# 目　录

[目　录 I](#_Toc76453375)

[第一章 需求分析 1](#_Toc76453376)

[1.1 课程设计题目 1](#_Toc76453377)

[1.2 课程设计任务及要求 1](#_Toc76453378)

[1.3 系统开发技术和环境 1](#_Toc76453379)

[1.3.1 IDEA与Tomcat简介 2](#_Toc76453380)

[1.3.2 JSP和Servlet简介 2](#_Toc76453381)

[1.3.3 Redis简介 3](#_Toc76453382)

[1.3.4 Ajax简介 3](#_Toc76453383)

[1.3.5二维码支付 3](#_Toc76453384)

[1.3.6 MVC 3](#_Toc76453385)

[第二章 系统设计 5](#_Toc76453386)

[2.1 系统设计方法三原则 5](#_Toc76453387)

[2.2 系统功能模块概述和分析 5](#_Toc76453388)

[2.3 系统功能模块设计 6](#_Toc76453389)

[2.4 系统数据库设计 7](#_Toc76453390)

[第三章 前台设计和实现 10](#_Toc76453391)

[3.1 首页设计 10](#_Toc76453392)

[3.2 用户模块 11](#_Toc76453393)

[3.2.1 用户注册 11](#_Toc76453394)

[3.2.2 用户登录 12](#_Toc76453395)

[3.2.3 用户退出 12](#_Toc76453396)

[3.3 商品模块 12](#_Toc76453397)

[3.3.1 商品展示 12](#_Toc76453398)

[3.3.2 商品详情 13](#_Toc76453399)

[3.4 购物车模块 13](#_Toc76453400)

[3.4.1添加商品到购物车 13](#_Toc76453401)

[3.4.2我的购物车 14](#_Toc76453402)

[3.5 订单模块 14](#_Toc76453403)

[3.5.1生成订单 14](#_Toc76453404)

[3.5.2我的订单 15](#_Toc76453405)

[3.5.3支付订单 16](#_Toc76453406)

[第四章 后台设计与实现 18](#_Toc76453407)

[4.1 管理员登录 18](#_Toc76453408)

[4.2 后台主页 18](#_Toc76453409)

[4.3用户管理页面 19](#_Toc76453410)

[4.4分类管理模块 19](#_Toc76453411)

[4.5商品管理模块 19](#_Toc76453412)

[4.6订单管理模块 20](#_Toc76453413)

[4.6.1查看所有订单 20](#_Toc76453414)

[4.6.2查看订单所有状态 21](#_Toc76453415)

[第五章 课程设计总结与体会 22](#_Toc76453416)

[5.1 课程设计总结 22](#_Toc76453417)

[5.2 心得体会 22](#_Toc76453418)

[参考文献 23](#_Toc76453419)

[致谢 24](#_Toc76453420)

# 第一章 需求分析

作为互联网的飞速发展，电子商务的热潮一度兴起，越来越多的商家在网上建立在线商城，向消费者展示一种新颖的购物理念。

网上购物在线网站致力于提供产品展示以及订购为核心的网上购物服务宣传自己的商店产品，并将自己的产品呈现给客户，让客户通过网站便能自由的选择购买商品。

该网站是通过用户登录浏览商品，加入购物车，提交订单，完成支付实现用户模块功能。网站后台主要包括商品管理，用户管理，订单管理来实现对用户信息以及购物信息的管理。

## 1.1 课程设计题目

在线购物商城的设计与实现

## 1.2 课程设计任务及要求

设计一个网上购物网站，它需要向用户提供静态和动态两类信息资源。所谓静态信息是指那些经常不变动或更新的资源，比如商城网站的登录注册，网站的使用注意事项，平台的简介等；动态信息是指那些随时变化的信息，比如商品的详细信息，商品的优惠信息，平台的促销活动等。另外网上购物网站还具有强大的交互功能，可以使商家和用户方便的传递信息，完成电子商务交易或EDI交易。

使用网上购物网站，既节省了传统线下商城需招募大量工作人员的资金问题，用户也可以清晰的看到商品的详细信息，从而足不出户便可买到心满意足的商品。另外对于商城的管理者，可以通过后台看到每天商品的销售情况，便于后期的商品进货以及销售策略调整。对于商品的所有订单以及销售问题，管理者无论身处哪里都可登录后台进行业务的处理，从而减轻了管理者的工作负担，让管理者的工作更加灵活。

## 1.3 系统开发技术和环境

软件体系：B/S结构（Browser/Server，浏览器/服务器模式），是WEB兴起后的一种网络结构模式，[WEB](https://baike.baidu.com/item/WEB/150564)浏览器是客户端最主要的应用软件。这种模式统一了客户端，将系统功能实现的核心部分集中到[服务器](https://baike.baidu.com/item/%E6%9C%8D%E5%8A%A1%E5%99%A8/100571)上，简化了系统的开发、维护和使用。客户机上只要安装一个浏览器，如[Netscape Navigator](https://baike.baidu.com/item/Netscape%20Navigator/1014148)或[Internet Explorer](https://baike.baidu.com/item/Internet%20Explorer/1537769)，服务器安装[SQL Server](https://baike.baidu.com/item/SQL%20Server/245994)、[Oracle](https://baike.baidu.com/item/Oracle/301207)、[MYSQL](https://baike.baidu.com/item/MYSQL/471251)等数据库。浏览器通过[Web Server](https://baike.baidu.com/item/Web%20Server/9306055) 同数据库进行数据交互。

### 1.3.1 IDEA与Tomcat简介

IDEA 全称 IntelliJ IDEA，是java编程语言开发的集成环境。IntelliJ在业界被公认为最好的java开发工具，尤其在智能代码助手、代码自动提示、重构、JavaEE支持、各类版本工具([git](https://baike.baidu.com/item/git/12647237)、[svn](https://baike.baidu.com/item/svn/3311103)等)、JUnit、CVS整合、代码分析、 创新的GUI设计等方面的功能可以说是超常的

Tomcat 服务器是一个免费的开放源代码的Web 应用服务器，属于轻量级应用[服务器](https://baike.baidu.com/item/%E6%9C%8D%E5%8A%A1%E5%99%A8)，在中小型系统和并发访问用户不是很多的场合下被普遍使用，是开发和调试JSP 程序的首选。对于一个初学者来说，可以这样认为，当在一台机器上配置好Apache 服务器，可利用它响应[HTML](https://baike.baidu.com/item/HTML)（[标准通用标记语言](https://baike.baidu.com/item/%E6%A0%87%E5%87%86%E9%80%9A%E7%94%A8%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80/6805073)下的一个应用）页面的访问请求。实际上Tomcat是Apache 服务器的扩展，但运行时它是独立运行的，所以当你运行tomcat 时，它实际上作为一个与Apache 独立的进程单独运行的。

### 1.3.2 JSP和Servlet简介

JSP（全称JavaServer Pages）是由[Sun Microsystems](https://baike.baidu.com/item/Sun%20Microsystems)公司主导创建的一种[动态网页技术](https://baike.baidu.com/item/%E5%8A%A8%E6%80%81%E7%BD%91%E9%A1%B5%E6%8A%80%E6%9C%AF/9415956)标准。JSP部署于网络服务器上，可以响应客户端发送的请求，并根据请求内容动态地生成[HTML](https://baike.baidu.com/item/HTML)、[XML](https://baike.baidu.com/item/XML)或其他格式文档的[Web](https://baike.baidu.com/item/Web)网页，然后返回给请求者。JSP技术以[Java](https://baike.baidu.com/item/Java)语言作为[脚本语言](https://baike.baidu.com/item/%E8%84%9A%E6%9C%AC%E8%AF%AD%E8%A8%80)，为用户的[HTTP](https://baike.baidu.com/item/HTTP)请求提供服务，并能与服务器上的其它Java程序共同处理复杂的业务需求。

Servlet（Server Applet）是[Java](https://baike.baidu.com/item/Java/85979) Servlet的简称，称为小服务程序或服务连接器，用Java编写的[服务器](https://baike.baidu.com/item/%E6%9C%8D%E5%8A%A1%E5%99%A8/100571)端程序，具有独立于平台和[协议](https://baike.baidu.com/item/%E5%8D%8F%E8%AE%AE/13020269)的特性，主要功能在于交互式地浏览和生成数据，生成动态[Web](https://baike.baidu.com/item/Web/150564)内容。

### 1.3.3 Redis简介

Redis（Remote Dictionary Server )，即远程字典服务，是一个开源的使用[C语言](https://baike.baidu.com/item/C%E8%AF%AD%E8%A8%80)编写、支持网络、可基于内存亦可持久化的日志型、Key-Value内存[数据库](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%BA%93/103728)，并提供多种语言的API。

对于那些不经常变化的数据比如导航栏上的分类信息，每次访问页面都向数据库发送请求，会导致数据库压力过大而且数据相应过慢。我们可以把这些经常使用的但又不轻易改变的信息保存到redis中，这样即可提高页面加载速度也可减少数据库的压力。

### 1.3.4 Ajax简介

Ajax即Asynchronous Javascript And XML（异步JavaScript和[XML](https://baike.baidu.com/item/XML/86251)）。 使用Ajax技术网页应用能够快速地将增量更新呈现在[用户界面](https://baike.baidu.com/item/%E7%94%A8%E6%88%B7%E7%95%8C%E9%9D%A2/6582461)上，而不需要重载（刷新）整个页面，这使得程序能够更快地回应用户的操作。

在登录注册过程中需要向后台发送请求来验证输入的信息是否合法。由于页面上信息量是很大的，如果发送同步请求最后会使整个页面重新加载，在这期间用户需要进行等待，无法进行其他操作。但这又是没有必要的，而我们使用ajax便可以避免这个问题，发送请求后，用户在数据被响应之前还可以去做其他的事比如继续浏览网页等，这样大大提高了用户的体验。

### 1.3.5二维码支付

二维码支付是一种基于账户体系搭起来的新一代无线支付方案。在该支付方案下，商家可把账号、商品价格等交易信息汇编成一个二维码，并印刷在各种报纸、杂志、广告、图书等载体上发布。

用户通过[手机客户端](https://baike.baidu.com/item/%E6%89%8B%E6%9C%BA%E5%AE%A2%E6%88%B7%E7%AB%AF/1944201)扫拍二维码或商家使用电子支付工具扫描用户的付款码，便可实现与商家账户的支付结算。最后，商家根据支付交易信息中的用户收货、联系资料，就可以进行商品配送，完成交易。

### 1.3.6 MVC模式

MVC即模型－视图－控制器，是Xerox PARC在八十年代为编程语言Smalltalk－80发明的一种软件设计模式，至今已被广泛使用。最近几年被推荐为Sun公司J2EE平台的设计模式，并且受到越来越多的使用ColdFusion和PHP的开发者的欢迎。

MVC是一种设计模式，它强制性的使应用程序的输入、处理和输出分开。使用MVC应用程序被分成三个核心部件：模型、视图、控制器。它们各自处理自己的任务.

（1）模型

模型表示企业数据和业务规则。在MVC的三个部件中，模型拥有最多的处理任务。例如它可能用象EJBs和ColdFusion Components这样的构件对象来处理数据库。被模型返回的数据是中立的，就是说模型与数据格式无关，这样一个模型能为多个视图提供数据。由于应用于模型的代码只需写一次就可以被多个视图重用，所以减少了代码的重复性。

（2）视图

视图是用户看到并与之交互的界面。对老式的Web应用程序来说，视图就是由HTML元素组成的界面，在新式的Web应用程序中，HTML依旧在视图中扮演着重要的角色，但一些新的技术已层出不穷，它们包括Macromedia Flash和象XHTML，XML/XSL，WML等一些标识语言和Web services.如何处理应用程序的界面变得越来越有挑战性。MVC一个大的好处是它能为你的应用程序处理很多不同的视图。在视图中其实没有真正的处理发生，不管这些数据是联机存储的还是一个雇员列表，作为视图来讲，它只是作为一种输出数据并允许用户操纵的方式。

（3）控制器

控制器接受用户的输入并调用模型和视图去完成用户的需求。所以当单击Web页面中的超链接和发送HTML表单时，控制器本身不输出任何东西和做任何的处理。它只是接收请求并决定调用哪个模型构件去处理请求，然后确定用哪个视图来显示模型处理返回的数据。

综上所述，MVC的处理过程是首先控制器接收用户的请求，并决定应该调用哪个模型来进行处理，然后模型用业务逻辑来处理用户的请求并返回数据，最后控制器用相应的视图格式化模型返回的数据，并通过表示层呈现给用户。

# 第二章 系统设计

## 2.1 系统设计方法三原则

人类在复杂的工程技术系统如能源、交通等建设方面，积累了丰富的经验，为研究复杂系统如管理系统提供了科学的指导性方法论，其主要原则如下：

（1）整体性原则。系统是相互联系，相互作用的诸要素组成的综合体。我们必须从整体和各组成部分的相互关系来考察事物，从整体目标和功能出发，正确处理系统各组成部分之间的相互关系和相互作用。

（2）分解—协调原则。就是把复杂问题化成若干相对简单的子问题以方便求解。若子系统的问题比较复杂，还可以再分。但在处理各类子问题时，必须根据系统的整体功能和目标，协调各子系统的行为、功能与目标，以保证整体功能目标的实现。

（3）目标优化原则。所谓目标优化原则对简单的系统来说，是求最优解，对复杂系统来说，求的是满意解。一定要注意，目标优化原则并不简单是求最优解的问题。

这里最需要指出的是：以上三原则是系统方法中处理复杂系统问题的三个主要原则，并非是全部原则。在处理实际问题时，还需要在这些原则的指导下，根据问题的特点，确定求解的具体方法和策略。

## 2.2 系统功能模块概述和分析

在线商城是最为普遍的电子企业对客户（B2B）模式，主要包括会员注册、

会员登录、订单管理、购物车、支付等基本功能。此外本系统也将实现在线图书销售系统的后端管理，包括商品的添加和修改、商品种类的添加和修改、订单的处理等功能。本系统完全基于JSP+Servlet技术，在系统的设计和开发过程中严格遵守软件工程的规范，运用软件设计模式，从而减少系统模块之间的耦合性，力求做到系统的的稳定性、可扩展性。

在线购物商城的主要功能如下：

1. 前台（商品购买）模块：
   1. 用户管理：注册会员、登录、激活、退出。
   2. 商品显示：根据商品的分类显示对应的商品。
   3. 购物车管理：向购物车中添加商品、删除商品、修改商品的数量、查看我的购物车。
   4. 订单管理：通过购物车中的商品生成订单、查看我的订单、查看某个订单的详细、订单支付、确认收货。
2. 后台（管理员部分）
   1. 管理员：管理员登录。
   2. 分类管理：查看所有分类、增加分类、修改分类、删除分类。
   3. 商品管理：查看所有商品、增加商品、修改商品、删除商品。
   4. 订单管理：按照状态查看订单、发货、查看订单详情。

## 2.3 系统功能模块设计

根据系统功能分析，可以画出系统的功能模块图。如图 2-1所示：

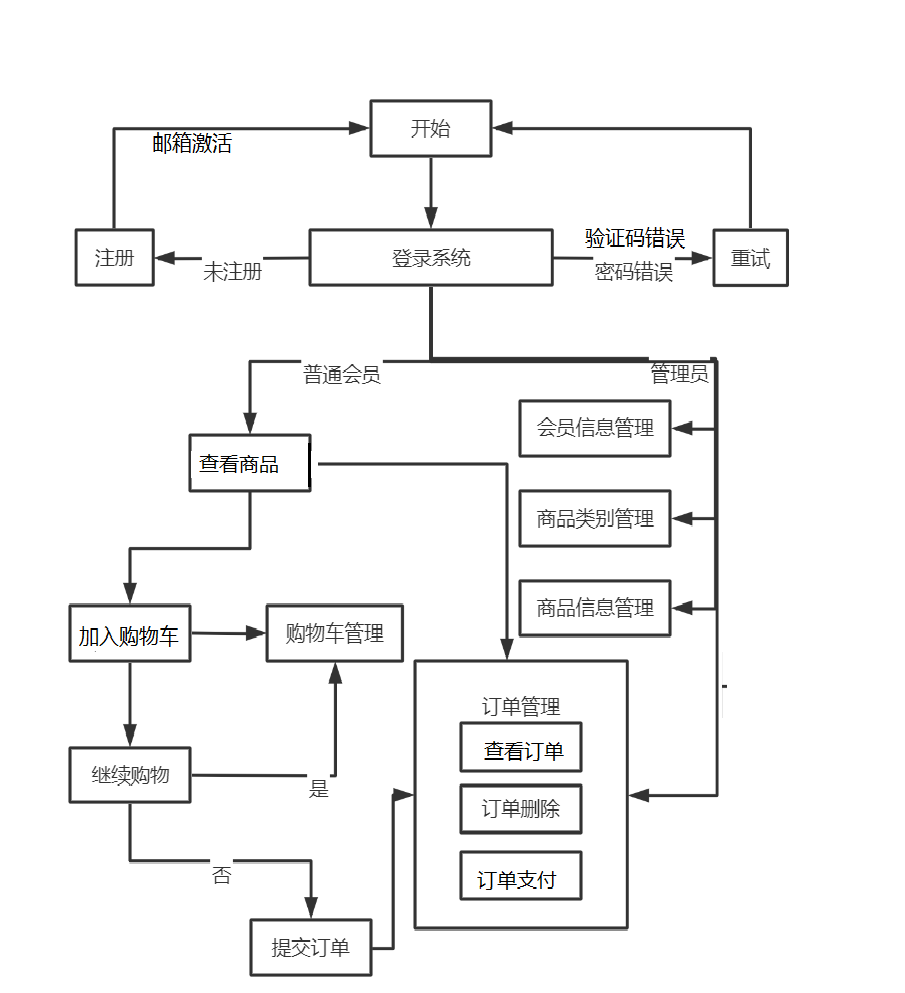


图2-1 商城网站功能模块图

## 2.4 系统数据库设计

数据库表设计主要是把概念结构设计时设计好的基本E-R图转换为与选用DBMS产品所支持的数据模型相符合的逻辑结构。它包括数据项、记录及记录间的联系、安全性和一致性约束等等。导出的逻辑结构是否与概念模式一致，从功能和性能上是否满足用户的要求，要进行模式评价。

数据库表设计

1.用户表（USER）用于存储用户的相关内容，如表2-1所示：

表2-1 用户表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名称 | 字段说明 | 类型 | 位数 | 属性 |
| 1 | uid | 用户编号 | varchar | 32 | 非空,主键 |
| 2 | username | 用户名 | varchar | 20 | 必填,非空 |
| 3 | password | 密码 | varchar | 20 | 必填,非空 |
| 4 | name | 姓名 | varchar | 20 | 非空 |
| 5 | email | 邮件 | varchar | 20 | 非空 |
| 6 | telephone | 电话 | varchar | 20 | 非空 |
| 7 | birthday | 生日 | date |  | 非空 |
| 8 | sex | 性别 | varchar | 10 | 非空 |
| 9 | state | 激活状态 | int | 11 | 非空 |
| 10 | code | 激活码 | varchar | 64 | 非空 |

2.产品表（PRODUCT）用于存储商品的详细信息，如表2-2所示：

表2-2 商品信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名称 | 字段说明 | 类型 | 位数 | 属性 |
| 1 | pid | 商品编号 | varchar | 32 | 主键，非空 |
| 2 | pname | 商品名称 | varchar | 50 | 非空 |
| 3 | market\_price | 市场价格 | double |  | 非空 |
| 4 | shop\_price | 商城价格 | double |  | 非空 |
| 5 | pimage | 商品图片 | varchar | 200 | 非空 |
| 6 | pdate | 生产日期 | date |  | 非空 |
| 7 | is\_hot | 是否热门 | int | 11 | 非空 |
| 8 | pdesc | 商品描述 | varchar | 255 | 非空 |
| 9 | pflag | 是否上架 | int | 11 | 非空 |
| 10 | cid | 类别编号 | varchar | 32 | 非空 |

3.订单表（ORDERS）用于存储订单的详细信息，如表2-3所示：

表2-3 订单详情表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名称 | 字段说明 | 类型 | 位数 | 属性 |
| 1 | oid | 订单id | varchar | 32 | 主键，非空 |
| 2 | ordertime | 订单时间 | datetime |  | 非空 |
| 3 | total | 总金额 | double |  | 非空 |
| 4 | state | 订单状态 | int | 11 | 非空 |
| 5 | address | 收获地址 | varchar | 30 | 非空 |
| 6 | name | 收货人 | varchar | 20 | 非空 |
| 7 | telephone | 联系电话 | varchar | 20 | 非空 |
| 8 | uid | 用户编号 | varchar | 32 | 非空 |

4.订单项表（ORDERITEM）用于存储订单的每一项，如表2-4所示

表2-4 订单项表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名称 | 字段说明 | 类型 | 位数 | 属性 |
| 1 | itemid | 订单项id | varchar | 32 | 主键，非空 |
| 2 | count | 商品数量 | int | 11 | 非空 |
| 3 | subtotal | 小计 | double |  | 非空 |
| 4 | pid | 商品id | varchar | 32 | 非空 |
| 5 | oid | 订单id | varchar | 32 | 非空 |

5.商品类别表（CATEGORY）用于存储商品的类别信息，如表2-5所示：

表2-5 商品类别表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名称 | 字段说明 | 类型 | 位数 | 属性 |
| 1 | cid | 类别编号 | varchar | 32 | 主键，非空 |
| 2 | cname | 类别名称 | varchar | 20 | 非空 |

# 第三章 前台设计和实现

## 3.1 首页设计

首页模块包括4个主要的部分：导航栏，轮播图，最新商品，最热商品。

页面如图3-1、3-2、3-3所示：



图3-1 购物商城的首页

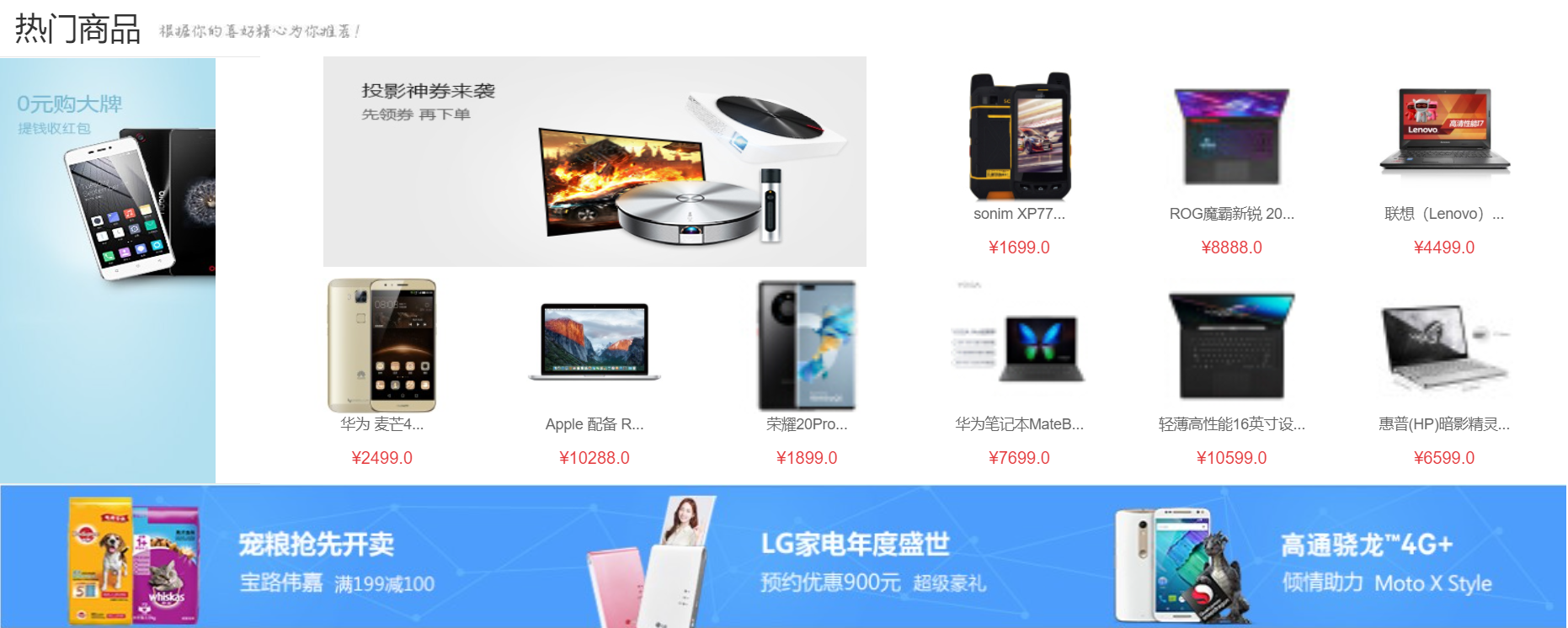


图3-2 购物商城的首页

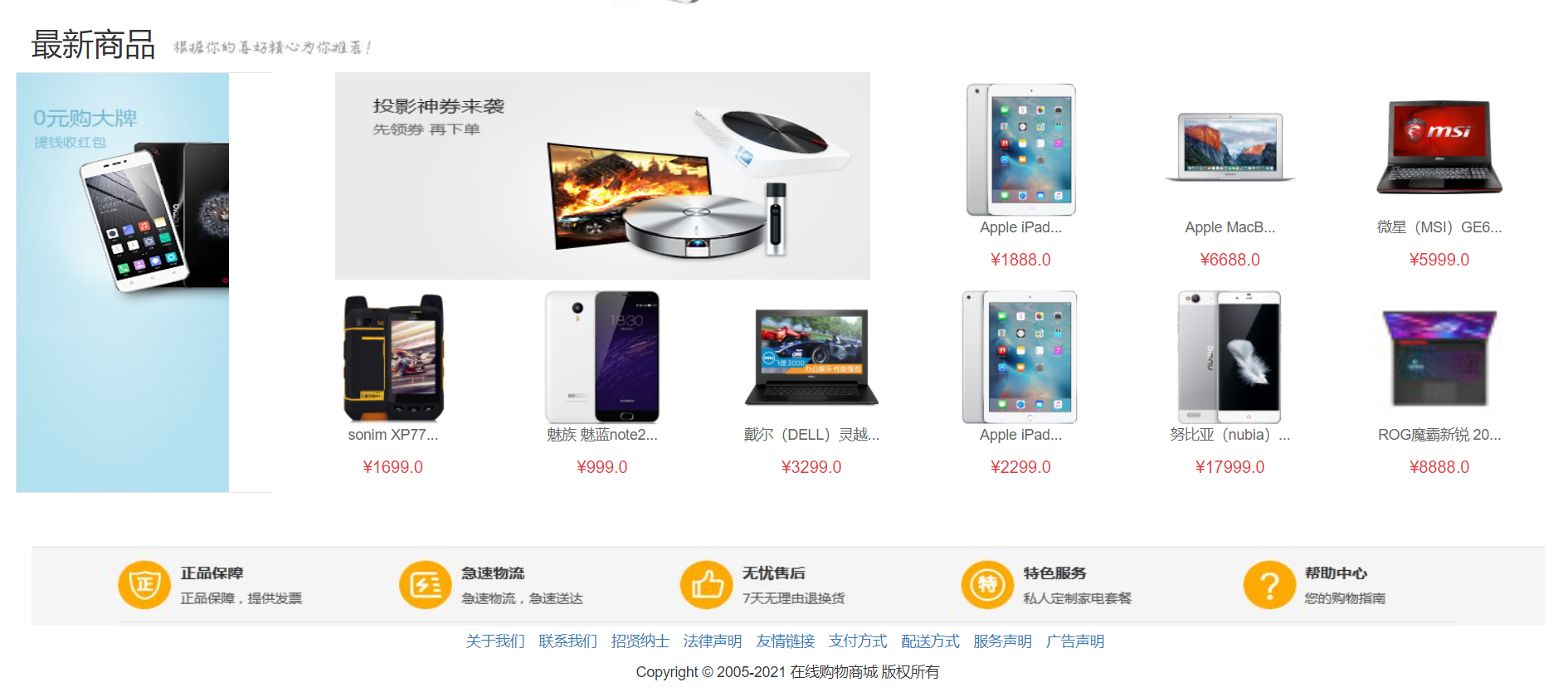


图3-3 购物商城的首页

## 3.2 用户模块

注册用户是构成网站主体的一个重要组成部分，网站设置注册用户的目的之一在于方便网站用户信息的管理。

### 3.2.1 用户注册

用户在登录之前需要先进行注册，在首页中点击注册按钮就可到达注册页面。

页面如图3-4所示：

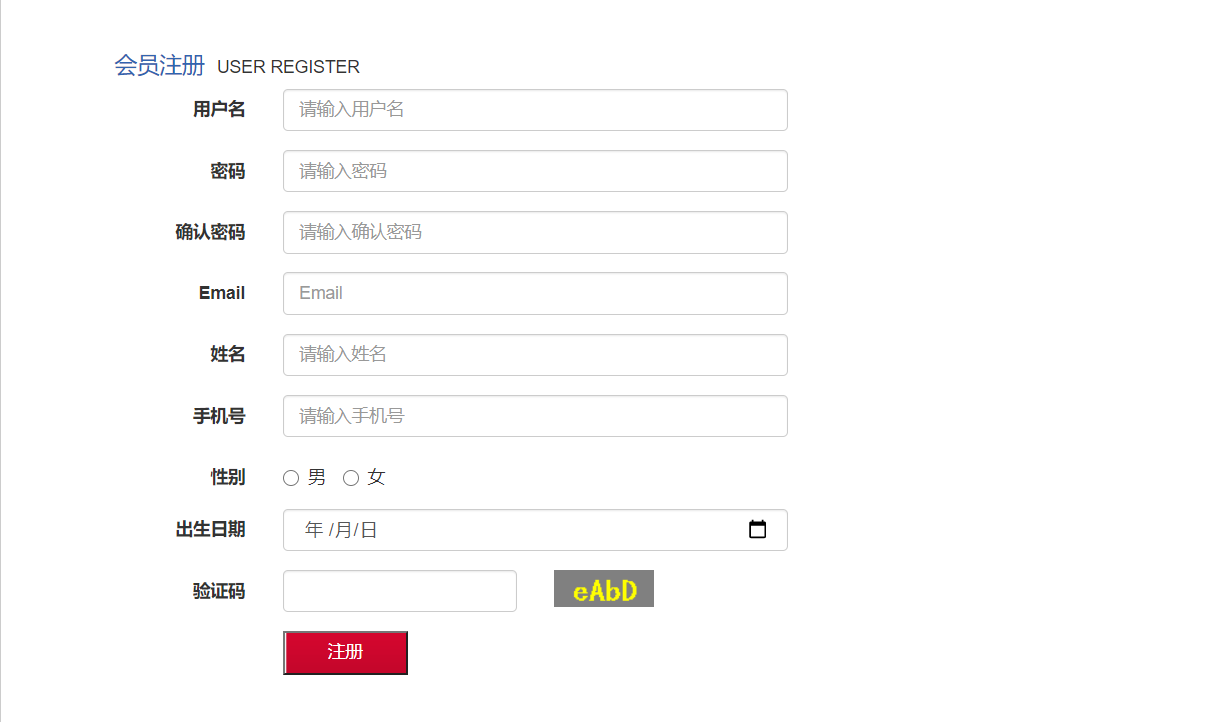


图3-4注册页面

### 3.2.2 用户登录

在首页点击“登录”按钮就可以来到登录页面

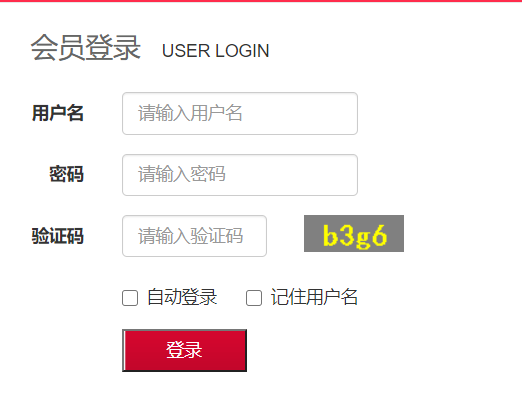


图3-5登录页面

### 3.2.3 用户退出

当用户登录之后点击“退出”按钮便可完成退出。

## 3.3 商品模块

### 3.3.1 商品展示

用户通过点击导航栏上对应商品的类别，便可到达指定的商品界面。页面如图3-6所示：

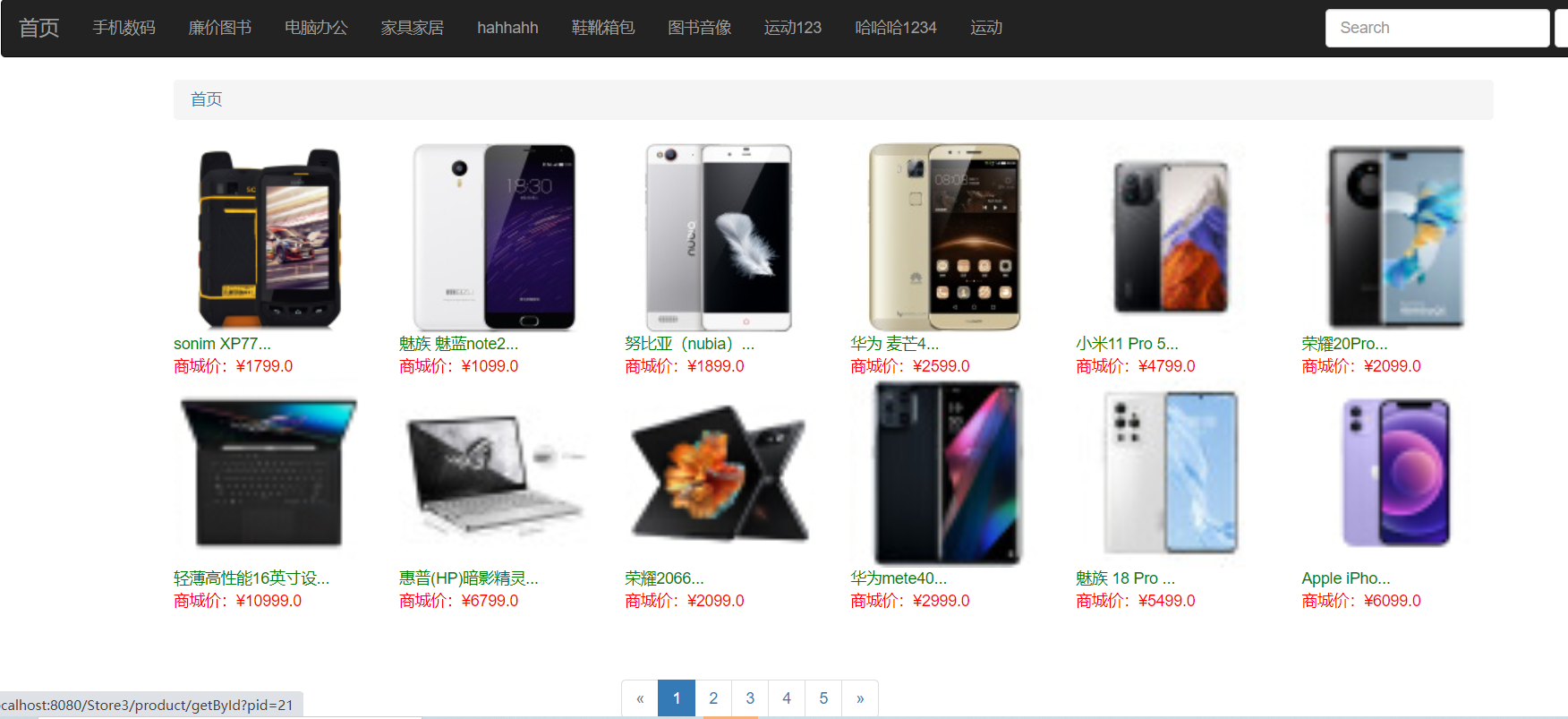


图3-6商品展示页面

### 3.3.2 商品详情

点击商品下方的超链接页面将跳转到该商品的详情页面。如图：3-7所示



图3-7商品详情页面

## 3.4 购物车模块

### 3.4.1添加商品到购物车

用户在商品详情页面，通过点击加入购物车，可将商品加入到购物车，页面也将挑战到购物车页面。如图3-8所示



图3-8购物车页面

### 3.4.2我的购物车

点击购物车，购物车页面会显示所有车中的所有条目，每个条目会显示出商品图片、商品名称、商品价格、数量、小计。在该页面也可以进行删除、更改商品数量、清空购物车等操作。如图3-9所示



图3-9购物车页面

## 3.5 订单模块

### 3.5.1生成订单

在购物车列表页面中，点击提交订单按钮，将会对购物车生成订单，页面将跳转到订单详情页面，同时购物车被清空，用户可回到商品页面继续购物，购物完成后再进行支付。如图3-10所示



图3-10 订单详情页面

### 3.5.2我的订单

单击我的订单，将会显示用户所有的订单的详细信息。如果用户未登录进行访问，则会被拒绝访问并给与提示。如图3-11、3-12所示。



图3-11提示页面



图3-12 订单详情页面

### 3.5.3支付订单

进入我的订单，对于未付款的商品点击去付款按钮，页面将跳转到支付页面。点击确认订单，将自动弹出订单确认页面。进行支付后页面将给与支付成功的提示信息。如长时间未支付，则给与支付超时，必须刷新二维码后再进行支付。如图3-13、3-14、3-15、3-16所示。



图3-13 订单确认页面



图3-14 订单信息确认页面



图3-15 支付成功页面

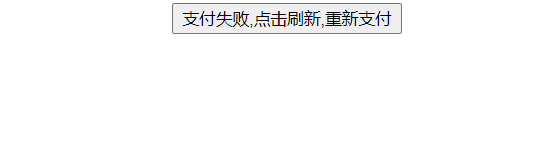


图3-15 支付超时页面

# 第四章 后台设计与实现

商城后台的设计是为管理员方便管理系统而设计的，其中包括分类管理、商品管理、以及订单管理、用户管理。

## 4.1 管理员登录

进入到管理员登录页面，登录成功后到达后台主页。如图4-1所示。

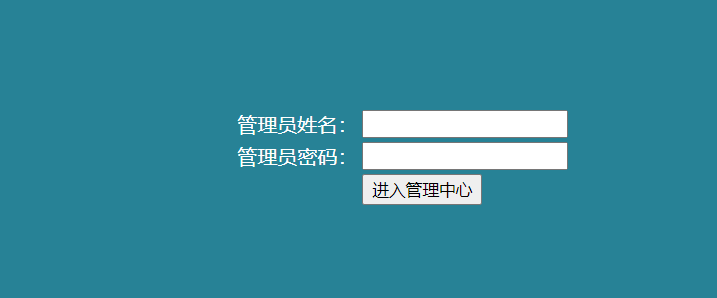


图4-1管理员登录界面

## 4.2 后台主页

用户名和密码验证通过后，进入到后台主页。如图4-2所示。



图4-2后台主界面

## 4.3用户管理页面

单击展示用户按钮可以展示所有用户，并可对用户进行删除和修改操作。



图4-3用户管理界面

## 4.4分类管理模块

单击分类管理模块可以对商品的分类信息进行增加、修改、删除操作。如图4-4所示。



图4-4用户管理界面

## 4.5商品管理模块

点击商品分类可以对商品进行增加、修改、删除、操作。如图4-5、4-6、4-7所示



图4-5商品展示界面

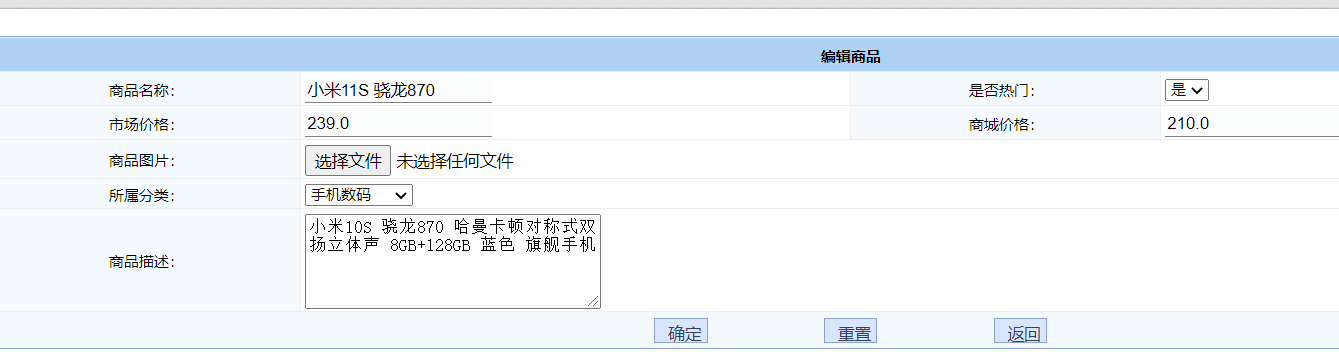


图4-6商品修改界面

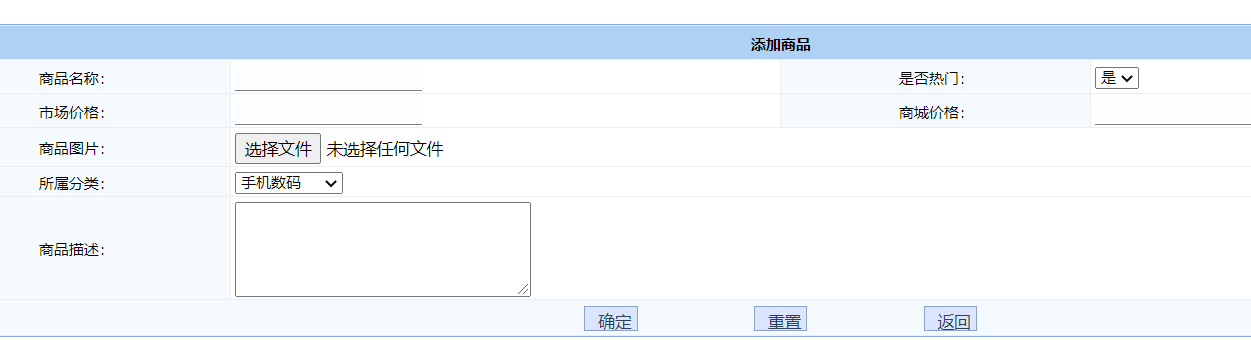


图4-7商品添加界面

## 4.6订单管理模块

### 4.6.1查看所有订单

点击订单列表，将显示所有的订单，对于每个订单，点击订单详情将显示出该订单的详细情况。如图4-8所示。



图4-8订单展示界面

### 4.6.2查看订单所有状态

点击订单管理下的按钮可以展示未付款订单、已付款订单、已发货订单、已完成订单。管理员在已付款界面对于已经支付的订单进行发货操作。如图4-10、4-11所示。



图4-10未付款订单界面



图4-11已付款订单界面

# 第五章 课程设计总结与体会

随着互联网技术的大力发展，人们的日常生活已经离不开网络。未来时代我们的生活和工作一定会更加借助于信息技术的发展，越来越方便、快捷、安全、高效，在线网络的发展过程以及当下的应用情况和发展趋势，我们可以充分地相信网络技术将极大的改变我们的生活和工作方式，使我们的生活变得更加美好、舒适。

## 5.1 课程设计总结

本文按照项目开发基本流程，采用JSP+Servlet技术，数据库MySQL,开发工具IDEA，实现了一个在线购物商城的设计和开发。

通过开发这个在线商城，我较全面的掌握了Java的基本知识和编程技巧，并再开发过程中我的Java开发能力得到了进一步的提高。如更深一步了解MVC设计模式、软件三层框架、SQL语言的使用。

## 5.2 心得体会

在开发过程中我学到了一些经验：系统分析的好坏将决定着的系统开发成功与否，一份好分析设计将是成功开发主要因素。我们在着手开发之前不要急于编程，先应有较长的时间去把分析做好，做好数据库设计工作，写出相关的开发文档等。然后再开始编写程序代码，这样做到每写一步代码心底有数，有条不絮。当然也有些还需待继续深入地方如：控件的应用等。

此外，我还觉得，我个人在这次设计中走了很多弯路。主要是因为平时很少接触软件开发工作，在应用方面缺乏经验，以后还需要更多的努力。

对我来说，这次设计的本身所产生的影响，还远远没有结束，我从本次毕业设计中学到了许多课本上没有的知识。通过自己的学习和努力；通过老师的指导和教育，使我不仅仅在知识水平和解决实际问题的能力上有了很大的提高。还从思想的深处体会到，要把自己的所学变成现实时所将面对的种种难题。

# 参考文献

[1] 陈庆荣. 基于B/S模式应用的研究与设计[J]. 福建电脑, 2018, 34(02): 125-126+67.

[2] 余炀, 曲毅, 孙亦乐. 基于ApacheTomcat的一站式Java应用服务器解决方案[J]. 中国金融电脑, 2018(01): 59-63.

[3] 王天顺, 程杰, 张志文. MyBatis在《Web中间件技术》课程中的应用[J].教育教学论坛, 2019(38): 204-205.

[4] 韩思凡. Web开发中的JSP与HTML的基础应用[J]. 科学技术创新, 2020(14): 71-72.

[5] 胡雅丽. 基于BootStrap+JavaScript语言简易计算器的实现[J]. 太原学院学报(自然科学版), 2020, 38(01): 73-76.

[6] 郭节. 基于Log4j改进的日志存储应用研究[J]. 软件导刊, 2011, 10(01): 60-61.

[7] 戎戟. Servlet过滤器应用分析[C]. 办公自动化学会. OA’2007办公自动化学术研讨会论文集. 办公自动化学会: 北京办公自动化杂志社, 2007: 160-162.

[8] 崔天鑫. Servlet监听器在基于移动Web在线考试系统中的应用[J]. 电脑编程技巧与维护, 2019(10): 51-53+82.

[9] 滕建华. Web系统功能测试一体化课程开发探索[J]. 信息与电脑(理论版), 2017(24): 232-234.

# 致谢

在这里我要感谢辛苦帮助我的老师和同学，正是由于他们的无私帮助，我才能顺利完成这次的程序设计。是你们一次又一次帮我讲解在设计过程中遇到的难点和复杂的逻辑结构。至此我依然能够记得老师耐心细致的在讲台上认真讲解本次课程设计的要点，面对同学们的疑惑和不解。您总是不厌其烦耐心的讲解。