

毕业设计（论文）材料之二（1）

安徽工程大学本科

毕业设计（论文）

专 业： 软件工程

题 目： 基于 SSM 的网上商场

作 者 姓 名： 郑 斌

导师及职称： 邹 姗（讲师） / 王斌（讲师）

导师所在单位： 计算机与信息学院

年 月 日

安徽工程大学

本科毕业设计（论文）任务书

2020 届 计算机与信息 学院

软件工程 专业

学生姓名： 郑斌

I 毕业设计（论文）题目

中文： 基于 SSM 的网上商城

英文： **Online Shopping Mall Based on SSM**

II 原始资料

- [1]陈海汝,何青,潘轩平,张大宇,胡铁平.基于云服务器的 SSM 框架后台搭建与实现[J].信息系统工程,2019(11):114-115+117.
- [2]蔡呈杰,王贵鑫.基于 SSM 框架的学子商城系统的设计与实现[J].科学技术创新,2019(32):69-71.
- [3]侯金奎,鹿旭东,陈春雷,王磊.基于模型驱动的 Web 应用服务系统开发理论框架[J].小型微型计算机系统,2018,39(10):2345-2352.
- [4]鞠宏军,余春燕.基于 SSM 的高并发慕课网的设计与实现[J].电脑知识与技术,2020,16(02):54-56+90.
- [5]戚琦,廖建新,王纯,武家春.基于敏捷方法的轻量级 J2EE 架构的应用[J].计算机系统应用,2007(02):53-56.

III 毕业设计（论文）任务内容

1、课题研究的意义

随着无店铺零售的悄然兴起，是商业经济发展到一定阶段的产物，也是顺应新的消费需求和技术进步的必然结果。无店铺零售则大多不存在地段方面的制约。二是经营成本。借助日益发展的科学技术，无店铺零售通常能够省却从生产到销售过程中的许多环节，从而达到有效降低经营成本与交易成本，带来消费者、商家双赢的局面。

鉴于互联网的优势以及对国内外相关电子商务网站现状的研究分析，我决定以基于 Web 的商城电子商务网站开发作为我的毕业设计主题。立足于设计一个在网络平台上运行的商城，以小小的力量推动中国互联网的发展。

2、本课题研究的主要内容：

- (1) 掌握实现网上商城系统的基本技术。
- (2) 收集资料，阅读文献。
- (3) 可行性分析，需求分析，系统功能实现，测试相关功能。
- (4) 本课题是一个基于 SSM 的网上购物平台，前台用户功能模块主要是实现了用户对公告、商品的浏览以及查询，登录后，可以实现物品购物车的加入和购买等功能。后台管理模块主要实现了管理员对商品、订单、投诉的管理。
- (5) 整理相关文档，并撰写毕业设计报告。

3、提交的成果：

- (1) 毕业设计（论文）正文；
- (2) 基于 SSM 的网上商场；
- (3) 至少一篇引用的外文文献及其译文；
- (4) 附不少于 10 篇主要参考文献的题录及摘要。

指导教师（签字）_____

教研室主任（签字）_____

批 准 日 期 _____

接 受 任 务 书 日 期 _____

完 成 日 期 _____

接受任务书学生（签字）_____

基于 SSM 的网上商场

摘 要

随着计算机技术和网络的发展，网络购物平台因为其成本小，没有时间，空间上的限制，慢慢成为新的零售模式和理念。网上购物已经普遍被大众所接受，尤其受到年轻人的喜爱。传统零售企业也在不断拓展线上业务，建立自己的网上购物平台去迎合市场的需要。建立一个网上商城购物平台不仅可以满足顾客的购买需求，也能帮助零售企业更好的发展线上市场，降低成本，使企业和顾客达到双赢。

本系统分为前台和后台部分，前台针对用户，主要包括购物车管理、订单管理、修改账户密码、添加收货地址信息、用户注册账号、登录系统等。后台是针对管理员，主要包括商品的修改删除添加理、处理订单的状态、用户个人的信息管理等。在最后的系统开发实现环节，使用面向对象来进行程序编写，使程序可重用性大大提高，让系统设计的更合理也更具有通用性，从而让系统实用性大大提高。建立后的网站系统是一个动态、交互式、具有用户账号注册登录、商品浏览、商品购买、系统管理、等功能的电子商务网站。

本系统以 Java 语言为基础，Ajax 和 Bootstrap 为开发页面的框架，通过 Servlet 来实现功能，JDBC 连接 MYSQL 数据库。本文着重介绍了网上商场系统的需求分析、实现的功能、业务流程、数据库设计、系统功能模块设计、设计等一系列开发流程。

关键词：JSP；Mysql；Java；网上购物系统

Online Shopping Mall Based on SSM

Abstract

With the development of computer technology and network, online shopping platform has gradually become a new retail mode and concept because of its small cost, no time and space constraints. Online shopping has been generally accepted by the public, especially by young people. Traditional retail enterprises are also expanding their online business and building their own online shopping platform to meet the needs of the market. The establishment of an online shopping mall platform can not only meet the purchase needs of customers, but also help retail enterprises to better develop the online market.

This system is divided into foreground and background parts. The foreground is aimed at users, mainly including shopping cart management, order management, account password modification, add receiving address information, user registration account, login system, etc. The background is for the administrator, mainly including the modification, deletion and addition of goods, the status of processing orders, the user's personal information management, etc. In the last part of the system development, the object-oriented programming is used to improve the reusability of the program, make the system design more reasonable and more general, so as to improve the practicability of the system. The established website system is a dynamic, interactive, e-commerce website with the functions of user account registration and login, commodity browsing, commodity purchase, system management, etc.

The system is based on Java language, Ajax and bootstrap are the framework of developing page, and functions are realized through servlet. JDBC connects MySQL database. This paper focuses on the requirements analysis, functions, business processes, database design, system function module design, design and a series of development processes.

Keywords: JSP; MySQL; Java; online shopping system

目 录

引言.....	1
第 1 章 绪论.....	2
1.1 研究背景.....	2
1.2 研究意义.....	2
1.3 发展状况.....	2
1.4 研究的主要内容.....	3
第 2 章 系统开发环境及项目开发工具.....	4
1.1 开发技术简介.....	4
1.1.1 JSP 简介.....	4
1.1.2 Servlet 相关简介.....	4
1.1.3 MySql 相关简介.....	4
1.1.4 IDEA 简介.....	5
1.2 系统运行环境及配置.....	5
第 3 章 需求分析.....	6
3.1 可行性分析.....	6
3.1.1 经济可行性分析.....	6
3.1.2 技术可行性分析.....	6
3.1.3 社会可行性分析.....	6
3.2 系统需求分析.....	6
3.3 结构化分析.....	7
3.3.1 数据流图.....	7
3.3.2 数据字典.....	10
第 4 章 系统设计.....	15
4.1 系统模块设计.....	15
4.1.1 前台面向用户的功能模块设计.....	15
4.1.2 后台模块的功能模块设计.....	16
4.2 系统流程设计.....	16
4.3 数据库设计.....	19
4.3.1 数据库概要设计.....	19
4.3.2 数据库表.....	22
第 5 章 系统实现.....	27
5.1 前台设计.....	27
5.1.1 用户注册.....	27
5.1.2 用户登录.....	27
5.1.3 商品推荐及信息浏览.....	28
5.1.4 添加商品到购物车.....	31
5.1.5 订单相关页面.....	33
5.2 后台实现部分.....	34
5.2.1 商品类型管理.....	34
5.2.2 商品管理.....	35

5.2.3 订单管理.....	36
5.2.4 公告管理.....	37
5.2.5 用户管理.....	38
第 6 章 系统测试.....	39
6.1 软件测试.....	39
6.1.1 注册登录测试.....	39
6.1.2 购物车功能测试.....	41
6.1.3 订单功能测试.....	42
6.1.4 商品评价测试.....	42
6.1.5 收货地址测试.....	43
6.1.6 商品分类管理测试.....	44
6.1.7 商品的添加, 修改测试.....	44
6.1.8 订单管理测试.....	45
6.1.9 用户管理测试.....	45
6.1.10 商品评论回复管理测试.....	46
6.2 测试计划执行情况.....	46
6.3 测试效果.....	47
总结与展望.....	48
致谢.....	49
参考文献.....	50
附录.....	51
附录 A 外文文献及其译文.....	51
附录 B 主要参考文献的题录及摘要或参考文献原文.....	60
附录 C 部分源代码.....	63

插图清单

图 2-1 原理图.....	4
图 3-1 顶层图数据流图	7
图 3-2 一层数据流图.....	7
图 3-3 用户注册及信息管理二层数据流图.....	8
图 3-4 订购商品二层数据流图.....	8
图 3-5 订单管理二层数据流图.....	9
图 3-6 商品管理二层数据流图.....	9
图 4-1 系统功能图.....	15
图 4-2 前台购物流程图.....	16
图 4-3 后台管理流程图.....	17
图 4-4 注册功能流程图.....	17
图 4-5 登录流程图.....	17
图 4-6 商品搜索购买流程图.....	18
图 4-7 商品管理流程图.....	18
图 4-8 商品的实体属性图.....	19
图 4-9 顾客的实体属性图.....	19
图 4-10 管理员的实体属性图.....	20
图 4-11 订单的实体属性图.....	20
图 4-12 收货地址的实体属性图	20
图 4-13 购物车的实体属性图.....	20
图 4-14 商品类别的实体属性图	21
图 4-15 系统公告的实体属性图	21
图 4-16 商品评价的实体关系图	21
图 4-17 ER 图.....	22
图 5-1 用户注册.....	27
图 5-2 用户登录.....	28
图 5-3 首页.....	29
图 5-4 商品信息图.....	30
图 5-5 注册.....	31
图 5-6 购物车.....	32
图 5-7 订单.....	33
图 5-8 编辑分类.....	34
图 5-9 添加商品.....	35
图 5-10 订单管理.....	36
图 5-11 公告管理.....	37
图 5-12 用户管理.....	38
图 6-1 注册测试.....	40
图 6-2 登录测试.....	40
图 6-3 购物车测试.....	41
图 6-4 订单测试.....	42

图 6-5 评论测试.....	42
图 6-6 商品信息测试.....	43
图 6-7 收货信息测试一.....	43
图 6-8 收货信息测试二.....	44
图 6-9 分类测试.....	44
图 6-10 添加商品测试.....	45
图 6-11 订单管理测试.....	45
图 6-12 用户管理测试.....	45
图 6-13 评论回复测试.....	46

表格清单

表 4-1 表名:user.....	23
表 4-2 表名:product.....	23
表 4-3 表名:orders.....	24
表 4-4 表名:orderitem.....	24
表 4-5 表名:car.....	25
表 4-6 表名:admin.....	25
表 4-7 表名:notice.....	25
表 4-8 表名:category.....	26
表 4-9 表名:goodscomments.....	26
表 6-1 测试计划表.....	46
表 6-2 测试结果表.....	47

引 言

目前互联网在以一种令人难以想象的冲击力在改变着人们的生活方式，不仅与人们的生活息息相关，还改变了生产方式。互联网的诞生和迅速发展不仅改变了传统信息传递方式，还打破了传统交流方式中时间和空间的各种限制，并且不断给人们带来了新的机会和挑战。

自 21 世纪以来，中国互联网高速发展，涉及到互联网基础设施建设，互联网政务，互联网学术以及互联网学术等方面。根据有关部门的相关报告，截至 2020 年，中国网民人数达到 9 亿，互联网普及率到达 64%。巨大的网民规模造就了中国这个世界上最大的消费市场。同时基于计算机科学信息技术的高速发展，数字经济也有了相当夯实的技术基础和用户基础。

本次毕业设计的课题是利用 SSM+bootstrap+jsp 技术搭建一个网上商场的购物平台。使用 Java 语言作为基础，SSM 为开发页面的框架，以 Servlet 来实现功能。为方便对数据库中的数据表进行有效管理，我使用 MySQL 数据库管理系统。Servlet 技术来实现相应的功能，编译软件使用 IntelliJ IDEA，服务器使用 Tomcat。

网上购物这个逐渐流行于 21 世纪的购物方式已经被越来越多的人所接受我国政府也正全面，积极稳妥的推进电子商务的发展。网上商店在增加移动支付功能，完善各项功能后以更大的势头发展。电子商务平台的发展代表着未来商品交易的发展方向 and 趋势，其应用将给社会和经济带来巨大的效益。大力发展电子商务在国民经济各个领域的应用，力争在越来越激烈的竞争中占据有利位置。因此对于网上商城的研究和实现，以及搭建一个实用的网上商店是很有必要的。

第 1 章 绪论

1.1 研究背景

21 世纪以来全球各国进入一个崭新的经济时代,信息通信技术和网络的迅猛发展创造了一个无边界的数字世界,产品和金融服务几乎可以在人们感受不到的时间跨度中在任何地方进行交易,以互联网和电子支付为特征的新经济席卷了全球每一个行业,影响了每一个人^[1]。

自从中国加入国际贸易组织 WTO 之后,传统大型企业的发展必然趋势是互联网化、全球化^[2]。通过建设电子商务基础平台,使得企业不必专注于复杂多变的信息技术,而用最小的资金投入、最简单的接入方式去联系最广泛的客户和供应商,进行最先进、最有效的电子商务应用服务,相关企业实行电子商务则成为企业发展与国际接轨的必要条件,所以电子购物平台的实现是非常迫切的^[3]。

当今的零售企业,如果依旧使用传统的方式例如:发传单,实体店内打广告宣传自己的产品、本公司的文化则显得十分落后,与现代企业的发展不相符,并且难以适应激烈的市场竞争,最终被淘汰。在这个流量为王的时代,互联网明显更适合宣传产品,随意点击一个网站都可以看到铺天盖地的产品广告。因此当代零售企业纷纷搭建具有本公司文化的网站,利用在互联网技术快速的发展让自己的公司跟上时代的潮流^[4]。

1.2 研究意义

电子商务是一种新兴的,处于发展过程中的现代商务方式。电子商务活动是通过公共计算机通信网络进行商务活动的现代方式。该方式将突破传统商务在时间、地域的限制,成为方便、快捷、安全可靠的新兴电子化商务活动模式。电子商务的发展速度之快。远远超过我们的想象,普及速度之快令人难以置信,它不仅带给人们商业机会、利润空间,更改变了人们的生活及工作方式。没有了时间、空间和人为条件上的限制,人民的生活和工作将变得更加方便、灵活和自如,信息渠道更宽,信息传输更快。

本次毕业设计是以 java Web 开发为基础的网上购物网站开发作为本人的毕业设计主要课题。立足于设计一个在网络平台上运行的商城,以小小的力量推动中国互联网的发展。

1.3 发展状况

随着现代计算机信息相关技术的深入,广泛地应用到现代社会的许多领域并且发挥着不可替代的作用。企业信息管理的必然趋势一定会涉及到计算机信息技术。计算机软件作为计算机信息技术应用的一部分,尽最大限度的发挥计算机的相关优势,将大量的数据交给计算机系统来处理,相对于人工管理来说,计算机系统有所无法比拟的优点^[5]。例如:查找迅速、可靠性相当高、存储能力极强,不受空间时间约束、保密性好,由于加密技术的存在,重要信息泄露的概率小、使用寿命长、成本不高等。以上这些特点不仅能够使企业信息管理的效率提升,也能节约成本,是真正的合理利用社会有限的资源,也是企业信息管理数字化、正规化,科学化与全球数字经济接轨的重要条件^[8]。

1.4 研究的主要内容

针对网上商场系统的特点，此系统的设计内容如下：

设计内容分为三部分包括前端设计，数据库设计以及后端代码设计，具体内容如下：

（1）前端设计：通过综合考量，采用 JSP 技术，并综合运用 BootsStrap 前端框架优化前端界面。

（2）数据库设计：本次系统设计数据量不会太大，所以使用 mysql 数据库，该数据库可免费使用，并且足够满足系统需求。

（3）后端整体架构设计：本系统采用 SSM 框架进行开发。

其主要功能如下：

（1）用户的功能模块主要是实现了用户对系统账号注册、登录系统、系统公告、商品的浏览以及查询订购商品操作，在注册登录后，可以实现将物品加入购物车并提交订单等功能。

（2）后端信息管理模块主要是实现管理员的登录，以及登陆之后对公告的修改、商品的添加删除修改、用户的订单状态管理、以及删除相关订单，管理前台用户信息等功能。

（3）注册登录模块：进入系统主页首先默认是游客角色。可以浏览搜索商品，在注册登陆之后成为会员才能加入商品到购物车并提交订单。

（4）购买商品模块：成功登陆之后，才可以将商品加入到购物车，选择完商品之后可以继续浏览商品加入购物车或者直接提交购物车里的商品到订单。

（5）管理员模块：登录管理员账户之后可以对商品进行各种操作，以及对公告的修改，对订单状态的处理，对用户信息的维护等。

第 2 章 系统开发环境及项目开发工具

1.1 开发技术简介

1.1.1 JSP 简介

JSP 就是 Java 服务器页面的简称。JSP 是一种 HTML 扩展对象，在其上我们可以通过特定的方式，让我们直接使用 Java 代码来帮助我们实现页面功能。其中有几个使我们在 JSP 常用的编码方式，我们习惯于用 TL 表达式来帮助实现功能，而不直接使用 Java 代码来实现功能^[9]。

JSP 在 JSP 代码执行前创建了变量，因此在 JSP 代码内可以直接使用这些变量，此变量又名为隐含变量，创建这些变量的作用是为了使我们使用这些对象方便。下图可以帮助我们理解这两者的关系：

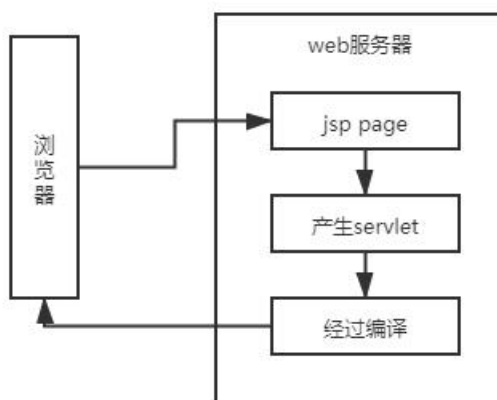


图 2-1 原理图

1.1.2 Servlet 相关简介

Servlet 全称是 java servlet,Servlet 是一种服务器端的 java 程序，具有独立性，它可以处理用户传过来的请求，也可以通过响应向用户输送数据。Servlet 就是处理用户的 HTTP 请求，然后回送 HTTP 响应，与传统的 java 程序不同，它由 web 服务器进行加载，该 web 服务器必须包含可以启动 servlet 的 jvm^[10]。

1.1.2 MySQL 相关简介

MySQL 数据库是一种小型数据库，它一般用于小型软件开发或程序员初学者进行学习的，不过我们都是使用 5.0 版本的，因为 6.0 之后版本的 MySQL 都是要钱的。MySQL 数据库本质是一种文件系统，我们一般都是用 SQL 语句来对它进行操作，从而对数据进行。MySQL 数据库是一种关系型数据库的管理系统，关系型数据库是指在不同的表中保存数据的数据库。它保存关系数据模型，即保存数据也保存数据的关系，文件与文件也有关系。因此关系型数据库是最常用的，它的特性是数据操作慢和安全性高。

MySQL 也是学校所教的数据库语言之一，我们常用 Navicat Lite 对 MySQL 进行操作，因为 Navicat Lite 可以简化数据库操作，它还可形象生动的展示数据的总体内容，并且还可以直接通过他操作数据库。例如：添加表和字段值等、修改表、表名和表中内容等。通常都会使用它来进行数据库实现，这是一个很方便的工具^[11]。

1.1.4 IDEA 简介

IDEA 全称 IntelliJ IDEA，是用于 java 开发的集成环境（也可用于其他语言），它支持代码的提示、JUnit 单元测试。IDEA 的开发效率不是一般的快，尤其是对于 java 语言，它容纳了开发者在开发中需要的绝大部分实用功能，可以让开发者脱离鼠标，做他们想做的事，尽最大的能力加快开发效率^[12]。Idea 的优点如下：

- （1）对于 JUnit 单元测试的完美支持。
- （2）不需要任何的插件，完全支持 JSP。
- （3）使用者众多，意味着社区它的开发者社区活跃，遇到与之相关的问题可以很快解决。
- （4）Idea 对项目版本控制的支持是绝无仅有的，无论是 SVN 还是 Git，亦或是 GitHub 都支持的非常好。尤其是对 Git 的支持，图形化的操作页面，会让你省却很多命令的烦恼。让普通的程序开发者在编程过程中直接在 IDEA 中就能完成代码的提交、拉取，查看版本控制服务器内容等等。
- （5）内嵌了 Maven 插件，而且也内嵌了 Maven，当然也可以使用外部的 Maven。
- （6）Idea 会把整个工程的文件都做 Index，所以搜索文件什么，都是毫秒级别就出来的。
- （7）我们还可以有选择地重新编译和加载部分字节码文件，再加上 IDEA 强大的 DEBUG 功能，可以很方便地帮助我们进行代码的调试工作。

1.1.5 J2EE

J2EE 是一种中间件的基础架构，由于它的存在，开发者只用花费精力去编写业务逻辑的代码，至于其他的系统深层问题，比如内存管理，gc，线程管理等都由系统本身完成。

J2EE：它将应用开发划分为多个不同的层次结构，并在每一个层上定义组件，非常典型的 J2EE 包括客户层，网页显示层，业务逻辑层，信息管理层。

1.2 系统运行环境

（1）服务器端：本系统运行在 windows 操作系统之上，项目采用部署本地服务器方式运行，服务器选用 Tomcat8，数据库采用 MySQL 5.7。

（2）客户端：google 浏览器。

第 3 章 需求分析

3.1 可行性分析

3.1.1 经济可行性分析

缩短了供货周期，压缩了库存，裁减了人员，就可以依据缩短时间的天数和库存物资的减少，以及裁减人员的工资使原材料和劳动力资源得到更合理的利用，从而降低了成本，避免了过多的消耗，减少管理费用的支出等等。所以此项目在经济上是可行的。

3.1.2 技术可行性分析

Java 是一种优秀的高级编程语言，无论是在 windows 系统还是 linux 系统，他都可以进行代码的开发，它最大的优点就是与平台无关。目前 J2EE 技术已经非常成熟，结合 jsp, servlet 及相关的关系数据库技术，完全可以达到并实现相关功能^[13]。

3.1.3 社会可行性分析

无论是国家还是当地政府都支持传统企业进军电子商务，出台了一系列优惠政策，希望以网络销售带动门店销售，并对网购平台诚信体系，支付手段等方面予以支持和政策指导，以推进实体市场和网上市场的结合。对于消费用户，会根据消费用户的调查报告设计系统会考虑到人性化界面和操作的方便性。所以社会可行性方面没有问题^[14]。

3.2 系统需求分析

通过对软件工程的理论的学习，系统的相关分析和设计是一个项目工程开发能否成功的重点，正确获取系统用户的需求为项目开发提供了基础^[15]。此网上购物系统的重点是提供产品展示，以及商品订购，是宣传自己商店的商品给用户，让客户能够通过自己的需要来选择要购买的商品：

因为传统购物存在一定的局限性和问题，所以经过总结，有下面几个需求：

顾客需求：

- (1) 普通用户可以通过系统浏览商品。
- (2) 普通用户可以通过系统搜索框按商品名称搜索商品。
- (3) 普通用户可以注册成为会员。用户可以注册成为会员，并可以登陆进入网站拥有更多功能。
- (4) 会员可以浏览和检索商品。
- (5) 会员可以加入购物车；购物车必须会员才可以操作。
- (6) 会员可以浏览并且管理自己的购物车，根据需求来添加商品或删除商品。
- (8) 会员可以浏览并且管理自己的订单。
- (9) 会员可以管理自己的账户信息，修改密码。

商场管理员的需求：

- （1）商场管理员可以对系统公告进行修改，发布。
- （2）商场管理员可以对顾客，商品，订单进行管理和维护。

购物车的需求：

- （1）查看购物车的内容。
- （2）清空所有商品的功能。
- （3）全选和选部分商品提交。

系统优势，方便用户，用户只用短短的几步便可以买到心仪的商品，极大限度的提高了网上商城的运营效率。搜索功能使用户在极短的时间便可以查询到自己想要的商品，节省了选购时间，在提高用户体验的同时，也降低了服务器压力。

3.3 结构化分析

3.3.1 数据流图

本系统的数据流图如下所示：

首先是顶层图，它主要说明了系统运行的边界。

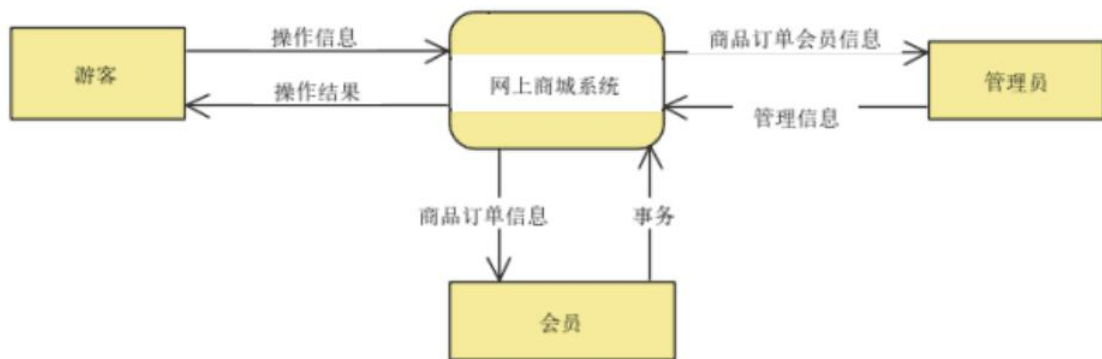


图 3-1 顶层图数据流图

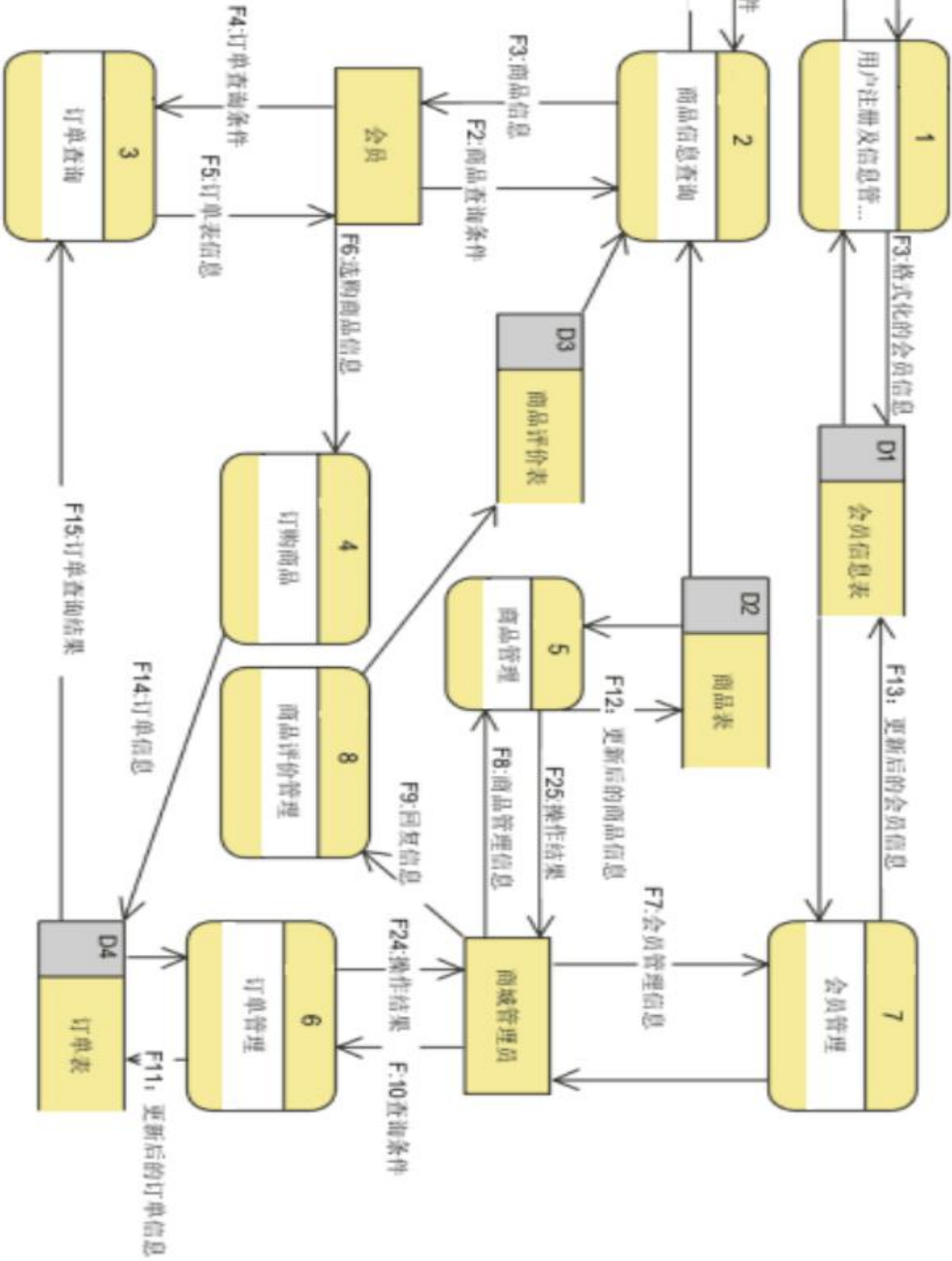


图 3-2 1 层数据流图

第2层数据流图是对上层的详细说明和拆分,更清楚的描述了系统数据的处理情况。如下所示:

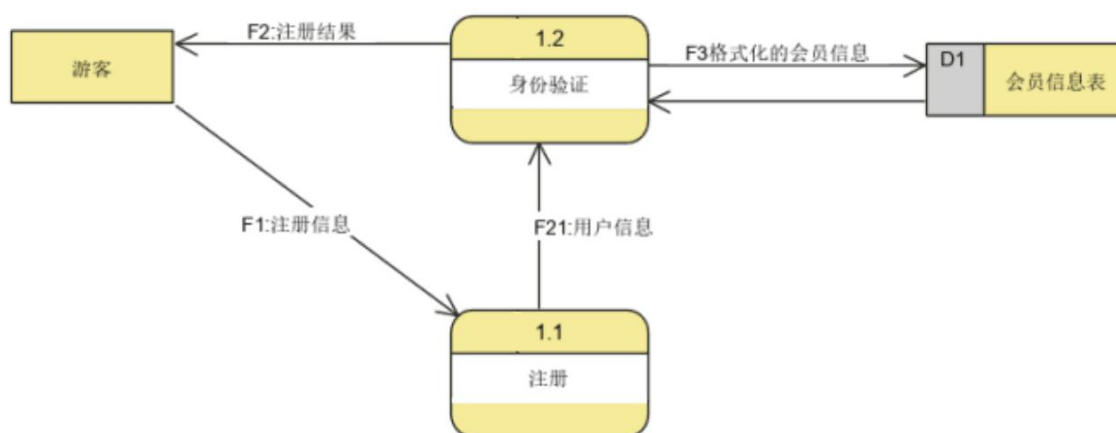


图 3-3 用户注册及信息管理 2 层数据流图

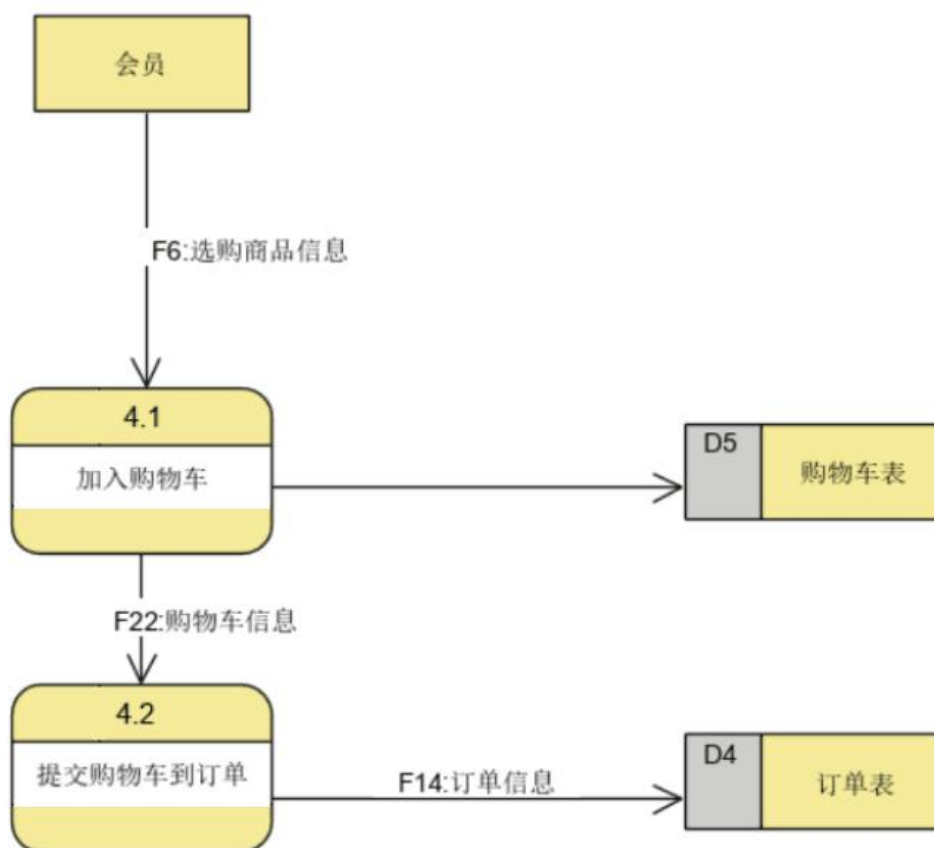


图 3-4 订购商品 2 层数据流图

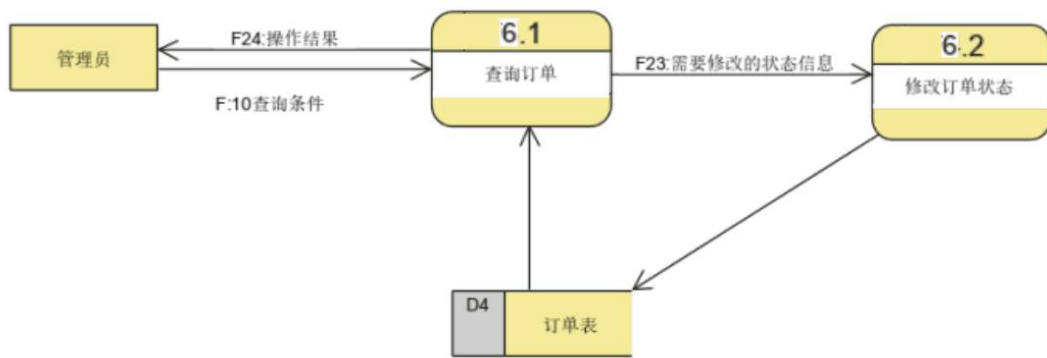


图 3-5 订单管理 2 层数据流图

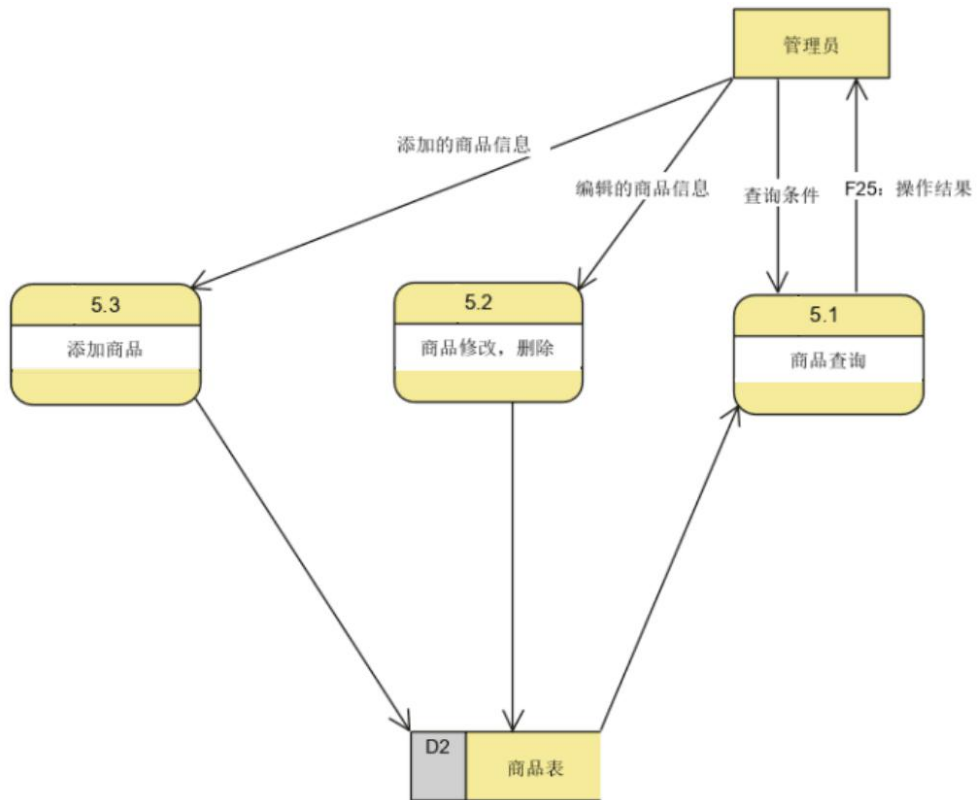


图 3-6 商品管理 2 层数据流图

3.3.2 数据字典

以下列出本系统的主要数据字典条目。

(1) 数据结构

数据结构名：用户

含义：系统的主体数据结构，定义了一个用户的有关信息。

组成：用户 id、用户名、密码、姓名、邮箱、电话号码、性别、生日。

数据结构名：管理员

含义：系统的主体数据结构，定义了一个管理员的用户名与密码

组成：管理员用户名、密码

(2) 数据项

1) 用户 id

数据项名：用户 id

数据项含义：唯一标识每一个用户

别名：用户编号

数据类型：字符型

取值范围：随机

2) 用户名

数据项名：用户名

数据项含义：用户的账号名称

别名：用户名称

数据类型：字符型

取值含义：手机号

3) 密码

数据项名：密码

数据项含义：账户密码

别名：口令

数据类型：字符型

4) 姓名

数据项名：姓名

数据项含义：用户的真实名字

别名：名字

数据类型：字符型

取值含义：用户名字

5) 邮箱

数据项名：邮箱

数据项含义：用户的邮箱

别名：电子邮箱

数据类型：字符型

6) 电话号码

数据项名：电话号码

数据项含义：用户的手机号

别名：手机号

数据类型：字符型

7) 性别

数据项名：性别

数据项含义：用户的性别

数据类型：字符型

8) 生日

数据项名：生日

数据项含义：用户的出生日期

别名：出生日期

数据类型：字符型

9) 管理员用户名

数据项名：管理员用户名

数据项含义：管理员的账号

别名：管理员的账号

数据类型：字符型

(3) 数据流定义：

1) 名称：注册信息

编号：F1

简述：未注册的用户提供自己的信息注册账号

数据来源：顾客

数据去向：会员信息表

数据组成：用户名+密码+姓名+电话+邮箱+性别+生日

2) 名称：商品查询条件

编号：F3

简述：用户查询商品输入的查询条件

数据来源：用户

数据去向：商品查询

数据组成：商品名

4) 名称：订单表信息

编号：F5

简述：所属这个用户的所有订单信息

数据来源：订单表

数据去向：用户

数据组成：用户名+总金额+收货信息

4) 名称：订购信息

编号：F6

简述：用户选择的商品的信息

数据来源：用户

数据去向：购物车表

数据组成：用户名+商品 id+数量

5) 名称：会员管理信息

编号：F7

简述：管理员对会员信息的修改或删除

数据来源：管理员

数据去向：会员管理

数据组成：会员信息表除用户名之外的各项信息

6) 名称：查询条件

编号：F10

简述：管理员搜索订单输入的条件

数据来源：管理员

数据去向：订单表

数据组成：订单号

7) 名称：订单信息

编号：F14

简述：用户提交的订单信息

数据来源：用户

数据去向：订单表

数据组成：购物车信息+收货信息+用户名

8) 名称：购物车信息

编号：F22

简述：用户提交购物车到订单

数据来源：用户

数据去向：订单表

数据组成：用户名+数量+总金额+商品 id

9) 名称：添加的商品信息

编号：F26

简述：管理员添加商品

数据来源：管理员

数据去向：商品表

数据组成：一个商品的完整信息

10) 名称：查询条件

编号：F28

简述：管理员搜索商品输入的条件

数据来源：管理员

数据去向：商品表

数据组成：商品的名字

(4) 数据存储

1) 数据存储编号：D1

数据存储名称：会员信息表

文件组成：用户 id, 用户名, 密码, 姓名, 电话, 邮箱, 性别, 生日

说明：记录会员所有信息

来源：会员信息

2) 数据存储编号：D2

数据存储名称：商品表

文件组成：商品 id, 商品名, 商品图片, 商品描述, 商品价格, 商品上架 日期, 商品库存, 商品销量, 商品类别

说明：记录商品信息

来源：商品信息

3) 数据存储编号：D4

数据存储名称：订单表

文件组成：订单 id，用户名，收货人，收货地址，收货人电话，下单时间，订单状态。

说明：记录所有的订单信息

来源：订单信息

4) 数据存储编号：D3

数据存储名称：商品评价表

说明：记录商品的评论信息

来源：用户评价商品信息

组成：订单子项编号、数量、总计、所属订单号、商品 id

5) 数据存储编号：D5

数据存储名称：购物车表

文件组成：购物车 id，用户名，商品 id，数量

说明：记录个人的购物车信息

来源：购物车信息

(5) 处理过程

1) 用户注册

处理过程名：注册

说明：无账号的游客根据个人信息注册账号

输入数据流：F1

输出数据流：F21

处理：游客根据自己的信息注册账号，密码成为会员

2) 商品信息查询

处理过程名：商品信息查询

说明：用户根据商品名查询商品

输入数据流：F2

输出数据流：F3

处理：系统根据输入的信息查询对应的商品给用户浏览

3) 加入购物车

处理过程名：加入购物车

说明：用户挑选商品加入购物车

输入数据流：F6

输出数据流：F22

处理：会员将心仪的商品添加到自己的购物车里

4) 订单查询

处理过程名：订单查询

说明：根据用户名查询该用户所有的订单信息

输入数据流：F4

输出数据流：F5

处理：系统通过登录的用户名查询该用户的订单信息提供给用户

5) 添加商品

处理过程名：添加商品

说明：系统管理员通过后台添加新商品

输入数据流：F26

处理：系统通过后台添加新商品到数据库中

6) 编辑商品

处理过程名：编辑商品

说明：系统管理员通过后台编辑已存在的商品信息

输入数据流：F27

处理：系统管理员通过后台修改商品的各种信息

7) 修改订单状态

处理过程名：修改订单状态

说明：系统管理员根据订单状态处理订单

输入数据流：F23

处理：系统管理员通过后台修改订单的状态（已付款的订单去发货）

第 4 章 系统设计

4.1 系统模块设计

在软件系统设计中，模块一般情况下都是按对应的功能划分的通常称为功能模块。因为这样做可以保证系统的数据稳定性，以及降低模块之间的耦合度。合理的功能模块划分不但能够提高开发者的开发效率，并且对于一些相似的模块组件可以复用，从而达到了减少开发系统时间的目的。而且这样做也有利于系统后期的维护^[16]。本系统分为两个部分，即前台部分主要面向顾客和后台部分面向管理员。

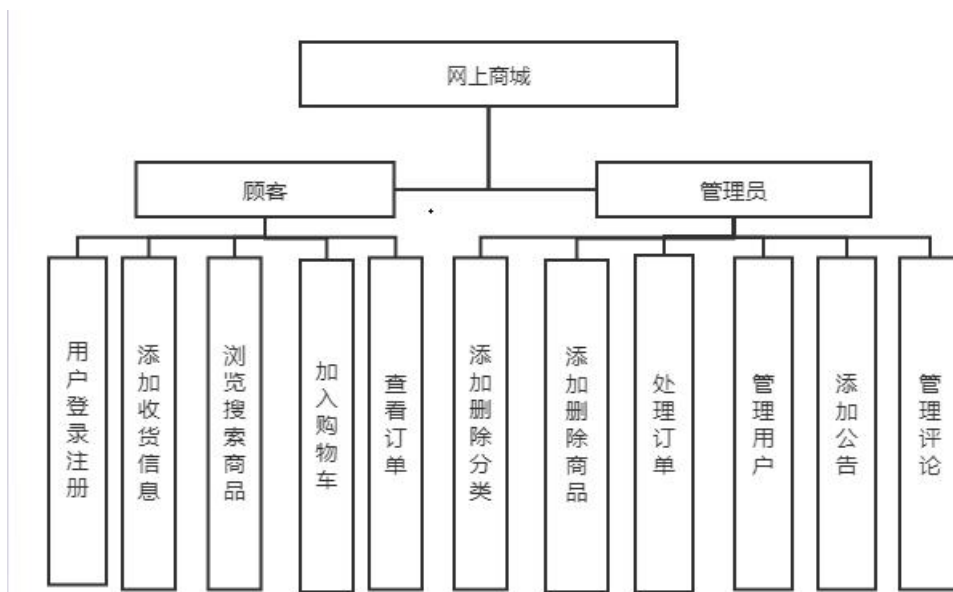


图 4-1 系统功能图

4.1.1 前台面向用户的功能模块设计

前台模块主要为用户提供系统账号的注册，登录，浏览商品信息，购买商品，填写并提交订单的服务。这样，将前台部分在细分为用户管理模块，商品查询模块，购物车模块，添加收货信息模块，订单模块。顾名思义，这些模块的名称基本解释了他们的功能。下面详细说明一下：

商品推荐浏览模块：网站的主页向顾客推荐的商品有两大类，一种是热门商品，主要是根据商品的销量的排名来显示相关商品给顾客。另外一种是以商品的上架的时间来显示商品给顾客浏览。

用户管理模块：为了方便网站的管理，必须由一套完整的用户管理体系，该网站用户管理模块主要实现用户的注册，登录，修改密码等功能。

购物车模块：在网上商城中，通常都会采用一种被称为购物车的技术来模拟显示生活，购物车应该一直和用户绑定的，无论什么时候用户在登陆之后都可以看到他原来添加的商品，在网上商城中所选商品必须通过购物车进行暂存，然后生成订单。可以随时添加查看修改清空购物车中的内容，他都会持久化数据到数据库中。

商品搜索模块：用户根据商品的名称可以搜索到所需要的商品信息。

添加收货信息模块：用户在登录之后添加个人的收货地址的相关信息，也可以删除收货地址。

订单模块：用户在购物车中选择自己需要商品提交订单，不需要的可以留在购物车里，可以查看订单状态，确认收货等等。

4.1.2 后台模块的功能模块设计

后台管理模块的各个功能如下：

管理员身份验证模块：为系统管理员提供一个后台系统的接口，该接口的功能是验证管理员的身份。输入登录账号和密码，如果通过验证则跳转后台主页，否则提示错误。

商品管理模块：

向商品表插入商品信息，也可以修改信息，也可以删除，向商品类别表添加商品的类别，删除商品的类别。

处理订单模块：管理员对用户订单的执行，修改订单状态，查看及其对已完成的订单的删除。

用户信息管理模块：管理顾客用户的信息。

公告管理模块：管理系统公告。

评论模块：顾客在已完成的订单中可以评价商品，管理员可以回复该评论信息。

4.2 系统流程设计

本系统分为前台购物流程和后台订单处理流程。

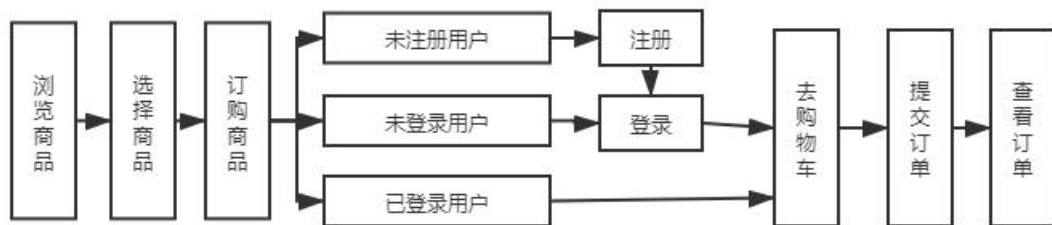


图 4-2 前台购物流程图

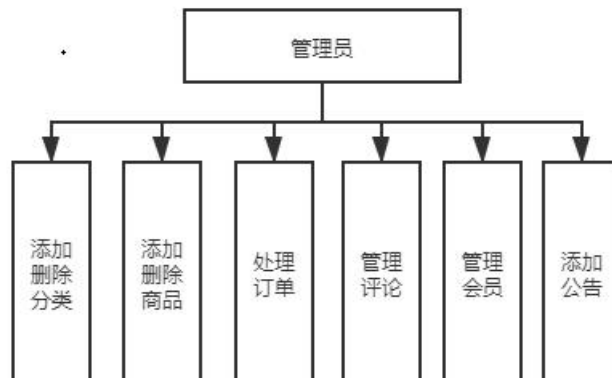


图 4-3 后台管理流程图

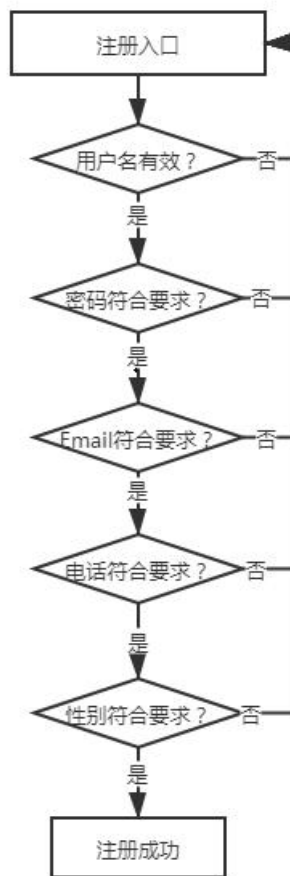


图 4-4 注册功能流程图



图 4-5 登录流程图

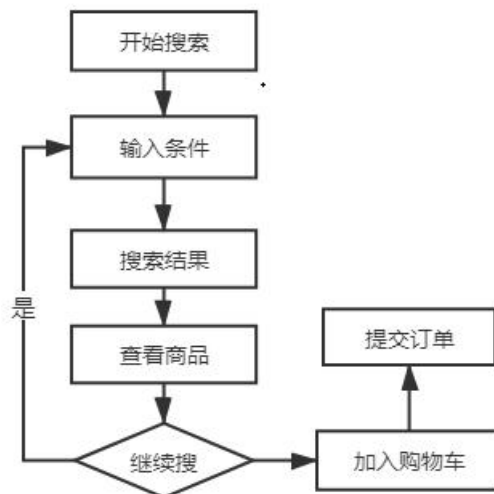


图 4-6 商品搜索购买流程图

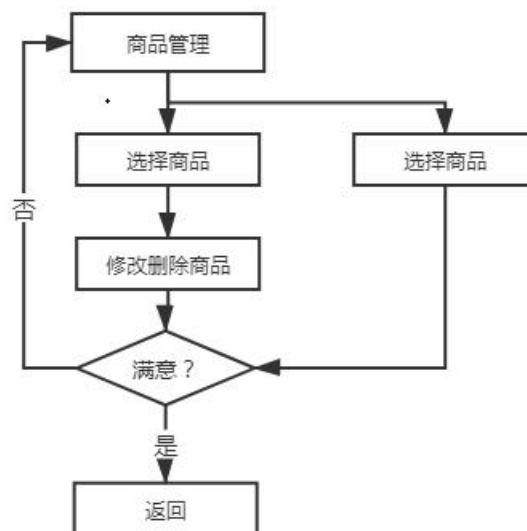


图 4-7 商品管理流程图

4.3 数据库设计

4.3.1 数据库概要设计

概念结构是许多模型的基础，相对于其他模型，它更独立与机器，更抽象，更稳定。概念结构极其重要，在实际中经常使用 ER 图来描述概念模型^[17]。

实体分析：根据网上商城的总体设计，我们可以首先确定该系统中的实体：管理员、会员、商品、订单、地址信息，公告共 6 个实体。其实体属性图，和 ER 图如下：

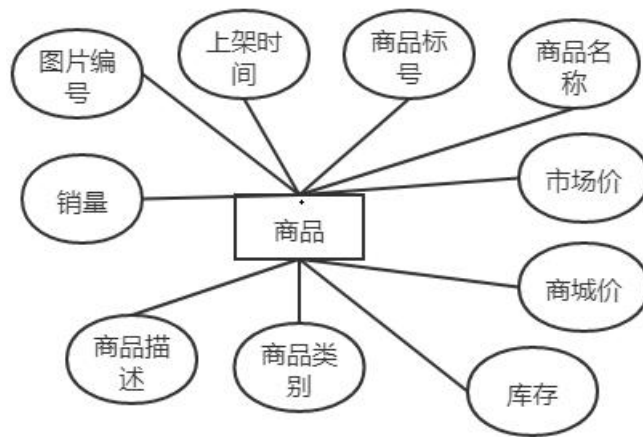


图 4-8 商品的实体属性图

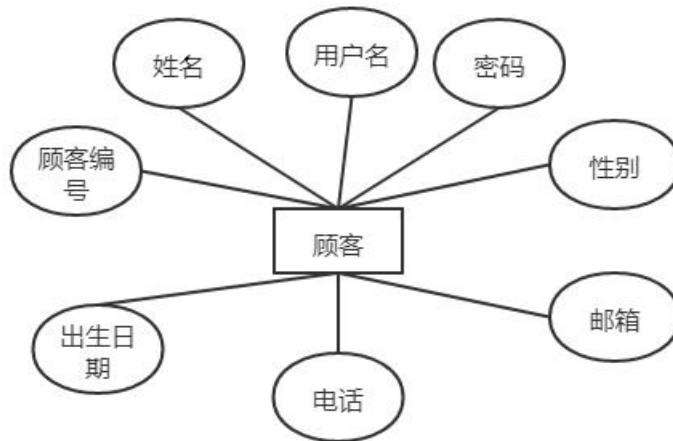


图 4-9 顾客的实体属性图

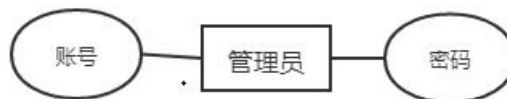


图 4-10 管理员的实体属性图

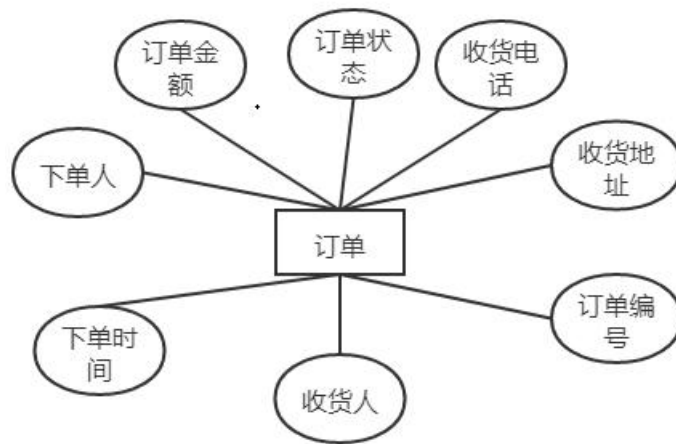


图 4-11 订单的实体属性图

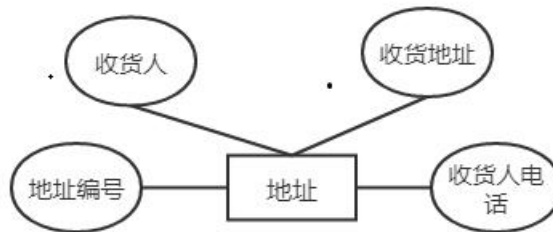


图 4-12 收货地址的实体属性图



图 4-13 购物车的实体属性图

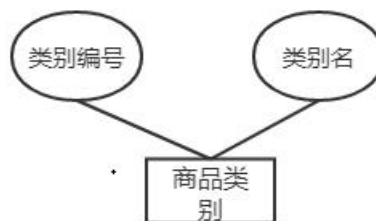


图 4-14 商品类别的实体属性图

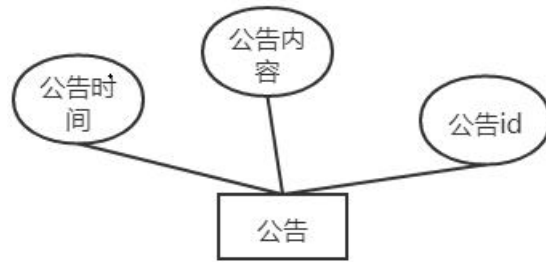


图 4-15 系统公告的实体属性图

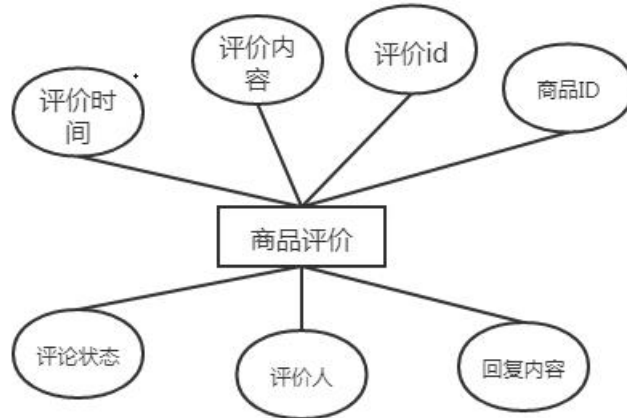


图 4-16 商品评价的实体关系图

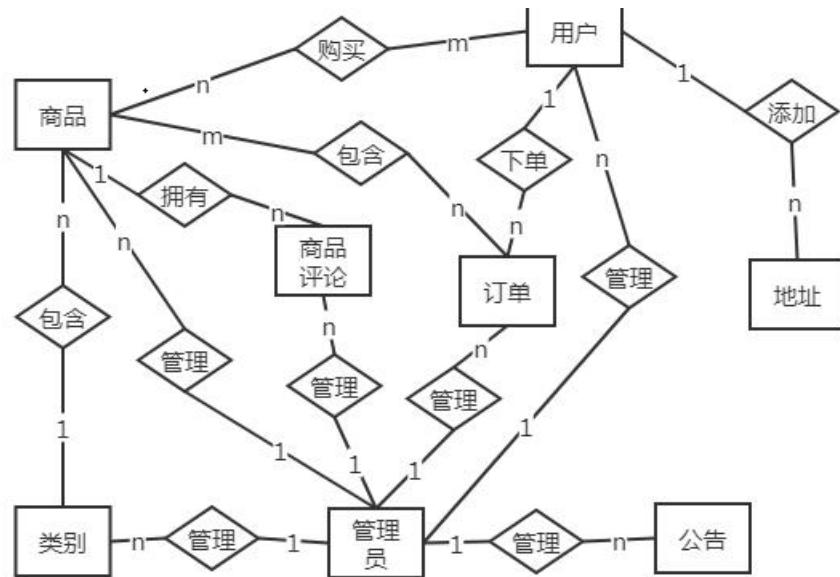


图 4-17 ER 图

4.3.2 数据库表

在本系统的数据库中，根据需求建立了如下基本表，具体如下

(1) 用户表

User(uid,username,password,name,email,telephone,sex,birthday)

(2) 收货信息表

Adress(aid,address,phone,person)

(3) 商品表

Product (pid,pimage,pname,pdesc,pdate,pmarket_price,pcount,is_latest,cid)

(4) 订单表

Orders (oid,address,phone,person,ordertime,state,total,username)

(5) 管理员表

admin (acount,pwd)

(6) 分类表

Category (cid,cname)

(7) 购物车表

Car (cid,count,pid,uid)

(8) 订单条目项表

Orderitem (itemid,oid,pid,subtotal,count)

(9) 评论表

Goodscomments (gid,comment,username,pid,state,anser)

根据以上分析数据库表结构如下：

数据库名：store

表 4-1 表名 user

序号	列名	数据类型	长度	主键	允许空	默认值	说明
1	Uid	varchar	255	是	不		用户的唯一 id
2	username	varchar	20		不		登录名字
3	password	varchar	10		不		用户密码
4	name	varchar	20				用户姓名
5	email	varchar	50				邮箱
6	telephone	varchar	20				电话
7	sex	varchar	10				性别
8	birthday	varchar	50				生日

表 4-2 表名 product

序号	列名	数据类型	长度	主键	允许空	默认值	说明
1	pid	varchar	255	是	不		商品唯一 id
2	pname	varchar	100		不		商品名称
3	pimage	varchar	255		不		商品图片
4	pdate	varchar	100		不		上架日期
5	pdesc	varchar	1000				商品描述
6	Shop_price	double	10		不		商品价格
7	Market_price	double	10		不		超市价格
8	pcount	int	11		不		库存
9	Is_latest	int	11		不		销量
10	cid	varchar	255		不		外键

表 4-3 表名：orders

序号	列名	数据类型	长度	主键	允许空	默认值	说明
1	Oid	varchar	255	是	不		订单唯一 id
2	address	varchar	100		不		该订单地址
3	name	varchar	20		不		收件人
4	telephone	varchar	20		不		电话
5	ordertime	data	20		不		订单时间
6	state	int	10		不		订单状态
7	total	double	50		不		总金额
8	username	varchar	20		不		外键

表 4-4 表名: orderitem

序号	列名	数据类型	长度	主键	允许空	默认值	说明
1	Itemid	varchar	255	是	不		订单的一个购物项
2	count	int	11		不		数量
3	subtotal	double	10		不		小计
4	oid	varchar	255		不		外键受订单表 oid 约束
5	pid	varchar	255		不		外键受商品表 oid 约束

表 4-5 表名: car

序号	列名	数据类型	长度	主键	允许空	默认值	说明
1	cid	varchar	255	是	不		购物车唯一 id
2	uid	varchar	255		不		外键
3	Pid	varchar	255		不		外键商品 id
4	count	int	11		不		数量

表 4-6 表名: admin

序号	列名	数据类型	长度	主键	允许空	默认值	说明
1	Acount	varchar	10	是	不		管理员账户
2	pwd	varchar	10		不		管理员密码

表 4-7 表名: notice

序号	列名	数据类型	长度	主键	允许空	默认值	说明
1	nid	varchar	255	是	不		公告 id
2	notice	varchar	255		不		公告内容
3	time	date	0		不		增加公告的时间

表 4-8 表名: category

序号	列名	数据类型	长度	主键	允许空	默认值	说明
1	cid	varchar	255	是	不		分类的 id
2	cname	varchar	255		不		分类名称

表 4-9 表名: goodscomments

序号	列名	数据类型	长度	主键	允许空	默认值	说明
1	gid	int		是	否		主键
2	username	vachar	20		否		用户名
3	comment	vachar	255		否		
4	time	date	0		否		
5	pid	vachar	255		否		商品 id
6	anser	vachar	255				评论回复
7	state	int	10		否	0	

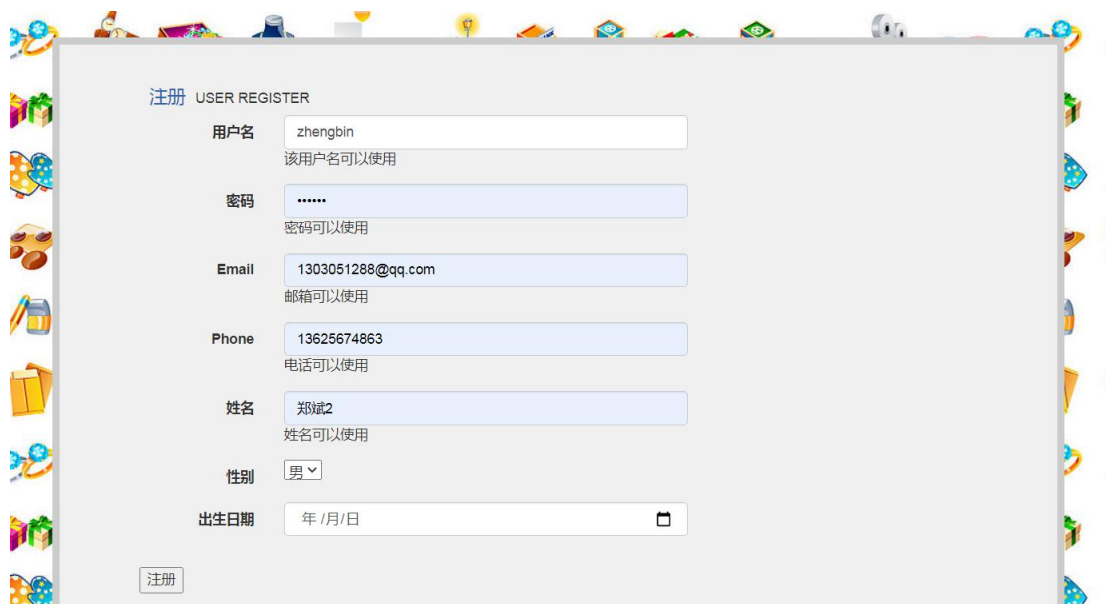
第5章 系统实现

5.1 前台设计

网站前台部分提供给用户使用，是系统对外的窗口。主要包括：网站首页、查看公告，产品展示，用户登录注册，购买商品，查看订单。由于篇幅有限，本文只对部分模块的实现进行描述。

5.1.1 用户注册

游客通过注册页面填写个人信息进行注册，每项必要信息都必须符合要求才能注册成功，注册成功即成为会员。否则注册失败。



The screenshot displays a web browser window with a desktop background of colorful icons. The browser's address bar shows a URL starting with 'http://'. The page title is '注册 USER REGISTER'. The form contains the following fields and feedback:

- 用户名 (Username):** Input field contains 'zhengbin'. Feedback: '该用户名可以使用' (This username can be used).
- 密码 (Password):** Input field contains '.....'. Feedback: '密码可以使用' (Password can be used).
- Email:** Input field contains '1303051288@qq.com'. Feedback: '邮箱可以使用' (Email can be used).
- Phone:** Input field contains '13625674863'. Feedback: '电话可以使用' (Phone number can be used).
- 姓名 (Name):** Input field contains '郑斌2'. Feedback: '姓名可以使用' (Name can be used).
- 性别 (Gender):** A dropdown menu with '男' (Male) selected.
- 出生日期 (Birth Date):** A date picker showing '年/月/日' (Year/Month/Day).

A '注册' (Register) button is located at the bottom left of the form.

图 5-1 用户注册

5.1.2 用户登录

用户登录页面主要用来接收用户输入的用户名和密码，并更新用户在网站中的状态信息。只有当用户名和密码都正确才能登陆成功，如果输入不正确，系统会显示登录失败。会员登录窗口的运行结果如图所示。

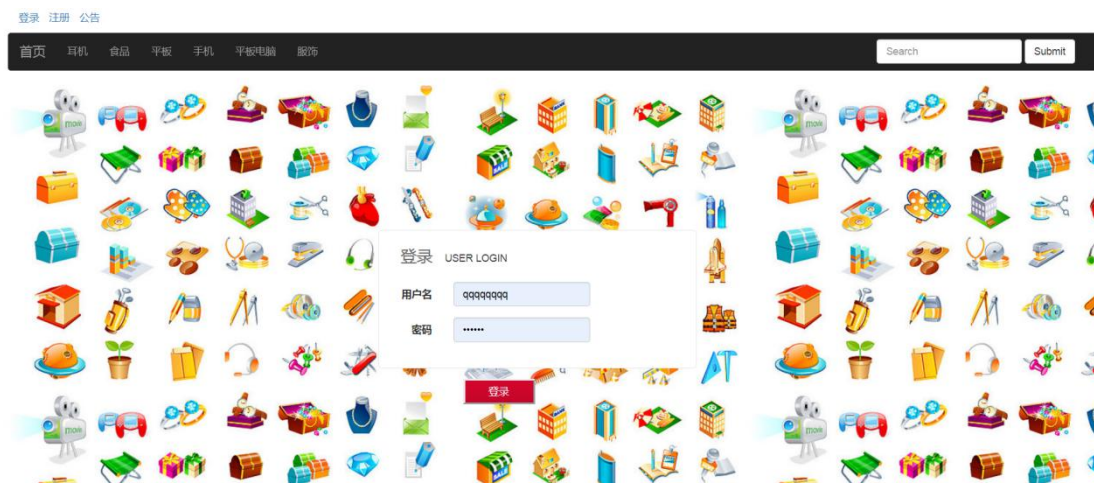


图 5-2 用户登录

//登陆验证

```
boolean isSuccess = service.findUserByUsernameAndPassword(user);
if(isSuccess) {// 成功
    mav.addObject("msg", "登录成功！， 请前往首页选购");// 添加到session
    session.setAttribute("user", user);
    mav.setViewName("index");
    return mav;
} else {// 失败
    mav.addObject("msg", "登录失败");
    mav.setViewName("msg");
    return mav;
}
```

5.1.3 商品推荐及信息浏览

用户在商品显示页面对具体商品进行详细了解，可以查看商品的评论等，查看已买过此商品的顾客的评价。并在商品显示页面对商品进行订购，已注册登录的会员可以加入购物车并提交订单。

网站主页有热门商品推荐和最新商品推荐，热门商品是根据商品销量的数量来显示排列的，最新商品是按照上架时间的先后来排列显示的。

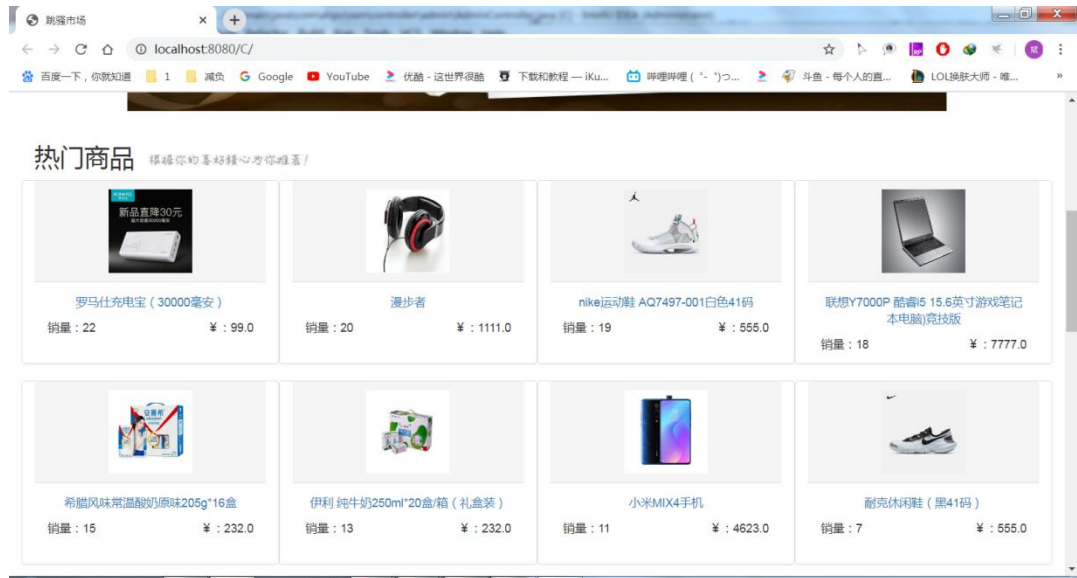


图 5-3 首页


```
//发送请求，获取最新商品信息
$.getJSON("${pageContext.request.contextPath}/latestProduct.action",function(result)
{
    $(result).each(function(index,element) {
        $("#latest").append("<div class='col-md-2' ><a
href='${pageContext.request.contextPath}/infoProduct.action?pid="+element.pid+"><im
gsrc='${pageContext.request.contextPath}/pic/"+element.pimage+" ></a><p><ahref='${pa
geContext.request.contextPath}/infoProduct.action?pid="+element.pid+">" +element.pnam
e+"</a></p>+element.shop_price+"</font></p></div>")
    })
})
})
@RequestMapping("/hotProduct")
public @ResponseBody List<Product> findHotProduct(){
    return service.findHotProduct();
}
```

```
}
```

```
@RequestMapping("/latestProduct")
public @ResponseBody List<Product> findlatestProduct(){
    return service.findlatestProduct();
}
```

单个商品的详细信息页面：

首页 > 手机 > 小米MIX4手机



小米MIX4手机

促销价: ¥ : 4623.0元/份


促销: 限时抢购

库存: 999

购买数量 (最多10件):

加入购物车

商品介绍



商品信息

主体 入网型号 以官网信息为准 产品名称 小米10青春版 上市年份 2020年 上市月份 4月 基本信息 机身长度 (mm) 164.02mm 机身重量 (g) 192g 机身材质工艺 以官网信息为准 机身宽度 (mm) 74.77mm 机身材质分类 其他 机身厚度 (mm) 7.88mm 运营商标志或内容 其他 主芯片 CPU品牌 高通(Qualcomm) 屏幕 屏幕材质类型 AMOLED 屏幕刷新率 其他 主屏幕尺寸 (英寸) 6.57英寸 后置摄像头 后摄主摄光圈 f/1.79 闪光灯 双LED灯 前置摄像头 前摄主摄光圈 其他 电池信息 电池是否可拆卸 电池不可拆卸 充电器 其他 无线充电 不支持无线充电 网络支持 最大支持SIM卡数量 2个 网络频率 (2G/3G) 以官网信息为准 5G网络 移动 5G; 联通5G; 电信5G 4G网络 4G FDD-LTE(联通); 4G TD-LTE(移动); 4G FDD-LTE(联通、电信)

图 5-4 商品信息图

未注册的用户想要登录系统必须要注册账户，只有所有的数据项符合要求才能注册成功，只要有一项不符合，便不能注册成功。

图 5-5 注册

用户注册模块

主要实现代码：

```
<script>
function m() {
    if($("#username").text()=="该用户名可以使用"&&$("#ptext").text()=="密码可以使用"&&$("#pemail").text()=="邮箱可以使用"&&$("#pphone").text()=="电话可以使用"&&$("#pname").text()=="姓名可以使用"){
        $(".form-horizontal").submit();
    } else {
        alert("请检查所有注册项符合要求！")
    }
}
}
</script>
```

5.1.4 添加商品到购物车

如果商品库存足够，用户就可以看见加入购物车的按钮，如果库存不足，就会看见库存不足。在商品加入购物车后，用户可以修改数量，另外，用户可以选择部分购物车内的商品提交到订单，而没被选中的商品会留在购物车内，最后也有删除所有商品的按钮，和删除单个商品的按钮。

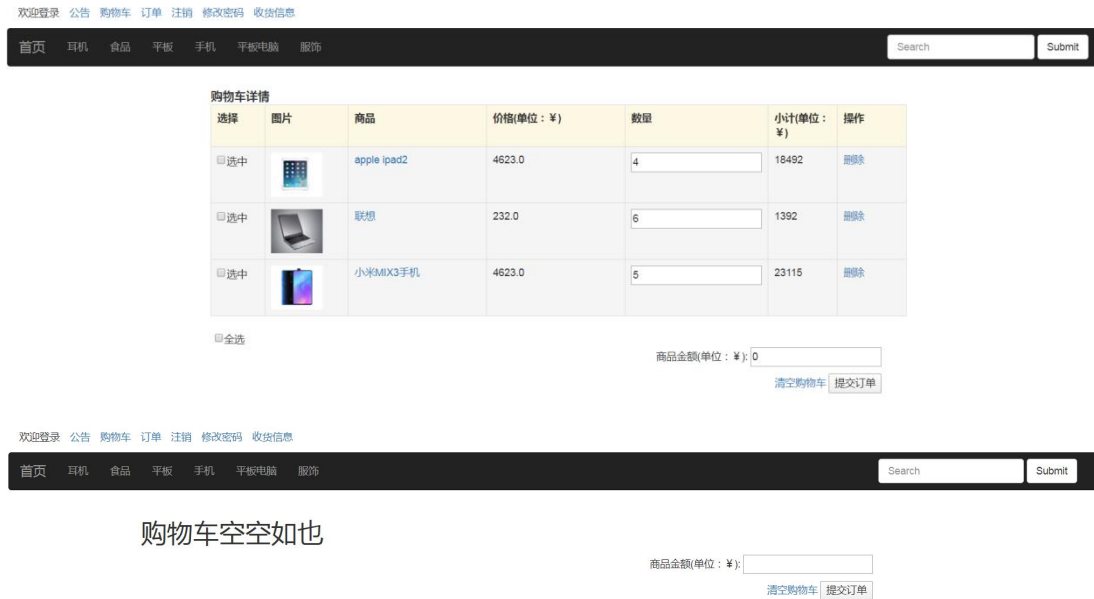


图 5-6 购物车

实现代码:

```
User user=(User)session.getAttribute("user");
//通过 pid 找到商品内容
Product p = service.findProductByPid(pid);
//将 count 加入到购物车中
Cart cart=new Cart();
if(service1.isIncart(cart)){
    service1.updateCart(cart);
}else {service1.insetcart(cart);
}
}
}



List<Cart> carts=service1.findCart(user);
ModelAndView mav=new ModelAndView();
mav.addObject("carts",carts);
System.out.println(carts);
mav.setViewName("cart");
return mav;
}
```

5.1.5 订单相关页面

当用户从购物车选中商品提交到订单时, 订单开始生成, 此时订单处于未付款状态, 这时候用户可以退出, 在订单页面可以看到订单状态, 只有当用户确认订单之后, 订单的状态才是已付款等待发货的状态。

订单详情

订单编号:bed0286b925645ba86e47326cc3825a2

图片	商品	价格	数量	小计
	apple	¥ 4623.0	5	¥ 23115.0
	小米MIX4手机	¥ 4623.0	5	¥ 23115.0

商品金额: **¥ 46230.0元**




收 货 人:




收 货 地 址:



收货人电话:

[新增收货信息](#)

选择银行:

☒ 工商银行 
☐ 中国银行 
☐ 农业银行 

☐ 交通银行 
☐ 平安银行 
☐ 建设银行 

☐ 光大银行 
☐ 招商银行 

[确认订单](#)

我的订单

订单编号	订单地址	收件人	电话号码	订单状态	订单时间
19cb68ff3af040b88c1143944c5dccb7				去付款	2020-05-02 订单详情
304d162920324dff8e6c9ddaf3d53937				去付款	2020-05-02 订单详情
7e9ba6ad3996497d8f2d3fac07fa25db				去付款	2020-05-02 订单详情
bed0286b925645ba86e47326cc3825a2	安徽合肥	郑斌	13625674863	已付款 (总金额46230.0) 等待发货	2020-05-02 订单详情
cf8c7ff351d94c2ba98fd7cea8fd062a				去付款	2020-05-02 订单详情

1 2 »

图 5-7 订单

代码实现:

Java 部分:

```

System.out.println(1);
List<Cart> carts=new ArrayList<Cart>();
for(String a:arr){
    service.findcid(a);
    carts.add(service.findcid(a));
}

User loginUser = (User)session.getAttribute("user");
if(loginUser == null) {
    mav.addObject("msg", "请先登录, 再继续购买! ");
    mav.setViewName("login");
    return mav;
}

```

```

}
List<Address> list=new ArrayList<>();
list=service1.getlist(loginUser);
return mav;
}

```

5.2 后台实现部分

5.2.1 商品类型管理

管理员在登录管理系统之后，可以修改已经存在的商品分类，添加商品分类，删除已存在的商品分类。

分类列表			
序号	分类名称	编辑	删除
1	耳机		
2	食品		
3	平板		
4	电器		
5	手机		
6	平板电脑		
7	服饰		

[添加](#)

添加分类	
分类名称：	<input type="text"/>
<input type="button" value="确定"/> <input type="button" value="重置"/> <input type="button" value="返回"/>	

图 5-8 编辑分类

代码实现：

Java 部分：

```

c.setCid(UUIDUtil.getUUId());
System.out.println(c);
service.addCategory(c);
mav.addObject("msg", "添加成功");
mav.setViewName("admin/welcome");
return mav;
}

service.deleteCategory(c);
mav.addObject("msg", "删除成功");
mav.setViewName("admin/welcome");
return mav;
}
}

```

5.2.2 商品管理

管理员在登录系统后可以查看所有的商品，通过搜索栏搜索商品，然后对想要更改的商品进行信息的更改，也可以通过添加商品按钮来加入新的商品。如下图所示：

序号	商品图片	商品名称	商品价格	是否热门	编辑	删除
1		apple ipad2	4623.0	否		
2		小米MIX3手机	4623.0	否		
3		黑猫手机	4623.0	否		
4		小米MIX4手机	4623.0	否		
5		联想	232.0	否		

添加商品	
商品名称：	<input type="text"/>
销售价格：	<input type="text"/>
市场价格：	<input type="text"/>
商品图片：	<input type="button" value="选择文件"/> 未选择任何文件
所属分类：	<input type="text" value="耳机"/>
商品描述：	<input type="text"/>
<input type="button" value="确定"/> <input type="button" value="重置"/> <input type="button" value="返回"/>	

图 5- 9 添加商品

代码实现：

Java 部分：

```
// 设置图片名称，不能重复，使用 uuid
String picName = UUIDUtil.getUUID();
// 获取提交上来的文件名
String oriName = pictrueFile.getOriginalFilename();
// 通过文件名获取后缀名
String extName = oriName.substring(oriName.lastIndexOf("."));
// 上传图片
pictrueFile.transferTo(new
File("C:\\Users\\Administrator\\Desktop\\C\\src\\main\\webapp\\pic\\" + picName +
extName));
System.out.print(picName);
service.addProduct(p);
mav.addObject("msg", "添加商品成功");
mav.setViewName("admin/welcome");
return mav;
}

String picName = UUIDUtil.getUUID();
// 获取提交上来的文件名
String oriName = pictrueFile.getOriginalFilename();
// 通过文件名获取后缀名
String extName = oriName.substring(oriName.lastIndexOf("."));
```

```

// 上传图片
p.setPimage(picName + extName);
service.updateProduct(p);
}

Product p = service.queryProductByPid(pid);
}

service.deleteProduct(p);
}
}

```

5.2.3 订单管理

管理员在登录系统后可以查看所有的订单，通过搜索栏搜索订单，然后对想要更改的订单进行订单状态的更改，对已经付款的订单进行发货处理，也可以将已经完成的订单进行删除，可以通过订单详情按钮来查看订单的具体商品内容。如下图所示：



图 5-10 订单管理

代码实现：

Java 部分：

```

public class AdminOrderController {
    PageBean page = service.listOrder(curPage);
    mav.setViewName("admin/order/list");
    mav.addObject("page", page);
    return mav;
}

PageBean page = service.listOrderQita(curPage, state);
System.out.println(page);
mav.setViewName("admin/order/list2");
mav.addObject("page", page);
return mav;
}

```

5.2.4 公告管理

管理员在登录系统后可以查看所有的公告信息，可以添加新的公告，可以删除已存在的公告信息。

序号	内容	时间	操作
1	本系统晚上10点停止营业	2020-05-01	删除
2	本系统周末不营业	2020-05-03	删除
3	明天所有商品9折出售	2020-05-03	删除

图 5-11 公告管理

代码实现：

Java 部分：

```

List<notice> list= service.findnotice();
ModelAndView mav=new ModelAndView();
mav.setViewName("/admin/notice");
mav.addObject("list",list);
return mav;
}

service.deletenotice(nid);
ModelAndView mav=new ModelAndView();
mav.setViewName("/admin/welcome");
mav.addObject("msg","删除成功");
return mav;
}

```

5.2.5 用户管理

管理员在登录系统后可以查看所有的用户信息，通过搜索栏搜索用户，然后对想要更改的用户进行用户信息的更改，也可以将用户进行删除。如下图所示：

用户名	密码	姓名	电话	邮箱	性别	出生日期	修改	删除
asdasdas	111111	郑晓	13625674863	1303051288@qq.com	男	2020-05-15	修改	删除
qqqqqqqq	123456	郑晓	13625674867	1303051288@qq.com	男	2020-04-18	修改	删除

修改用户信息	
用户名：	asdasdas
密码：	111111 密码可以使用
姓名：	郑晓 姓名可以使用
邮箱：	1303051288@qq.com 邮箱可以使用
电话：	13625674863 电话可以使用
性别：	男 性别可以使用

图 5-12 用户管理

代码实现：

```
function m() {
    if($("#ptext").text()=="密码可以使用"&&$("#pemail").text()=="邮箱可以使用"
    "&&$("#pphone").text()=="电话可以使用"&&$("#pname").text()=="姓名可以使用"
    "&&($("#sex").val()=="男"||$("#sex").val()=="女")){
        $("#userAction_save_do").submit();
    }else {
        alert("请检查所有项符合要求且不为空！")
    }
}
```

Java 部分：

```
PageBean page =service.listUser( curPage);
mav.addObject("page",page);
mav.setViewName("admin/usermanage/messageuser");
return mav;
}

service.deleteuser(uid);
mav.addObject("msg", "删除成功");
mav.setViewName("admin/welcome");
return mav;
}

User user =service.finduser(uid);
mav.setViewName("admin/usermanage/edit");
```


第6章 系统测试

6.1 软件测试

为了在正式上线之前，尽量发现并解决清除软件中不易见的错误或者 bug，以此来提高软件的可用性，需要进行软件测试。此部分将对系统进行单元测试和综合测试，使用的测试方法是黑盒测试法。合适的选取测试用例力求最大程度覆盖此系统功能的方方面面，降低系统出现错误的可能性。

测试用例包括被测试的功能，因输入的数据和预期的输出结果，测试所用的数据选用少量，有效的测试数据进行测试，尽可能覆盖所有方面。

（1）数据的容错性测试：当用户输入非法数据（非法类型，不符合要求的数据）时，系统应能给出提示并进行相应的处理。当用户输入正确的数据，系统给出的结果应该与预期结果相同。

（2）接口与接口的测试，测试各个模块之间的数据传输，保证数据输入输出的一致性和正确性。

（3）压力测试：输入少量数据运行各个功能，输入大量数据运行测试。

（4）边界值分析法：确定边界情况，针对系统在测试过程中主要输入一些合法或者非法的数据，主要在边界值附件取。效率：完成预定的功能，系统运行的时间。

6.1.1 注册登录测试

The image displays two screenshots of a web application's '注册 USER REGISTER' (Registration) form, illustrating test results for different input scenarios.

Top Screenshot (Invalid Username):

- 用户名 (Username):** Input is 'qqqqqqqq'. The feedback message is '该用户名已存在' (This username already exists).
- 密码 (Password):** Input is '.....'. The feedback message is '密码需要6个数字' (Password needs 6 digits).
- Email:** Input is '1303051288@qq.com'. The feedback message is '邮箱可以使用' (Email can be used).
- Phone:** Input is '13625674863'. The feedback message is '电话可以使用' (Phone number can be used).
- 姓名 (Name):** Input is '郑斌'. The feedback message is '姓名可以使用' (Name can be used).
- 性别 (Gender):** A dropdown menu with '男' (Male) selected.
- 出生日期 (Date of Birth):** Input is '2020/05/16'.
- A '注册' (Register) button is visible at the bottom.

Bottom Screenshot (Valid Inputs):

- 用户名 (Username):** Input is 'abcdefgd'. The feedback message is '该用户名可以使用' (This username can be used).
- 密码 (Password):** Input is '.....'. The feedback message is '密码可以使用' (Password can be used).
- Email:** Input is '1303051288@qq.com'. The feedback message is '邮箱可以使用' (Email can be used).
- Phone:** Input is '13625674863'. The feedback message is '电话可以使用' (Phone number can be used).
- 姓名 (Name):** Input is '郑斌'. The feedback message is '姓名可以使用' (Name can be used).
- 性别 (Gender):** A dropdown menu with '男' (Male) selected.
- 出生日期 (Date of Birth):** Input is '2020/05/16'.
- A '注册' (Register) button is visible at the bottom.

图 6-1 注册测试

只要当所有项符合要求时，才会注册成功，当鼠标焦点失去的时候会自动判断输入是否符合。



图 6-2 登录测试

如果账号密码匹配的话，就会登陆成功并进入首页。不匹配的话会显示登录失败。

6.1.2 购物车功能测试

购物车详情

选择	图片	商品	价格(单位：¥)	数量	小计(单位：¥)	操作
<input type="checkbox"/> 选中		惠普笔记本	7777.0	<input type="text" value="1"/>	7777	删除
<input type="checkbox"/> 选中		伊利安慕希	232.0	<input type="text" value="1"/>	232	删除

☐ 全选商品金额(单位：¥): [清空购物车](#) [提交订单](#)

购物车详情

选择	图片	商品	价格(单位：¥)	数量	小计(单位：¥)	操作
<input checked="" type="checkbox"/> 选中		惠普笔记本	7777.0	<input type="text" value="4"/>	31108	删除
<input checked="" type="checkbox"/> 选中		伊利安慕希	232.0	<input type="text" value="5"/>	1160	删除

☒ 全选商品金额(单位：¥): [清空购物车](#) [提交订单](#)

购物车详情

选择	图片	商品	价格(单位：¥)	数量	小计(单位：¥)	操作
<input checked="" type="checkbox"/> 选中		惠普笔记本	7777.0	<input type="text" value="4"/>	31108	删除
<input type="checkbox"/> 选中		伊利安慕希	232.0	<input type="text" value="5"/>	1160	删除

☐ 全选商品金额(单位：¥): [清空购物车](#) [提交订单](#)

购物车详情

选择	图片	商品	价格(单位：¥)	数量	小计(单位：¥)	操作
<input type="checkbox"/> 选中		伊利安慕希	232.0	<input type="text" value="5"/>	1160	删除

☐ 全选商品金额(单位：¥): [清空购物车](#) [提交订单](#)

图 6-3 购物车测试

刚加入购物车的商品是没有被选中的，所以购物车的总金额为 0，只会计算被选中商品的价格，可实现单选，全选，多选，删除商品等功能。



图 6-6 商品信息测试

在商品信息页面可以看到顾客的评价以及店家的回复等等。

6.1.5 收货地址测试

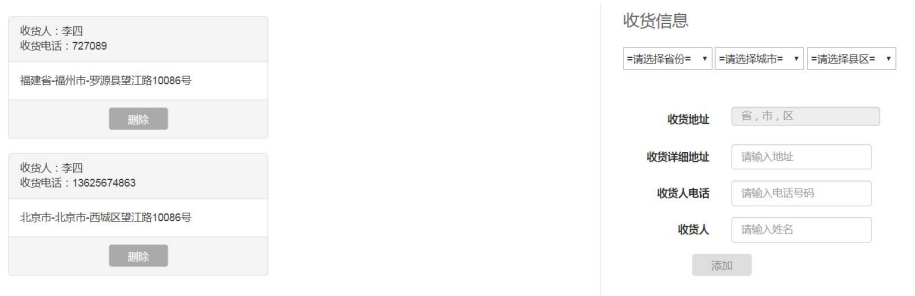


图 6-7 收货信息测试一

用户在登陆之后可以点击收货信息进入此页面，完成对收货信息的添加。

订单详情

订单编号:7b8cc5e9b4d047dd82051fd3c5a96c71

图片	商品	价格	数量	小计
	nike运动鞋 AQ7497-001白色41码	¥555.0	1	¥555.0

商品金额: ¥555.0元

收 货 人: 李四 ▼

收货地址: 福建省-福州市-罗源县望江路10086号 ▼

收货人电话: 727089 ▼

[新增收货信息](#)

图 6-8 收货信息测试二

在确认订单的页面可以选择地址也可以新增收货信息。

6.1.6 商品分类管理测试

分类列表				添加
序号	分类名称	编辑	删除	
1	耳机	✎	✖	
2	食品	✎	✖	
3	平板	✎	✖	
4	电器	✎	✖	
5	运动器材	✎	✖	
6	手机	✎	✖	
7	笔记本电脑	✎	✖	
8	服饰	✎	✖	

添加分类	
分类名称:	<input type="text"/>
确定	重置 返回

图 6-9 分类测试

管理员登录之后，可以添加分类，修改已经存在的分类。经测试，该功能无异常，可以使用。

6.1.7 商品的添加，修改测试

商品列表						
请输入商品名称						查找商品
序号	商品图片	商品名称	商品价格	商品数量	编辑	删除
1		联想R7000 R5-4800H 16G 512G SSD GTX1650 100%sRGB(幻影黑)	5300.0	111		
2		华为MatePad 10.4英寸麒麟810 4GB+64GB(夜阑灰)	1111.0	111		
3		Apple iPad mini 5(白色) 2019年新款平板电脑 7.9英寸64g内存	2333.0	110		
4		惠普(HP)暗影精灵6 Air i7-10750H 16G 1TSSD RTX2060 6G独显	8500.0	1108		
5		nike运动鞋 AQ7497-001白色41码	555.0	536		

1 2 3 4 5 6 7 8 »

添加商品			
商品名称 (必填) :			
市场价格 (必填) :		商城价格 (必填) :	
商品图片 (必填) :	选择文件 未选择任何文件	数量 (必填) :	
所属分类 (必填) :	耳机 ▼		
商品描述 (必填) :			
<div>确定 重置 返回</div>			

图 6-10 添加商品测试

管理登录后台后，可以添加商品，只有当所有信息都被填写之后，才能添加商品，也可以对已存在的商品信息进行修改，或者删除商品，以上功能都没有问题，可正常使用。

6.1.8 订单管理测试

网上商城管理系统									
2020年5月25日 星期一									
首页									
分类管理									
商品管理									
订单管理									
未付款订单									
已发货订单									
已付款订单									
已完成订单									
用户及公告管理									
评论回复管理									

订单列表									
序号	订单编号	订单金额	下单时间	订单地址	收货人电话	收货人	订单状态	订单详情	操作
1	06770eeef9f946e5b677bc4d13219d65	26115.0	2020-05-13	安徽合肥	13625674863	郑斌	订单完成	订单详情	删除
2	181edae51e284fe58ac53eb09daebd17	89870.0	2020-05-13	安徽合肥	13625674863	郑斌	订单完成	订单详情	删除
3	2dec123de5a9405c96327628a90c6b99	7219.0	2020-05-13	安徽合肥	13625674863	李四	订单完成	订单详情	删除
4	3d9256c2efba4ebf8981aee6d5648befe	30330.0	2020-05-13	安徽合肥	13625674863	郑斌	订单完成	订单详情	删除
5	5f1683b1704d4458be2a49f169a8af00	8108.0	2020-05-25	福建省 福州市 罗源县 城江路10036号	727089	李四	订单完成	订单详情	删除

1 2 3 »

图 6-11 订单管理测试

订单有 4 种状态，管理员可根据订单处于状态的不同进行订单查询，并对相关订单的状态进行操作，也可以查看订单的内容，对于不需要的订单也可删除。经过测试，订单状态修改流程可以正常使用。

6.1.9 用户管理测试

2020年5月25日 星期一							
首页	用户列表						
分类管理					请输入用户名	搜索用户	
商品管理	用户名	密码	姓名	电话	邮箱	性别	修改
订单管理	qqqqqqqq	123123	郑斌	13625674867	1303288@qq.com	男	删除
用户及公告管理	uuuuuuuu	123456	郑斌2	13625674861	1477288@qq.com	男	删除
公告管理	qwesadzz	123456	郑斌2	13625674862	4741288@qq.com	男	删除
用户管理	asddfasd	111111	郑斌21212	13625674863	13477451288@qq.com	男	删除
评论回复管理	asdasdas	111111	郑斌	13625674864	58851288@qq.com	男	删除

图 6-12 用户管理测试

管理员登录管理系统之后，可以搜索，查看用户的信息资料，并且可以修改除用户名之外的信息，也可以删除用户账号，经测试，此功能无问题。

6.1.10 商品评论回复管理测试

2020年5月25日 星期一						
首页	商品评论列表					
分类管理	序号	评论内容	评论时间	评论人	回复	操作
商品管理	1	还不错	2020-5-20 13:11:03	qqqqqqqq		确定
订单管理	2	非常好喝啊	2020-5-16 20:05:20	qqqqqqqq		确定
用户及公告管理	3	非常大	2020-5-16 16:09:04	qqqqqqqq		确定
评论回复管理	4	非常大	2020-5-16 7:36:29	qqqqqqqq		确定
未回复评论	5	非常好喝啊	2020-5-16 7:36:29	qqqqqqqq		确定

2020年5月25日 星期一						
首页	商品评论列表					
分类管理	序号	评论内容	评论时间	评论人	回复	操作
商品管理	1	用得非常好	2020-5-24 13:53:18	qqqqqqqq	商家会再接再厉	
订单管理	2	用得非常好	2020-5-24 13:53:18	qqqqqqqq	商家会再接再厉	
用户及公告管理	3	质量貌似不大好	2020-5-22 10:52:52	qqqqqqqq	商家会再接再厉	
评论回复管理	4	非常好	2020-5-22 10:51:00	qqqqqqqq	商家会再接再厉	
	5	不是太好	2020-5-22 10:51:00	qqqqqqqq	商家会再接再厉	

图 6-13 评论回复测试

用户在确认收货之后，可以对商品进行评价，并且管理员可以对其进行回复操作，如上图。经测试，此功能可以正常运行。

6.2 测试计划执行情况

表 6-1 测试计划表

测试类型	测试内容	测试目的	测试方法
功能测试	前台用户注册登录， 后台管理员登录	确定所有相关功能正 确与否	黑盒测试法
功能测试	商品的浏览，查询， 加入购物车，以及购 物车操作，订单功能	确定功能是否正常， 数据是否正确	边界值法，等价类法
功能测试	后台管理	增删改查是否正确， 是否能正确走预期的 过程	手动测试
用户界面测试	导航，连接，颜色等	页面风格是否合理， 一致	手工测试
安全测试	密码，超时	系统是否安全	黑盒法，手动测试

6.3 测试效果

测试计划是對此网上商城系统所有的功能进行的测试，并且对相关功能的按钮做了多次测试，但是，测试计划不包含没有被发现的问题，所以当系统通过测试用例，并进行修改之后，再进行多次测试，系统无错误就认为通过测试。根据实际情况，此软件对顾客购物模块，订单模块，以及管理员管理模块都进行了安全性限制，需要验证才能进行操作，因此前后都有对应的验证功能。需要改进的方面支付功能不了解，后续应该可以加上支付宝，微信等支付功能模块。测试结果：基本完成网上购物平台系统的功能模块，基本达到了用户的需求。

表 6-2 测试结果表

测试内容	结论
界面	主界面比较友好，测试效果良好
后台	后台账号安全性有待提高，密码较为简单
注册，登录	功能基本通过测试，后续可以考虑短信验证
功能（订单，商品，购物车）	基本完成需求，无大碍

结论与展望

本课题“基于 SSM 的网上商场”，从选题到开题，再到系统实现，论文定稿，所有要求均按要求实现。具体工作总结如下：

(1) 首先介绍了本项目的背景，以及国内外学者对该类型课题的研究现状，明确了项目开发的目的是和意义。

(2) 通过充分调研，选取了合适的开发技术以及项目应用平台，为项目开发做好了前期准备工作。

(3) 接着通过网上商城系统的业务需求，分析其功能性需求，对系统整体架构做了明确设计。

(4) 在完成需求分析的基础上，进行了系统设计部分，分析系统各模块功能，以及系统所涉及数据的交互情况，对项目数据库进行设计，建立较为健全的数据库表。

(5) 在对系统进行全面分析设计后，按功能模块逐步完成系统开发，最终系统需求全部按要求完成，并且保证系统稳定运行。

主要实现的功能模块有两大类：首先从顾客的角度看，此系统完成了用户账号注册登录功能，商品浏览功能，加入商品到购物车的功能，查看订单的功能，添加收货地址，评论商品的功能。以上功能能够满足用户网上购物的必要流程，保障系统正常运作。从管理员的角度看，此系统实现了添加商品，查看订单，处理订单状态，添加系统公告，管理用户账号，回复商品评价等功能，可以保障系统管理员对此系统的正常维护。

通过对本课题的研究，我对 java web 开发有了更深刻的认知，激发了我对编程的兴趣，编码能力也因此得到了很大的提高。在后台开发中，加深理解了 SSM 框架的原理，并且我还学习了 MySQL 数据库方面的相关知识，能实现对单表，多表进行增删改查。

虽然系统总体的功能都实现了，但是我还是感觉到了一些系统功能设计的不合理，但是由于自己个人技术水平的不够、经验的不足和时间的仓促，让我无法对系统进行大规模的修改和完善，因此我还是非常遗憾的，为此为列下来以下几点还可进行改进的地方：

(1) 在订单模块中，没有实现具体的付款功能，应当调用支付宝或者微信接口来实现付款。

(2) 注册登录时应当使用手机验证码来注册登录更加安全。

(3) 可以根据用户习惯来推荐商品，每个用户推荐的商品应当不同。

(4) 一些系统页面的美化不足，论文一些内容的表述存在不当之处。

致 谢

白驹过隙，忽然而已。转眼间丰富多彩的校园生活已成过往。值此本科毕业论文完结之际，谨对多年来予以我帮助、激励的良师诤友致以最恳挚的谢忱。

首先我要感谢邹老师在系统设计开发以及论文编写的各个阶段予以的中肯见解，从最初的选题、文献搜寻到系统的需求分析、实现最后到论文编写、纠错等各阶段都给予我莫大的帮助。在老师的指点中看到了她一丝不苟、专注的学术作风。让我学会了对任何真实的事物都应保持这种严谨的态度，这也是我们前进的目标更是成功的关键。其次，我还要小组同学在系统开发、调试阶段对我的指点，帮助我解决诸多了 bug 才能顺利运行系统。

感谢所有传授我知识的老师，他们含辛茹苦的教导，让我拥有了足够的知识，让我对系统的实现和论文编写充满了动力和信心。同时也感谢和我同组的同学，因为大家的和睦相处、互相帮助，让我们结下了深深的友谊，也让我们彼此都拥有一个更好的开发环境和论文编写环境。在完成毕业设计的过程中，我深深的感觉到，作为一个合格的大学生，我们应该不仅仅只会程序编写技术，更重要的是对系统整体结构的布局和设计。在今后的工作和学习当中，我会不断完善自己，让自己变得更加强大，为我的母校争光，也为自己获得更美好的未来。

作者：

日期： 年 月 日

参考文献

- [1]陈海汝,何青,潘轩平,张大宇,胡铁平.基于云服务器的SSM框架后台搭建与实现[J].信息系统工程,2019(11):114-115+117.
- [2]蔡呈杰,王贵鑫.基于SSM框架的学子商城系统的设计与实现[J].科学技术创新,2019(32):69-71.
- [3]侯金奎,鹿旭东,陈春雷,王磊.基于模型驱动的Web应用服务系统开发理论框架[J].小型微型计算机系统,2018,39(10):2345-2352.
- [4]鞠宏军,余春燕.基于SSM的高并发慕课网的设计与实现[J].电脑知识与技术,2020,16(02):54-56+90.
- [5]戚琦,廖建新,王纯,武家春.基于敏捷方法的轻量级J2EE架构的应用[J].计算机系统应用,2007(02):53-56.
- [6]王钱,王蓉,张利.基于iBatis的通用数据持久层的研究与设计[J].微计算机信息,2007(12):172-173+128.
- [7]王艳清,陈红.基于SSM框架的智能web系统研发设计[J].计算机工程与设计,2012,33(12):4751-4757.
- [8]徐刚,翟梦娇.基于SSM的美容资讯商务网站的设计与实现[J].商丘职业技术学院学报,2019,18(06):65-71.
- [9]叶惠仙,沈文杰.基于Bootstrap+spring boot框架的在线考试系统开发[J].网络安全技术与应用,2019(12):54-57.
- [10]张文龙,吴林辉,杨晨耀,蒋卫祥.基于SSM框架+vue的Web网盘系统的设计与实现[J].电脑知识与技术,2019,15(34):62-63+65.
- [11]张俊萍,朱小冬,侯娜,张鲁,梁欣.基于SSM的软件体系结构开发过程研究[J].计算机测量与控制,2011,19(08):2029-2032.
- [12]湛湘倩,狄文辉,孙冬.基于SSH框架与AJAX技术的JavaWeb应用开发[J].计算机工程与设计,2009,30(10):2590-2592+2596.
- [13]朱运乔.基于SpringBoot+SSM框架的Web应用系统搭建与实现[J].电脑编程技巧与维护,2019(10):23-25.
- [14]陈峰.基于SSM框架的B2C网上商城系统的设计与实现[D].湖南大学,2018.
- [15]唐文静,张志勤,王庆军.基于SSH2的辅助选择的网上商城的设计与实现[J].计算机与数字工程,2019,47(06):1550-1554.
- [16]Abdellatif, T. and F. Boyer. A node allocation system for deploying JavaEE systems on Grids. 2009. Hammemet, Tunisia.
- [17]Li, Z. and Z. Weixi. Design of tourism e-business system based on JavaEE multi-pattern. 2012. Sanya, China.
- [18]Andrea Capiluppi, Nemitari Ajienka, Steve Counsell. The Effect of Multiple Developers on Structural Attributes: A Study Based on Java Software[J]. Elsevier Inc., 2020.

附 录

附录 A 外文文献及其译文

Core Java™ Volume II—Advanced Features

When Java technology first appeared on the scene, the excitement was not about a well-crafted programming language but about the possibility of safely executing applets that are delivered over the Internet (see Volume I, Chapter 10 for more information about applets). Obviously, delivering executable applets is practical only when the recipients are sure that the code can't wreak havoc on their machines. For this reason, security was and is a major concern of both the designers and the users of Java technology. This means that unlike other languages and systems, where security was implemented as an afterthought or a reaction to break-ins, security mechanisms are an integral part of Java technology.

Three mechanisms help ensure safety:

- Language design features (bounds checking on arrays, no unchecked type conversions, no pointer arithmetic, and so on).
- An access control mechanism that controls what the code can do (such as file access, network access, and so on).
- Code signing, whereby code authors can use standard cryptographic algorithms to authenticate Java code. Then, the users of the code can determine exactly who created the code and whether the code has been altered after it was signed.

Below, you'll see the cryptographic algorithms supplied in the `java.security` package, which allow for code signing and user authentication.

As we said earlier, applets were what started the craze over the Java platform. In practice, people discovered that although they could write animated applets like the famous "nervous text" applet, applets could not do a whole lot of useful stuff in the JDK 1.0 security model. For example, because applets under JDK 1.0 were so closely supervised, they couldn't do much good on a corporate intranet, even though relatively little risk attaches to executing an applet from your company's secure intranet. It quickly became clear to Sun that for applets to become truly useful, it was important for users to be able to assign different levels of security, depending on where the applet originated. If an applet comes from a trusted supplier and it has not been tampered with, the user of that applet can then decide whether to give the applet more privileges.

To give more trust to an applet, we need to know two things:

- Where did the applet come from?
- Was the code corrupted in transit?

In the past 50 years, mathematicians and computer scientists have developed sophisticated algorithms for ensuring the integrity of data and for electronic signatures. The `java.security` package contains implementations of many of these algorithms. Fortunately, you don't need to understand the underlying mathematics to use the algorithms in the `java.security` package. In the next sections, we show you how message digests can detect changes in data files and how digital signatures can prove the identity of the signer.

A message digest is a digital fingerprint of a block of data. For example, the so-called

SHA1 (secure hash algorithm #1) condenses any data block, no matter how long, into a sequence of 160 bits (20 bytes). As with real fingerprints, one hopes that no two messages have the same SHA1 fingerprint. Of course, that cannot be true—there are only 2^{160} SHA1 fingerprints, so there must be some messages with the same fingerprint. But 2^{160} is so large that the probability of duplication occurring is negligible. How negligible? According to James Walsh in *True Odds: How Risks Affect Your Everyday Life* (Merritt Publishing 1996), the chance that you will die from being struck by lightning is about one in 30,000. Now, think of nine other people, for example, your nine least favorite managers or professors. The chance that you and all of them will die from lightning strikes is higher than that of a forged message having the same SHA1 fingerprint as the original. (Of course, more than ten people, none of whom you are likely to know, will die from lightning strikes. However, we are talking about the far slimmer chance that your particular choice of people will be wiped out.)

A message digest has two essential properties:

- If one bit or several bits of the data are changed, then the message digest also changes.
- A forger who is in possession of a given message cannot construct a fake message that has the same message digest as the original.

The second property is again a matter of probabilities, of course. Consider the following message by the billionaire father: "Upon my death, my property shall be divided equally among my children; however, my son George shall receive nothing."

That message has an SHA1 fingerprint of

2D 8B 35 F3 BF 49 CD B1 94 04 E0 66 21 2B 5E 57 70 49 E1 7E

The distrustful father has deposited the message with one attorney and the fingerprint with another. Now, suppose George can bribe the lawyer holding the message. He wants to change the message so that Bill gets nothing. Of course, that changes the fingerprint to a completely different bit pattern:

2A 33 0B 4B B3 FE CC 1C 9D 5C 01 A7 09 51 0B 49 AC 8F 98 92

Can George find some other wording that matches the fingerprint? If he had been the proud owner of a billion computers from the time the Earth was formed, each computing a million messages a second, he would not yet have found a message he could substitute.

A number of algorithms have been designed to compute these message digests. The two best-known are SHA1, the secure hash algorithm developed by the National Institute of Standards and Technology, and MD5, an algorithm invented by Ronald Rivest of MIT. Both algorithms scramble the bits of a message in ingenious ways. For details about these algorithms, see, for example, *Cryptography and Network Security*, 4th ed., by William Stallings (Prentice Hall 2005). Note that recently, subtle regularities have been discovered in both algorithms. At this point, most cryptographers recommend avoiding MD5 and using SHA1 until a stronger alternative becomes available. (See <http://www.rsa.com/rsalabs/node.asp?id=2834> for more information.)

The Java programming language implements both SHA1 and MD5. The `MessageDigest` class is a factory for creating objects that encapsulate the fingerprinting algorithms. It has a static method, called `getInstance`, that returns an object of a class that extends the `MessageDigest` class. This means the `MessageDigest` class serves double duty:

- As a factory class

- As the superclass for all message digest algorithms

For example, here is how you obtain an object that can compute SHA fingerprints:

```
MessageDigest alg = MessageDigest.getInstance("SHA-1");
```

(To get an object that can compute MD5, use the string "MD5" as the argument to getInstance.)

After you have obtained a MessageDigest object, you feed it all the bytes in the message by repeatedly calling the update method. For example, the following code passes all bytes in a file to the alg object just created to do the fingerprinting:

```
InputStream in = . . .
```

```
int ch;
```

```
while ((ch = in.read()) != -1)
```

```
alg.update((byte) ch);
```

Alternatively, if you have the bytes in an array, you can update the entire array at once:

```
byte[] bytes = . . .;
```

```
alg.update(bytes);
```

When you are done, call the digest method. This method pads the input—as required by the fingerprinting algorithm—does the computation, and returns the digest as an array of bytes.

```
byte[] hash = alg.digest();
```

The program in Listing 9-15 computes a message digest, using either SHA or MD5. You can load the data to be digested from a file, or you can type a message in the text area.

Message Signing

In the last section, you saw how to compute a message digest, a fingerprint for the original message. If the message is altered, then the fingerprint of the altered message will not match the fingerprint of the original. If the message and its fingerprint are delivered separately, then the recipient can check whether the message has been tampered with. However, if both the message and the fingerprint were intercepted, it is an easy matter to modify the message and then recompute the fingerprint. After all, the message digest algorithms are publicly known, and they don't require secret keys. In that case, the recipient of the forged message and the recomputed fingerprint would never know that the message has been altered. Digital signatures solve this problem.

To help you understand how digital signatures work, we explain a few concepts from the field called public key cryptography. Public key cryptography is based on the notion of a public key and private key. The idea is that you tell everyone in the world your public key. However, only you hold the private key, and it is important that you safeguard it and don't release it to anyone else. The keys are matched by mathematical relationships, but the exact nature of these relationships is not important for us. (If you are interested, you can look it up in *The Handbook of Applied Cryptography* at <http://www.cacr.math.uwaterloo.ca/hac/>.)

The keys are quite long and complex. For example, here is a matching pair of public and private Digital Signature Algorithm (DSA) keys.

Public key:

Code View:

p:

```
fca682ce8e12caba26efccf7110e526db078b05edecbcd1eb4a208f3ae1617ae01f35b91a47e6
```

df63413c5e12ed0899bcd132acd50d99151bdc43ee737592e17

q: 962edddcc369cba8ebb260ee6b6a126d9346e38c5

g:678471b27a9cf44ee91a49c5147db1a9aaf244f05a434d6486931d2d14271b9e35030b71fd73da179069b32e2935630e1c2062354d0da20a6c416e50be794ca4

y:

c0b6e67b4ac098eb1a32c5f8c4c1f0e7e6fb9d832532e27d0bdab9ca2d2a8123ce5a8018b8161a760480fadd040b927281ddb22cb9bc4df596d7de4d1b977d50

Private key:

Code View:

p:

fca682ce8e12caba26efccf7110e526db078b05edecbcd1eb4a208f3ae1617ae01f35b91a47e6df63413c5e12ed0899bcd132acd50d99151bdc43ee737592e17

q: 962edddcc369cba8ebb260ee6b6a126d9346e38c5

g:

678471b27a9cf44ee91a49c5147db1a9aaf244f05a434d6486931d2d14271b9e35030b71fd73da179069b32e2935630e1c2062354d0da20a6c416e50be794ca4

x: 146c09f881656cc6c51f27ea6c3a91b85ed1d70a

It is believed to be practically impossible to compute one key from the other. That is, even though everyone knows your public key, they can't compute your private key in your lifetime, no matter how many computing resources they have available.

It might seem difficult to believe that nobody can compute the private key from the public keys, but nobody has ever found an algorithm to do this for the encryption algorithms that are in common use today. If the keys are sufficiently long, brute force—simply trying all possible keys—would require more computers than can be built from all the atoms in the solar system, crunching away for thousands of years. Of course, it is possible that someone could come up with algorithms for computing keys that are much more clever than brute force. For example, the RSA algorithm (the encryption algorithm invented by Rivest, Shamir, and Adleman) depends on the difficulty of factoring large numbers. For the last 20 years, many of the best mathematicians have tried to come up with good factoring algorithms, but so far with no success. For that reason, most cryptographers believe that keys with a "modulus" of 2,000 bits or more are currently completely safe from any attack. DSA is believed to be similarly secure.

Figure 9-12 illustrates how the process works in practice.

Suppose Alice wants to send Bob a message, and Bob wants to know this message came from Alice and not an impostor. Alice writes the message and then signs the message digest with her private key. Bob gets a copy of her public key. Bob then applies the public key to verify the signature. If the verification passes, then Bob can be assured of two facts:

- The original message has not been altered.
- The message was signed by Alice, the holder of the private key that matches the public key that Bob used for verification.

You can see why security for private keys is all-important. If someone steals Alice's private key or if a government can require her to turn it over, then she is in trouble. The thief or a government agent can impersonate her by sending messages, money transfer instructions, and so on, that others will believe came from Alice.

The X.509 Certificate Format

To take advantage of public key cryptography, the public keys must be distributed. One of the most common distribution formats is called X.509. Certificates in the X.509 format are widely used by VeriSign, Microsoft, Netscape, and many other companies, for signing e-mail messages, authenticating program code, and certifying many other kinds of data. The X.509 standard is part of the X.500 series of recommendations for a directory service by the international telephone standards body, the CCITT.

The precise structure of X.509 certificates is described in a formal notation, called "abstract syntax notation #1" or ASN.1. Figure 9-13 shows the ASN.1 definition of version 3 of the X.509 format. The exact syntax is not important for us, but, as you can see, ASN.1 gives a precise definition of the structure of a certificate file. The basic encoding rules, or BER, and a variation, called distinguished encoding rules (DER) describe precisely how to save this structure in a binary file. That is, BER and DER describe how to encode integers, character strings, bit strings, and constructs such as SEQUENCE, CHOICE, and OPTIONAL.

中文译文：

Java 核心技术 卷 II 高级特性

当 Java 技术刚刚问世时，令人激动的并不是因为它是一个设计完美的编程语言，而是因为它能够安全地运行通过因特网传播的各种 applet。很显然，只有当用户确信 applet 的代码不会破坏他的计算机时，用户才会接受在网上传播的可执行的 applet。正因为如此，无论过去还是现在，安全都是设计人员和 Java 技术使用者所关心的一个重大问题。这就意味着，Java 技术与其他语言和系统有所不同，在那些语言和系统中安全是事后才想到要去实现的，或者仅仅是对破坏的一种应对措施，而对 Java 技术来说，安全机制是一个不可分割的组成部分。

Java 技术提供了以下三种确保安全的机制：

- (1)语言设计特性（对数组的边界进行检查，无不检查类型的转换，无指针算法等）。
- (2)访问控制机制，用于控制代码能够执行的功能（比如文件访问，网络访问等）。
- (3) 代码签名，利用该特性，代码的作者就能够用标准的加密算法来表明 Java 代码的身份。这样，该代码的使用者就能够准确地知道谁创建了该代码，以及代码被标识后是否被修改过。

下面，我们要介绍 java.security 包提供的加密算法，用来进行代码的标识和用户身份认证。

正如我们前面所说，applet 是在 Java 平台上开始流行起来的。实际上，人们发现尽管他们可以编写像著名的“nervous text”那样栩栩如生的 applet，但是在 JDK1.0 安全模式下无法发挥其一整套非常有用的作用。例如，由于 JDK1.0 下的 applet 要受到严密的监督，因此，即使 applet 在公司安全内部网上运行时的风险相对较小，applet 也无法在企业内部网上发挥很大的作用。Sun 公司很快就认识到，要使 applet 真正变得非常有用，用户必须可以根据 applet 的来源为其分配不同的安全级别。如果 applet 来自值得信赖的提供商，并且没有被篡改过，那么 applet 的用户就可以决定是否给 applet 授予更多的运行特权。

如果要给予 applet 更多的信赖，你必须知道下面两件事：

- (1)applet 来自哪里？
- (2)在传输过程中代码是否被破坏？

在过去的 50 年里，数学家和技术机科学家已经开发出各种各样成熟的算法，用于确保数据和电子签名的完整性，在 java.security 包中包含了许多这些算法的实现。在下面几节，我们将要介绍消息摘要是如何检测数据文件中的变化的，以及数字签名是如何证明签名者的身份的。

消息摘要是数据块的数字指纹。例如，所谓的 SHA1（安全散列算法#1）可将任何数据块，无论其数据有多长，都压缩为 160 位（20 字节）的序列。与真实的指纹一样，人们希望任何两条消息都不会有相同的 SHA1 指纹。当然这是不可能的——因为只存在 2^{160} 个 SHA1 指纹，所有肯定会有某些消息具有相同的指纹。因为 2^{160} 是一个很大的数字，所以存在重复指纹的可能性微乎其微，那么这种重复的可能性到底小到什么程度呢？根据 James Walsh 在他的《True Odds:How Risks Affect Your Everyday Life》,Merriitt Publishing 出版社 1996 年出版，一书中所阐述的，你和他们所有的人都死于雷击的概率，比伪造的消息与原来消息具有相同的 SHA1 指纹的概率还要高。（当然，可能有你不认识的其他 10 个以上的人会死于雷击，但这里我们讨论的是你选择的特定的人的死亡概率）。

消息摘要具有两个基本属性：

(1) 如果数据的 1 位或者几位改变了，那么消息摘要也将改变。

(2) 拥有给定消息的伪造者不能创建与原消息具有相同摘要的假消息。

当然，第二个属性又是一个概率问题。让我们来看看下面这位亿万富翁下的遗嘱：“我死了之后，我的财产将由我的孩子平分，但是，我的儿子 George 应该拿不到一个子。”

这份遗嘱的 SHA1 指纹为：

2D 8B 35 F3 BF 49 CD B1 94 04 E0 66 21 2B 5E 57 70 49 E1 7E

这位有疑心病的父亲将这份遗嘱交给一位律师保存，而将指纹交给另一位律师保存。现在，假设 George 能够贿赂那位保存遗嘱的律师，他想修改这份遗嘱，使得 Bill 一无所得。当然，这需要将原指纹改为下面这样完全不同的位模式：

2A 33 0B 4B B3 FE CC 1C 9D 5C 01 A7 09 51 0B 49 AC 8F 98 92

那么 George 能够找到与该指纹相匹配的其他文字吗？如果从地球形成之时，他就很自豪地拥有 10 亿台计算机，每台计算机每秒钟处理一百万条信息，他依然无法找到一个能够替换的遗嘱。

人们已经设计出大量的算法，用于计算这些消息摘要，其中最著名的两种算法是 SHA1 和 MD5。SHA1 是由美国国家标准和技术学会开发的加密散列算法，MD5 是由麻省理工学院的 Ronald Rivest 发明的算法。这两种算法都使用了独特巧妙的方法对消息中的各个位进行扰乱。如果要了解这些方法的详细信息，请参阅 William Stallings 撰写的《Cryptography and Network Security》一书，该书由 Prentice Hall 出版社于 2005 年出版。值得注意的是，最近人们在这两种算法中发现了某些微妙的规律性，因此许多密码人员建议最好避免使用 MD5，而应该使用 SHA1 算法，直到有更强的加密算法出现。（查看 <http://www.rsa.com/rsalabs/node.asp?id=2834> 以了解更多的信息）。

Java 编程语言已经实现了 SHA1 和 MD5。MessageDigest 类是用于创建封装了指纹算法的对象的“工厂”，它的静态方法 getInstance 返回继承了 MessageDigest 类的某个类的对象。这意味着 MessageDigest 类能够承担下面的双重职责：

(1) 作为一个工厂类。

(2) 作为所有消息摘要算法的超类。

例如，下面是如何获取一个能够计算 SHA 指纹的对象的方法：

```
MessageDigest alg = MessageDigest.getInstance("SHA-1");
```

（如果要获取计算 MD5 的对象，请使用字符串“MD5”作为 getInstance 的参数。）

当你已经获取 MessageDigest 对象之后，通过反复调用 update 方法，将信息中的所有字节提供给该对象。例如，下面的代码将文件中的所有字节传给上面建立的 alg 对象，以执行指纹算法：

```
InputStream in=...
int ch;
while((ch=in.read())!=-1)
    alg.update((byte) ch);
```

另外，如果这些字节存放在一个数组中，那就可以一次完成整个数组的更新：

```
byte[] bytes = . . . ;
alg.update(bytes);
```

当完成上述操作后，调用 digest 方法。该方法填充输入信息——指纹算法需要的——并且进行相应的计算，然后以字节数组的形式返回消息摘要。

```
byte[] hash=alg.digest();
```

程序清单 9-15 中的程序计算了一个消息摘要，既可以用 SHA，也可以使用 MD5 来计算。可以从文件加载需要计算摘要的数据，也可以直接将信息输入文本区域。图 9-11 显示了该应用程序的画面。

消息签名

在上一节中，我们介绍了如何计算原始消息的消息摘要和指纹的方法。如果消息改变了，那么改变后的消息的指纹与原消息的指纹将不匹配。如果消息和它的指纹是分开传送的，那么接收者就可以检查消息是否被篡改过。但是，如果消息和指纹同时被截获了，对消息进行修改，再重新计算指纹，这是一件很容易的事情。毕竟，消息摘要算法是公开的，不需要使用任何密钥。在这种情况下，假消息和新指纹的接收者永远不会知道消息已经被篡改。数字签名解决了这个问题。

为了了解数字签名的工作原理，我们需要解释关于公共密钥加密技术领域中的几个概念。公共密钥加密技术是基于公共密钥和私有密钥这两个基本概念的。它的设计思想是你可以将公共密钥告诉世界上的任何人，但是，只有自己才拥有私有密钥，重要的是你要保护你的私有密钥，不将它泄漏给其他任何人。这些密钥之间存在一定的数学关系，但是这种关系的具体性质对于实际的编程来说并不重要(如果你有兴趣，可以参阅 <http://www.cacr.math.uwaterloo.ca/hac/> 站点上的《The Handbook of Applied Cryptography》一书)。

密钥非常长，而且很复杂。例如，下面是一对匹配的数字签名算法(DSA)公共密钥和私有密钥。

公共密钥：

p:

fca682ce8e12caba26efccf7110e526db078b05e6ecbcdle4a208f3ae1617ae01f35b91a47e6df63413c5e12ed0899bcd132acd50d99151bdc43ee737592e17

q: 962edddcc369cba8ebb260ee6b6a126d9346e38c5

g: 678471b27a9cf44ee91a49c5147db1a9aaf244f05a434d6486931d2d14271b9e35030b71fd73da179069b32e2935630elc2062354d0da20a6c416e50be794ca4

y:

c0b6e67b4ac098ebla32c5f8c4clfee7e6fb9d832532e27d0bdab9ca2d2a8123ce5a8018b816la6048efadd040b927281ddb22cb9bc4df596d7de4dlb977dS0

私有密钥：

p:

fca682ce8e12caba26efccf7110e526db078b05edecbcdle4a208f3ae1617ae01f35b91a47e6df63413c5e12ed0899bcd132acd50d99151bdc43ee737592e17

q: 962edddcc369cba8ebb260ee6b6a126d9346e38c5

g:

678471b27a9cf44ee91a49c5147db1a9aaf244f05a434d6486931d2d14271b9e35030b71fd73da179069b32e2935630elc2062354d0da20a6c416e50be794ca4

x: 146c09f881656cc6c51f27ea6c3a91b85edld70a

在现实中，几乎不可能用一个密钥去推算出另一个密钥。也就是说，即使每个人都都知道你的公共密钥，不管他们拥有多少计算资源，他们一辈子也无法计算出你的私有密钥。

任何人都无法根据公共密钥来推算私有密钥，这似乎让人难以置信。但是时至今日，还没有人能够找到一种算法，来为现在常用的加密算法进行这种推算。如果密钥足

够长，那么要是使用穷举法——也就是直接试验所有可能的密钥——所需要的计算机将比用太阳系中的所有原子来制造的计算机还要多，而且还得花费数千年的时间。当然，可能会有人提出比穷举更灵活的计算密钥的算法。例如，RSA 算法（该加密算法由 Rivest, Shamir 和 Adleman 发明）就利用了对数值巨大的数字进行因子分解的困难性。在最近 20 年里，许多优秀的数学家都在尝试提出好的因子分解算法，但是迄今为止都没有成功。据此，大多数密码学者认为，拥有 2000 位或者更多位“模数”的密钥目前是完全安全的，可以抵御任何攻击。DSA 被认为具有类似的安全性。

图 9-12 展示了这项工作的处理过程。

假设 Alice 想要给 Bob 发送一个消息，Bob 想知道该消息是否来自 Alice，而不是冒名顶替者。Alice 写好了消息，并且用她的私有密钥对该消息摘要签名。Bob 得到了她的公共密钥的拷贝，然后 Bob 用公共密钥对该签名进行校验。如果通过了校验，则 Bob 可以确认以下两个事实：

- (1)原始消息没有被篡改过。
- (2)该消息是由 Alice 签名的，她是私有密钥的持有者，该私有密钥就是 Bob 与她用于校验的公共密钥相匹配的密钥。

你可以看到私有密钥的安全性为什么是最重要的。如果某个人偷了 Alice 的私有密钥，或者政府要求她交出私有密钥，那么她就麻烦了。小偷或者政府代表就可以假扮她的身份来发送消息和资金转账指令等等，而其他人则会相信这些消息确实来自于 Alice。

X.509 证书格式

为了利用公共密钥这种密码系统，必须将公共密钥分发出去。最通用的一种签名证书格式称为 X.509 格式。X.509 格式的证书被 VeriSign、微软、网景和其他许多公司广泛应用于对电子邮件消息进行签名，对程序代码进行认证，以及对许多其他类型的数据进行认证等等。X.509 标准是由国际电话标准机构，即国际电报电话咨询委员会(CCITT)提出的用于目录服务的 X.500 系列建议的组成部分。

X.509 证书的具体结构是用一种形式化表示来描述的，称为“抽象语法表示法 #1”(abstract syntax notation)即 ASN.1。图 9-13 显示了第 3 版 X.509 格式的 ASN.1 定义。虽然具体的语法对我们并不重要，但是你可以看到，ASN.1 为证书文件的结构给出了精确的定义。“基本编码规则”(basic encoding rules)，即 BER，精确地描述了如何将该结构保存为二进制文件。也就是说，BER 描述了如何对整数、字符串、位串以及诸如 SEQUENCE、CHOICE 和 OPTIONAL 的结构进行编码的方法。

附录 B 主要参考文献的题录及摘要或参考文献原文

【1】【篇名】基于云服务器的 SSM 框架后台搭建与实现

【作者】陈海汝,何青,潘轩平,张大宇,胡铁平.

【刊名】信息系统工程

【中文关键词】云服务器; SSM 框架; 微信小程序; MSQL 数据库;

【中文摘要】摘要：云服务器已经成为了中小型企业和个人最适用的服务器平台,便利和维护费用低等优点使其成为了小型项目首选服务器选择。SSM 框架是商业中最常用的后台框架,框架具有维护便利、拓展性高和移植性强等优点,将 SSM 后台框架部署在云服务器可以在很短的时间搭建后台并上线项目。

【2】【篇名】基于 SSM 框架的学子商城系统的设计与实现

【作者】蔡呈杰,王贵鑫

【刊名】科学技术创新

【中文关键词】学子商城; 系统; 软件;

【中文摘要】摘要：从 20 世纪 90 年代开始,随着互联网,大数据的来临,人类进入快节奏的信息时代,即互联网时代,其中网上浏览商品,购买商品已经是人们都会做的事情,已经成为人们生活不可或缺的一部份,因此开发出一款用户体验好,用户数据的安全性高,可维护性强的商城平台是多么重要。虽然目前已经存在很多商城平台,但其中还是有很多不足和需要改进的地方,这次设计是基于 SSM 框架开发的一款学子商城项目,该项目继承了已经存在项目的一部分优点,并在此基础上有了创新和改进,比如使用分层架构的思想,用户登录需要输入验证码等,不但提高了安全性,而且后端代码易于管理,方便后期功能的添加与修改,满足更多用户的需求。

【3】【篇名】基于模型驱动的 Web 应用服务系统开发理论框架

【作者】侯金奎,鹿旭东,陈春雷,王磊.

【刊名】小型微型计算机系统

【中文关键词】Web 应用服务系统;模型驱动开发;模型转换;模型精化;

【中文摘要】摘要：从软件设计实践出发,针对基于 Web 应用服务和设计模式的软件设计方法,提出了一种新的模型驱动开发理论框架.该框架以类型范畴理论为基础,以服务构件标识为范畴对象,以服务构件之间的交互和依赖关系为范畴态射,从而用范畴图表给出了软件体系结构模型的框架描述.设计模式看成是构件规范的模板,模板的实例化和规范的集成过程就是软件设计和实现的过程,从而建立起了模型转换和精化中特性描述和一致性研究的统一框架.应用研究表明,该框架符合模型驱动软件设计的本质要求,为模型转换和模型驱动的 Web 应用服务开发提供了新的认知、分析和研究的指导架构.

【4】【篇名】基于 SSM 的高并发慕课网的设计与实现

【作者】鞠宏军,余春燕.

【刊名】电脑知识与技术

【中文关键词】负载均衡;慕课;SSM;Nginx;Redis;

【中文摘要】摘要：为了帮助 IT 学子能够更加便利的获取学习资源,该系统采用 B/S 架构的方式,将大量学习资源通过互联网整合在一起,提供给用户学习使用。系统主要实现视频学习、发布讨论、撰写博客、站内搜索、程序在线测评等功能,用户也可以通过评论的方式实现问题解答、交流互动。

【5】【篇名】基于敏捷方法的轻量级 J2EE 架构的应用

【作者】戚琦,廖建新,王纯,武家春

【刊名】计算机系统应用

【中文关键词】敏捷建模;J2EE;Struts;Spring;Hibernate;

【中文摘要】摘要：本文提出 Sfruls+Spring+Hibernate 三个开源框架的松耦合层次结构,加之 Tomcat 开源服务器组成一个轻量级的 J2EE 架构。文章结合敏捷软件开发方法说明了轻量级 J2EE 架构设计实现的过程,并针对目标系统彩信短信管理平台设计了 Struts+Spring+Hibernate 的系统概念模型。本文希望通过目标系统的开发探索敏捷开发的思想在轻量级 J2EE 系统架构中的应用。

【6】【篇名】基于 iBatis 的通用数据持久层的研究与设计

【作者】王钱,王蓉,张利

【刊名】微计算机信息

【中文关键词】持久层对象; iBatis; SQL;

【中文摘要】摘要：目前持久层开发仍然是持久层对象和数据库表格一一对应,严重影响了开发和维护效率,针对目前传统的持久层开发思想,本文提出了一种基于 iBatis 框架的通用数据持久层设计方案,避免了重复劳动。

【7】【篇名】基于 SSM 框架的智能 web 系统研发设计

【作者】王艳清,陈红

【刊名】计算机工程与设计

【中文关键词】智能; web 系统; 布局; 效率; SSM 框架;

【中文摘要】摘要：为了解决传统 CMS 系统不能对 web 布局进行修改的局限性,降低维护 web 系统成本及风险,提高管理效率,提出了集建站和维护系统于一体智能 web 系统设计。设计中使用了 SSM 框架,很大程度上提高了开发效率及系统的可靠性和易用性;Ajax、jQuery、jqGrid 等技术的采用实现了拖动页面布局修改 web 布局的效果。此外还对传统的菜单管理和内容管理等功能进行了改进,从而提高了 web 系统维护效率,降低维护风险,更方便管理人员。

【8】【篇名】基于 SSM 的美容资讯商务网站的设计与实现

【作者】徐刚,翟梦娇

【刊名】商丘职业技术学院学报

【中文关键词】SSM; 购物; 美妆信息;

【中文摘要】摘要：随着人们对于美好生活的向往,人们越来越注重个人外在美.虽然网上购物的网站较多,但集购物与美容资讯于一体的网站较少.因此,以 Spring+Spring MVC+My Batis 作为基本架构,设计并实现了一个网站.该网站不仅具有基本的用户购物功能,还提供了各种美妆信息供用户参考,以使用户选择合适的商品,同时后台设计简洁直接,便于管理.与一般网站相比,该网站不仅可以帮助用户选择合适的商品,还具有小巧、直接、高效、自动化、延迟较低等特点,进而可以更好地提高用户体验.

【9】【篇名】基于 Bootstrap+spring boot 框架的在线考试系统开发

【作者】叶惠仙,沈文杰

【刊名】网络安全技术与应用

【中文关键词】在线考试系统;Bootstrap;spring boot 框架;信息化;

【中文摘要】摘要：Internet 的开放分布性特点使各种考试突破了时间和空间限制,该在线考试系统设计使用 Bootstrap+spring boot 框架,实现用户管理,题库管理、成绩管理、考试自动组合出卷与评卷管理等功能。本框架能够减少底层开发工作量,系统模块之间具有高内聚、低耦合的特点,进而提升系统的可用性和可维护性。本系统能满足各科课程教学考试考核需求,简化并改善传统纸质考试模式,提高考试效率与学院信息化建设。

【10】【篇名】基于 SSM 框架+vue 的 Web 网盘系统的设计与实现

【作者】张文龙,吴林辉,杨晨耀,蒋卫祥.

【刊名】电脑知识与技术

【中文关键词】文件上传; SSM; vue,Web 网盘系统; 系统设计;

【中文摘要】摘要：针对当前 Web 网盘系统存在的问题,提出了利用 SSM+vue 的技术架构,进行技术方案设计。该文首先通过 UML 的用例图分析了 Web 网盘系统的需求;其次应用 SSM 框架设计了系统的架构;前端使用 vue 实现前后端分离,最后介绍了系统的实现。SSM+vue 框架能简化 Web 网盘系统的开发,可以有效地降低各层之间的耦合度,提高系统的可维护性。

【11】【篇名】The Effect of Multiple Developers on Structural Attributes: A Study Based on Java Software

【作者】Andrea Capiluppi; Nemitari Ajienka; Steve Counsell;

【刊名】The Journal of Systems & Software

【外文关键词】Object oriented; Metrics; Collaborative development; Open source; Software structure;

【外文摘要】Context : Long-term software projects employ different software developers who collaborate on shared artifacts. The accumulation of changes pushed by different developers leave traces on the underlying code, that have an effect on its future maintainability, and even reuse.

Objective : This study focuses on the how the changes by different developers might have an impact on the code: we investigate whether the work of multiple developers, and their experience, have a visible effect on the structural metrics of the underlying code.

Method : We consider nine object-oriented (OO) attributes and we measure them in a GitHub sample containing the top 200 ‘forked’ projects. For each of their classes, we evaluated the number of distinct developers contributing to its source code, and their experience in the project.

Results : We show that the presence of multiple developers working on the same class has a visible effect on the chosen OO metrics, and often in the opposite direction to what the guidelines for each attribute suggest. We also show how the relative experience of developers in a project plays an important role in the distribution of those metrics, and the future maintenance of the Java classes.

Conclusions : Our results show how distributed development has an effect on the structural attributes of a software system and how the experience of developers plays a fundamental role in that effect. We also discover workarounds and best practices in 4 applied case studies.

附录 C 部分源代码

主要代码:

```
package com.ahpu.ssm.controller.admin;
@Controller
@RequestMapping("/admin")
public class AdminController {
    @Autowired
    UserService service;
    @Autowired
    ProductService service1;
    @Autowired
    OrderService service2;
    @RequestMapping("/getuser")
    public ModelAndView getuser(int curPage){
        ModelAndView mav=new ModelAndView();
        PageBean page =service.listUser( curPage);

        mav.addObject("page",page);
        mav.setViewName("admin/usermanage/messageuser");
        return mav;
    }
    @RequestMapping("/finduserbyname")
    public ModelAndView finduserbyname(String uname){
        ModelAndView mav=new ModelAndView();
        User u=service.finduser(uname);

        mav.addObject("u",u);
        mav.setViewName("admin/usermanage/finduser");
        return mav;
    }
    @RequestMapping("/deleteuser")
    public ModelAndView deleteuser(String uid){
        ModelAndView mav=new ModelAndView();
        service.deleteuser(uid);
        mav.addObject("msg", "删除成功");
        mav.setViewName("admin/welcome");
        return mav;
    }
    @RequestMapping("/updateuser")
    public ModelAndView updateuser(String username){
        ModelAndView mav=new ModelAndView();
        User user =service.finduser(username);
```

```
        mav.addObject("user", user);
        mav.setViewName("admin/usermanage/edit");
        return mav;

    }

    @RequestMapping("/updatemessage")
    public ModelAndView updatemessage(User user){
        ModelAndView mav=new ModelAndView();
        service.updatemessage(user);

        mav.addObject("msg", "修改成功");
        mav.setViewName("admin/welcome");
        return mav;

    }

    @RequestMapping("/adminLogin")
    public String login(){
        return "admin/adminLogin";
    }

    @RequestMapping("doLogin")
    @ResponseBody
    public String dologin(@Param("admin") admin admin,HttpSession session){
        //    admin a=new admin();
        //    a.setAcount(ccount);
        //    a.setPwd(pwd);
        if(service.adminLogin(admin)){
            session.setAttribute("admin", admin);
            String msg="yes";
            return msg;
        }else {
            String msg="no";
            return msg;
        }

    }

}

@RequestMapping("/addProduct")
public String addProduct() {
    return "admin/product/add";
}
```

```
}

@RequestMapping("/index")
public String index() {
    return "admin/home";
}

@RequestMapping("/top")
public String top() {
    return "admin/top";
}

@RequestMapping("/bottom")
public String bottom() {
    return "admin/bottom";
}

@RequestMapping("/left")
public String left() {
    return "admin/left";
}

@RequestMapping("/welcome")
public String welcome() {
    return "admin/welcome";
}

@RequestMapping("/msg")
public ModelAndView msg() {
    int s0=service2.findcount(0);
    int s1=service2.findcount(1);
    int s2=service2.findcount(2);
    int s3=service2.findcount(3);
    int s4=service2.findcount(4);
    List list= new ArrayList();

    list.add(s0);
    list.add(s1);
    list.add(s2);
    list.add(s3);
    list.add(s4);

    List list1=new ArrayList();
    list1.add(service.selectcount(0));
    System.out.println(service.selectcount(0));
    list1.add(service.selectcount(1));

    ModelAndView mav=new ModelAndView();
```

```
mav.addObject("list",list);
mav.addObject("list1",list1);
mav.setViewName("admin/msg");
return mav;
}
@RequestMapping("/notice")
public ModelAndView notice(){
    List<notice> list= service.findnotice();
    ModelAndView mav=new ModelAndView();
    mav.setViewName("/admin/notice");
    mav.addObject("list",list);

    return mav;
}
@RequestMapping("/deletenotice")
public ModelAndView deletenotice(String nid){
    service.deletenotice(nid);
    ModelAndView mav=new ModelAndView();
    mav.setViewName("/admin/welcome");
    mav.addObject("msg","删除成功");
    return mav;
}
@RequestMapping("/updatenotice")
@ResponseBody
public String updatenotice(String notice){

    notice n=new notice();
    n.setNotice(notice);
    n.setTime(new Date(System.currentTimeMillis()));
    service.addnotice(n);

    return "success";
}

@RequestMapping("/anser")
public ModelAndView anser(int curPage ){
    ModelAndView mav=new ModelAndView();
    PageBean page=service1.findallcomments(curPage);
    mav.addObject("page",page);
    mav.setViewName("admin/product/comments");
```

```
        return mav;
    }

    @RequestMapping("/yesanser")
    public ModelAndView yesanser(int curPage ){
        ModelAndView mav=new ModelAndView();
        PageBean page=service1.findallcommentsyes(curPage);
        mav.addObject("page",page);
        mav.setViewName("admin/product/commentsyes");

        return mav;
    }

    @RequestMapping("/updateanser")
    public ModelAndView updateanser(String anser,String gid ){
        ModelAndView mav=new ModelAndView();
        Comments comments=service1.findcommentsbygid(gid);
        System.out.println(        comments.getComment()
);
        comments.setAnser(anser);
        comments.setState(1);
        service1.updatecomments(comments);
        mav.addObject("msg","回复成功");
        mav.setViewName("admin/welcome");

        return mav;
    }
}
```