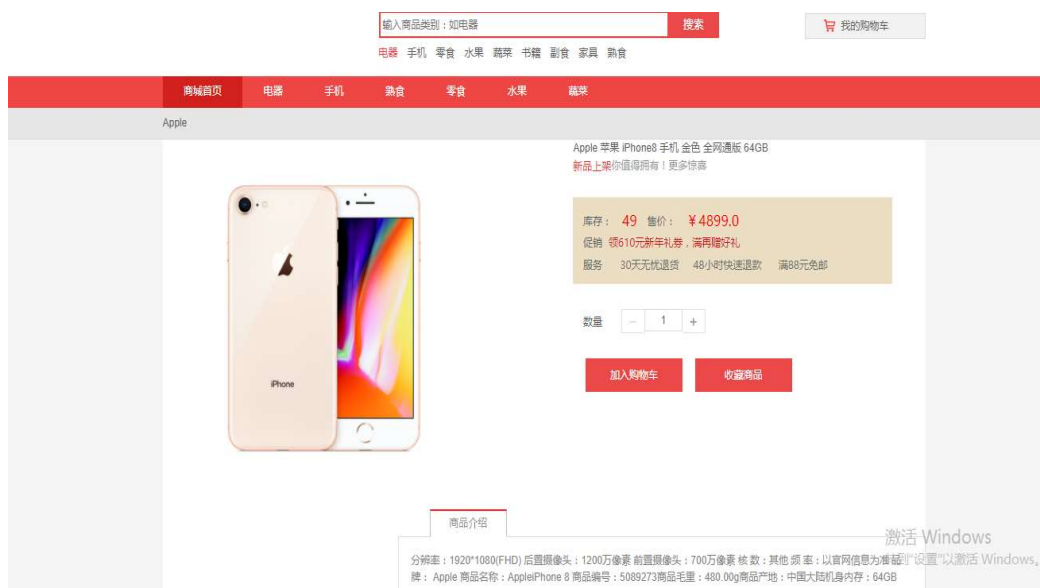


图 5.5

实现代码如下：

```
//检索商品信息
@RequestMapping(value = "/searchPage.do")
public String showGoodsByPage(HttpServletRequest request, Model model) {
    HttpSession session = request.getSession();
    String goodstype = request.getParameter("goodstype");
    String condition = (String) session.getAttribute("condition");
    if (goodstype != null) {
        session.setAttribute("condition", goodstype);
        condition = (String) session.getAttribute("condition");
        List<Goods> allGoods = goodsService.goodsByType(goodstype);
        if (allGoods.size() == 0) {
            model.addAttribute("message", goodstype);
            return "views/goods/searchError";
        }
    }
    this.goodsService.showGoods(request, model, condition);
    return "views/goods/search";
}
```

5.5 查看订单

在用户界面中，可以在订单中查看自己购买的商品，，查看订单信息，完成支付操作、确认收货、追加评价以及取消订单。如图 5.6 所示：

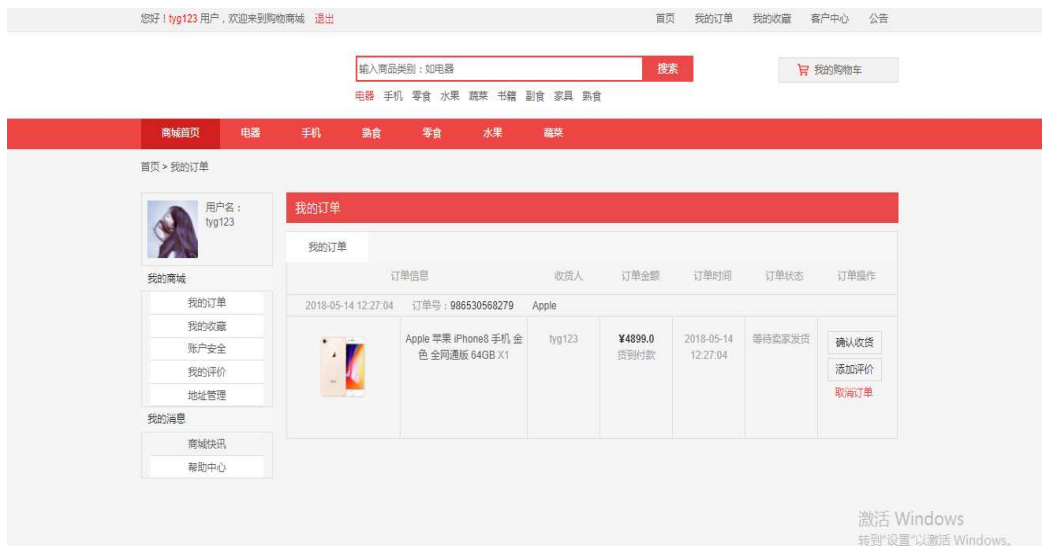


图 5.6

实现代码如下：

```

// 生成订单
// 订单列表
@RequestMapping(value = "/orderList.do")
public ModelAndView orderList(HttpServletRequest request,
    HttpServletResponse response) throws IOException {
    String username = request.getParameter("username");
    List<Order> order = orderService.selectOrder(username);
    ModelAndView mv = new ModelAndView();
    mv.addObject("order", order);
    mv.setViewName("views/order/orderList");
    return mv;
}

// 删除订单
@RequestMapping(value = "/deleteMyOrder.do")
@ResponseBody
public String deleteMyOrder(@RequestParam int id) throws IOException {
    orderService.deleteOrder(id);
    return "true";
}

// 生成订单订单
@RequestMapping(value = "/addOrder.do")
@ResponseBody
public String addOrder(@RequestParam int carid, HttpSession session)
    throws IOException {
    int goodsid = carService.findgoodsid(carid);
    int num = carService.findnum(carid);
    int kucun = goodsService.findKucun(goodsid);
    String shopname = goodsService.findshopname(goodsid);
    String username = (String) session.getAttribute("username");
    Date day = new Date();
    SimpleDateFormat df = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
    String ordertime = df.format(day);
    orderService.addOrder(" " + carid, goodsid, num, ordertime, username,
        shopname);
    goodsService.updateKucun(goodsid, kucun-num);
    carService.deleteCar(carid);
    return "true";
}

```


5.6 评价

用户在完成订单后可以对商品进行评价，如图 5.7 所示：

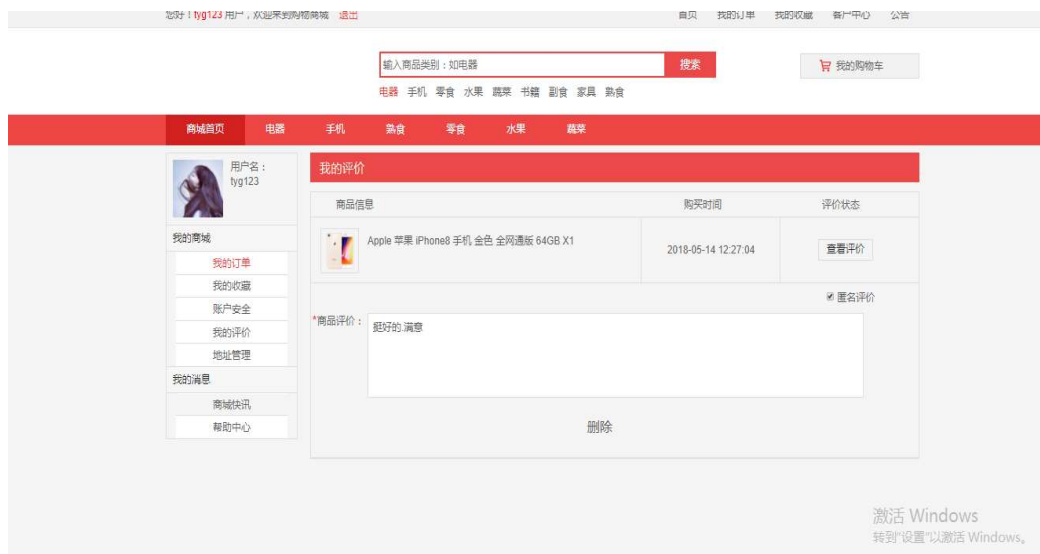


图 5.7

实现代码如下：

```
//生成评论

@RequestMapping(value = "/showPingjia.do")
public ModelAndView showPingjia(HttpServletRequest request, HttpServletResponse response) throws IOException {
    String orderid=request.getParameter("orderid");
    String goodsid=request.getParameter("goodsid");
    List<Pingjia> pingjia=pingjiaService.selectPingjia(orderid, Integer.parseInt(goodsid));
    ModelAndView mv = new ModelAndView();
    mv.addObject("pingjia", pingjia);
    mv.setViewName("views/pingjia/showPingjia");
    return mv;
}

@RequestMapping(value = "/myPingjia.do")
public ModelAndView myPingjia(HttpServletRequest request, HttpServletResponse response) throws IOException {
    String username=request.getParameter("username");
    List<NewOrder> pingjia=pingjiaService.myPingjia(username);
    ModelAndView mv = new ModelAndView();
    mv.addObject("pingjia", pingjia);
    mv.setViewName("views/pingjia/myPingjia");
    return mv;
}

//添加评价内容
@RequestMapping(value = "/insertPingjia.do")
public ModelAndView insertPingjia(HttpServletRequest request, HttpServletResponse response) throws IOException {
    String orderid=request.getParameter("orderid");
    String goodsid=request.getParameter("goodsid");
    pingjiaService.insertPingjia(orderid, Integer.parseInt(goodsid));
    List<Pingjia> pingjia=pingjiaService.selectPingjia(orderid, Integer.parseInt(goodsid));
    ModelAndView mv = new ModelAndView();
    mv.addObject("pingjia", pingjia);
    mv.setViewName("views/pingjia/showPingjia");
    return mv;
}

//将评价内容致为空
@RequestMapping(value = "/delete.do")
@ResponseBody
public void deletePingjia(@RequestParam String id) throws IOException {
    pingjiaService.update(Integer.parseInt(id));
}
}
```

5.7 管理个人信息

用户可以在个人信息模块来查看自己的详细信息并且修改个人信息，如图 5.8 所示：

商城首页 电器 手机 熟食 零食 水果 蔬菜

首页 > 账户安全

用户名：tyg123

我的商城

我的订单

我的收藏

账户安全

我的评价

地址管理

我的消息

商城快讯

帮助中心

账户安全

用户名 | tyg123

登录密码 | *****

联系方式 | 15667027560

邮箱 | 15667027560@163.com

确定修改

安全提示：
1. 注意防范进入钓鱼网站，不要轻信各种即时通讯工具发送的商品或支付链接，谨防网络诈骗。
2. 建议您安装杀毒软件，并定期更新操作系统等软件补丁，确保账户及交易安全。

激活 Windows
转到“设置”以激活 Windows。

图 5.8

在此页面还可以增加以及修改已存在收货地址，如图 5.9 所示：

您好！tyg123 用户，欢迎来到购物商城 退出

首页 我的订单 我的收藏 客户中心 公告

输入商品类别：如电器 搜索

购物车

电器 手机 零食 水果 蔬菜 书籍 副食 家具 熟食

商城首页 电器 手机 熟食 零食 水果 蔬菜

首页 > 会员中心 > 添加地址

用户名：tyg123

我的商城

我的订单

我的收藏

账户安全

我的评价

地址管理

我的消息

商城快讯

帮助中心

添加地址

地区：陕西省西安市长安区

详细地址：西安邮电大学长安校区

邮政编码：710121

姓名：谭发钢

手机号码：15667027560

地址类别：家

提交

激活 Windows
转到“设置”以激活 Windows。

图 5.9

实现代码如下：

```
// 添加新地址
@RequestMapping(value = "/addAddress.do")
public ModelAndView operateAddress(HttpServletRequest request,
    HttpServletResponse response) throws IOException {
    String username = request.getParameter("username");
    String area = request.getParameter("area");
    String address = request.getParameter("address");
    String code = request.getParameter("code");
    String realname = request.getParameter("name");
    String phone = request.getParameter("phone");
    String addresstype = request.getParameter("addresstype");
    Address a = new Address();
    a.setAddress(address);
    a.setAddresstype(addresstype);
    a.setArea(area);
    a.setCode(code);
    a.setRealname(realname);
    a.setPhone(phone);
    a.setUsername(username);
    addressService.insert(a);
    return new ModelAndView("forward:/addressList.do?username=" + username);
}
/*
 * 添加新地址
 */
}
```

```
// 更新新地址
@RequestMapping(value = "/upAddress.do")
public ModelAndView upAddress(HttpServletRequest request,
    HttpServletResponse response) throws IOException {
    String id = request.getParameter("id");
    String username = request.getParameter("username");
    String area = request.getParameter("area");
    String address = request.getParameter("address");
    String code = request.getParameter("code");
    String realname = request.getParameter("name");
    String phone = request.getParameter("phone");
    String addresstype = request.getParameter("addresstype");
    addressService.update(Integer.parseInt(id), area, address, code,
        realname, phone, addresstype);
    return new ModelAndView("forward:/addressList.do?username=" + username);
}
}
```

5.8 管理员界面

此商城的管理员想要进入后台系统，第一件事就是需要登录，如果信息对的话可以进去，在这有一个要点，这个界面里面没有管理员注册这个，因为要保证安全，管理员账号密码是直接存储在数据库里面的，修改可直接在数据中进行修改，如图 5.10 所示：



图 5.10

实现代码如下：

```
// 管理员登录
@RequestMapping(value = "/adminLogin.do")
@ResponseBody
public int adminLogin(@RequestParam String username, String password,
    HttpServletRequest request) throws IOException {
    HttpSession session = request.getSession(true);
    if (password != ""
        && password.equals(adminService.adminLogin(username))) {
        session.setAttribute("adminname", username);
        return 1;
    } else
        return 0;
}
```

5.9 管理员用户管理界面

管理员可以对用户信息进行修改以及删除，如图 5.11 所示：

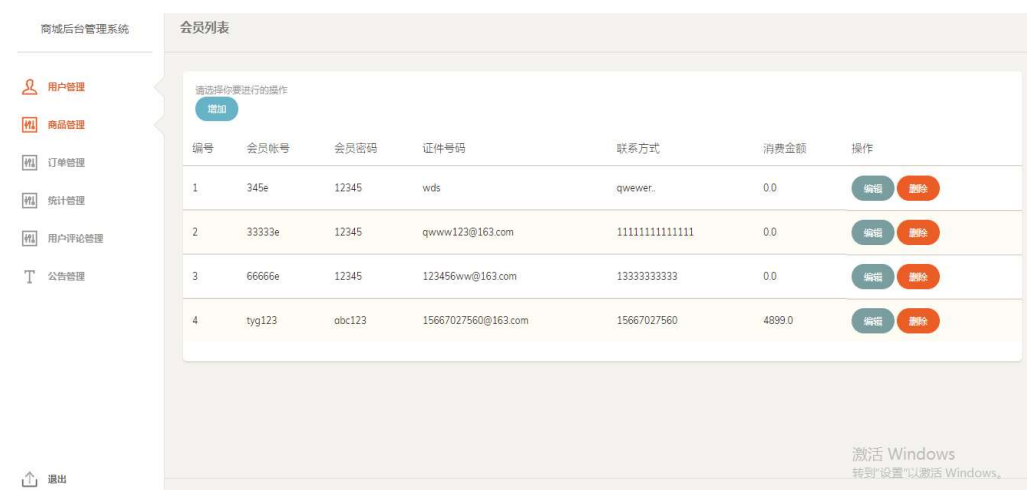


图 5.11

实现代码如下：

```
// 用户列表
@RequestMapping(value = "/selectUser.do")
public ModelAndView selectUser(HttpServletRequest request,
    HttpServletResponse response) throws IOException {
    List<User> user = adminService.selectUser();
    double money=0;
    for(int i=0;i<user.size();i++){
        List<Order> order=orderService.selectOrder(user.get(i).getUsername());
        for(int j=0;j<order.size();j++){
            money+=order.get(j).getNum()*order.get(j).getGoodsprice();
        }
        user.get(i).setMoney(money);
        money=0;
    }
    ModelAndView mv = new ModelAndView();
    mv.addObject("findUserList", user);
    mv.setViewName("views/admin/userTable");
    return mv;
}
```

5.10 管理员订单管理

管理员可以查看订单，并确定是否发货以及删除订单，如图 5.12 所示：

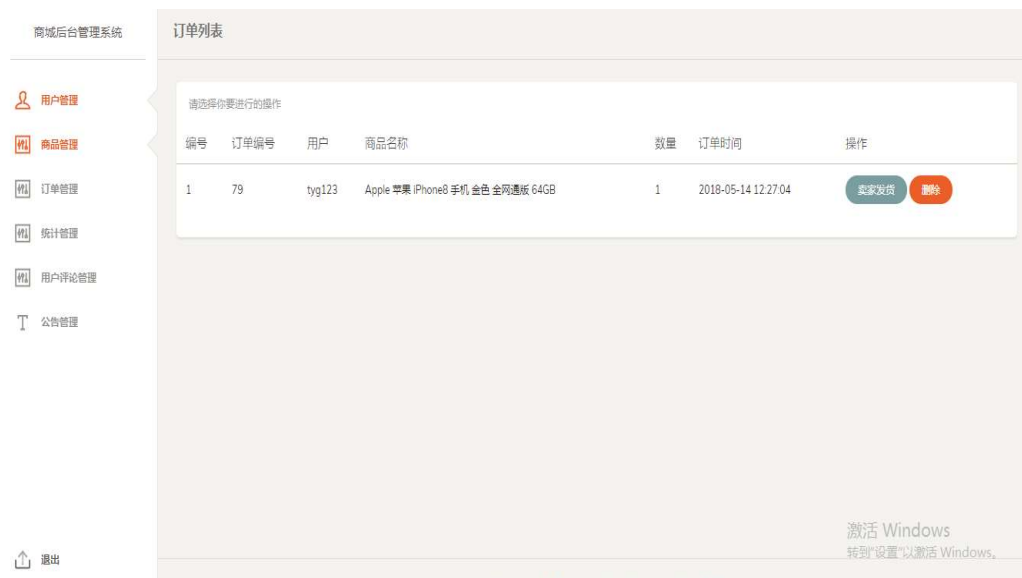


图 5.12

实现代码如下：

```
// 管理订单
@RequestMapping(value = "/adminOrder.do")
public ModelAndView adminOrder(HttpServletRequest request,
    HttpServletResponse response) throws IOException {
    List<Order> order = adminService.selectOrder();
    ModelAndView mv = new ModelAndView();
    mv.addObject("order", order);
    mv.setViewName("views/admin/adminOrder");
    return mv;
}

// 删除订单
@RequestMapping(value = "/deleteOrder.do")
public ModelAndView deleteOrder(HttpServletRequest request,
    HttpServletResponse response) throws IOException {
    String id = request.getParameter("id");
    adminService.deleteOrder(Integer.parseInt(id));
    ModelAndView mv = new ModelAndView();
    return new ModelAndView("forward:/adminOrder.do");
}
```

5.11 管理员公告管理

管理员增加以及修改公告，如图 5.13 所示：

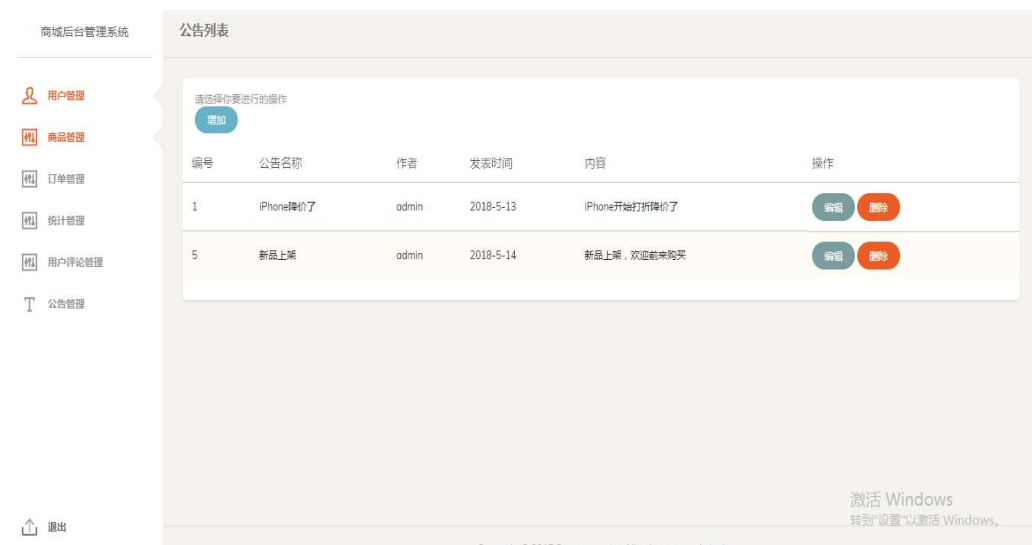


图 5.13

实现代码如下：

```
// 使要修改的信息在页面显示
@RequestMapping("/upNews")
public String upnews(Integer id, String newsname, String author,
    String date, String context, HttpSession session)
    throws IOException {
    newsname = new String(newsname.getBytes("ISO-8859-1"), "UTF-8");
    author = new String(author.getBytes("ISO-8859-1"), "UTF-8");
    context = new String(context.getBytes("ISO-8859-1"), "UTF-8");
    List list = new ArrayList();
    list.add(id);
    list.add(newsname);
    list.add(author);
    list.add(date);
    list.add(context);
    session.setAttribute("upList", list);
    return "views/admin/upNews";
}

// 更新新闻信息
@ResponseBody
@RequestMapping(value = "/updateNewsInfo")
public String upAdminInfo(@RequestParam Integer id, String newsname,
    String author, String date, String context) {
    newsService.updateNews(id, newsname, author, date, context);
    return "true";
}
```

结束语

这次我的毕业论文主要是对我写出来的商城系统进行一些说明。主要对商城中的主要的一些功能进行了介绍，在此系统的设计过程中我是严格按照分模块来一步步编写以及测试的，所以我的论文也是按照模块化来介绍的。此商城系统我用的语言是 java 语言，而在前端页面的编写中用的是 JSP+HTML 技术，开发用的数据库是 MYSQL，用此来创建编写中用到的变量以及存储商城系统中一些信息，还对 tomcat, myeclipse 等开发工具更熟练的运用。本论文的编写完全按照严格的企业级别的开发流程来的，先是要对项目的是否可行性进行一些分析，然后再进行需求方面的分析，然后构思此商城系统的构架以及所需要用到的东西，在最后是划分模块并且再进行开发。

在此系统的开发过程中，我也是遇到了很多的问题，在此过程中有过烦躁，有过放弃等想法，但是最终还是完成了此设计，这是让我十分开心的，收获了许多。遇事冷静分析，逐步解决错误，善于对问题进行一些思考，并在解决后及时总结经验。我在此过程中也是收到了许多的经验以及多位同学的帮助，这使得我的开发可以顺利进行，让我的知识了解的更加的牢固。

致 谢

感谢老师在项目中对我的指导，使我受益匪浅，在老师的大力帮助下终于克服了重重地难关，顺利的完成了这次设计。我所学到的不仅仅是知识，也有对生活的态度，一种积极向上、不断进取、一丝不苟的生活态度。感谢我系各位老师对该项目的大力支持。各位同学对我的帮助，对我在该项目中遇到的很多问题给予了很大的帮助。

感谢父母在我大学以及项目期间对我的关怀，还有寝室的同学在四年中给我的帮助关怀，在我的论文中给我最近的帮助。

在本次毕业论文的开发过程中，我得到了指导老师的技术指导，在这里我要特别感谢我的指导老师对我的大力帮助。我的指导老师非常悉心指导我和大力的支持我，在系统开发的架构总体结构还有功能上都给予了我很大的帮助，与此同时，特给我们提供这么好这么安静的环境，而且对我在编程开发设计的过程中很多细节上的工作进行了耐心给力的教导，对于现在的我能顺利完成我的毕业论文是起到了决定性的作用。通过开发本软件，我的 Java 开发能力得到了进一步的提高。

参考文献

- [1]谷庆华,李成贵.基于 Java 语言实现数据库的访问[J].计算机技术与发展,2015(02):13-16.
- [2]裴生雷.Ajax+JSP 开发模式的研究及应用[J].计算机技术与发展,2013,23(01):242-245+248.
- [3]李贤华.基于 JSP 技术的大型网上购物系统的设计与实现[J].计算机与现代化,2015(08):82-84+89.
- [4]李洋.SSM 框架在 Web 应用开发中的设计与实现[J].计算机技术与发展,2016,26(12):190-194.
- [5]陈文超.浅析 Tomcat Web 服务器基于 Apache 的集群与负载均衡[J].科技资讯,2012(06):28-30.
- [6]刘冲,张海玥,张卫东,周秀中,林少凡.配置 Tomcat 使 Apache 服务器支持 Java 动态网页编程[J].计算机应用,2016(S1):109-110.
- [7]欧锋,邹敏,李晓桢.Java 技术框架概述[J].计算机系统应用,2012,21(08):236-239.
- [8]陆洲.Java EE 核心模式研究[J].计算机与数字工程,2014,42(01):81-84.
- [9]贾文潇,邓俊杰.基于 Java 的 Web 开发技术浅析[J].电子测试,2016(08):65+86.
- [10]Daniel Tejera,Alejandro Alonso,Miguel A. de Miguel. Diseño de un Software de Intermediación de Comunicación para Sistemas Distribuidos de Tiempo Real Críticos en Java[J]. Revista Iberoamericana de Automática e Informática Industrial RIAI,2013,10(2).
- [11]Andreas Lochbihler. Analysing Java's safety guarantees under concurrency[J]. it – Information Technology,2014,56(2).
- [12]Matthew Moodie. Pro Apache Tomcat 6[M].Apress:2015-06-15.