

摘 要

随着社会的进步，互联网的不断发展，人们的经济水平和思想理念也在日益提高。这就导致了大部分人的消费方式和消费理念也在发生变化，更多人倾向不出门就可以买到物美价廉的商品，这就促使了网购平台的快速发展。为了能提供更好的消费方式来顺应当下的发展，本项目的开发已成为一种必然趋势。

本系统使用 Java 语言作为开发语言，以 IDEA 作为集成开发环境，采用 SpringCloud 框架进行搭建，选用 MySQL5.7 作为数据库支持。用户在注册登录后能够通过平台查看商品及其详细信息，并可以将商品添加至购物车后进行支付，还可以进行修改个人信息和收货地址，查看订单等操作。经过系统测试，测试结果证明本系统功能完善，界面简洁合理，能够满足用户群体的需求，且具有较高的可用性和实用性。

关键词：Java 语言；Spring Cloud 框架；MySQL 数据库

ABSTRACT

With the progress of society and the continuous development of the Internet, people's economic level and ideological concepts are also improving day by day. This has led to changes in the consumption patterns and consumption concepts of most people, and more people tend to buy high-quality and low-cost goods without going out, which has promoted the rapid development of online shopping platforms. In order to provide better consumption patterns to adapt to the next development, the development of this project has become an inevitable trend.

The system uses Java language as the development language, IDEA as the integrated development environment, SpringCloud framework for construction, and MySQL 5.7 as database support. After registering and logging in, users can view the products and their details through the platform, and can add the products to the shopping cart to pay, modify personal information and delivery address, view orders and other operations. After system testing, the test results prove that the system has perfect functions, simple and reasonable interface, can meet the needs of user groups, and has high usability and practicality.

Key words: Java language; Spring Cloud Framework; MySQL database

目 录

第 1 章 绪论	1
1.1 课题研究的背景及意义	1
1.2 国内外研究现状	1
1.3 项目内容概述	2
第 2 章 系统分析	3
2.1 可行性分析	3
2.1.1 经济可行性	3
2.1.2 技术可行性	3
2.2 用户需求分析	3
2.3 功能需求分析	4
2.4 硬件分析	6
第 3 章 系统设计	7
3.1 系统总体设计思想	7
3.1.1 系统结构设计	7
3.1.2 关键性开发技术介绍	7
3.2 功能设计	8
3.2.1 整体设计描述	8
3.2.2 系统流程设计	9
3.3 数据库表设计	11
第 4 章 系统实现	13
4.1 数据库连接的实现	13
4.2 登录验证功能的实现	14
4.3 购物车功能的实现	15
4.4 订单功能的实现	18
4.5 支付功能的实现	24
4.6 商品分类功能的实现	26
4.7 个人中心功能的实现	28
4.8 商品推荐功能的实现	30
4.9 商品管理功能的实现	33
4.10 会员管理功能的实现	36

4.11 分类管理功能的实现.....	37
4.12 订单管理功能的实现.....	40

第 5 章 系统测试	43
5.1 软件测试	43
5.2 系统测试	43
5.2.1 前台首页的测试	43
5.2.2 购物车模块的测试	43
5.2.3 登陆模块的测试	44
5.2.4 商品管理模块测试	44
总结	46
参考文献	47
致谢	48

第 1 章 绪论

随着互联网的快速发展，人们的生活变得越来越丰富，办公如今慢慢的向自动化发展，人力也在不断地减少，效率也在不断的提高，随之而来的电商平台也孕育而生 Error! Reference source not found.。

1.1 课题研究的背景及意义

在互联网技术高速发展的当今，人们的生活节奏越来越快，因此，都想尽量从忙碌的工作中挤出一点时间，比如睡觉、出行、购物等，都要将效率最大化，睡觉、出行等可以挤压出来的时间十分有限，然而，在购物上的效率利用是有着很大的空间的。因此，开发一种效率高、操作简单、方便快捷的购物平台成了解决用户对于效率需求的最佳途径。

此次开发的线上数码产品交易平台是一个具有搜索、查看、购买、管理等功能的 Java 应用软件。在现在的社会中，随着生活的不断加快，工作带给人们的压力越来越大。很多人很想去购买自己想要的商品，但是可以用来购物的时间却很少。人们在晚上下班后已经非常疲惫，特别是回到家中后根本不想出门去购物。特别对于数码商品来说型号版本众多，需要精挑细选，而且数码商品专卖店的营业时间和消费者上下班的时间一样。这时，线上数码产品交易平台便可以体现出它的研究价值，在人们下班后回到家中时，便可以打开电脑登陆本平台查看各类数码商品的信息，找到自己喜欢的商品便可以直接进行购买。省去了逛专卖店的时间，提高了时间的利用率。

1.2 国内外研究现状

在国外的发达国家和地区，网上购物已经发展形成了非常完善的体系，而且还发布了对于网上购物的相关政策及法律法规。在国内，网上购物发展趋势迅猛。到 2022 年年底，我国网上零售额 13.79 万亿元，同比增长 4%，实现稳步增长，跨境电商发展迅速，电商业态新模式彰显新活力。

随着网络信息技术的迅速发展，中国电商也逐步由发展迈向了规范。根据中国商务部于 2020 年发布的《中国电子商务报告（2020）》报告，在 2016-2020 年期间，全国电子商务交易额从 26.10 万亿元增长到 37.21 万亿元，年均增长 9.3%。中国网购用户规模已达到 7.82 亿，连续多年保持全球规模最大、最具活力的网络零售市场 Error! Reference source not found.。因此我国网上购物前景巨大，无论是从交易额还是人数上来看，都位居世界前列。

1.3 项目内容概述

本次开发的项目名称叫线上数码产品交易平台。线上数码产品交易平台是一个功能强大的应用软件。用户功能里包括商品查看、添加购物车、商品购买等，用户注册后便可在本网站进行购物，在商品详情中点击购买或添加购物车，这件商品便会加入到购物车中，用户可以对购物车进行删除、结算等操作。

在开发中要求开发者熟练操作 IDEA 开发环境，掌握 Spring Cloud 框架和 Java 开发语言，能解决开发中的问题，并能调试解决代码错误问题。在这个项目中，另一个重点就是如何解决 MySQL5.7 数据库中处理交互问题。数据库承载着整个系统数据，既能对数据进行处理，又能对数据进行存储，因此非常重要。在开发的过程中，会涉及到后台管理对数据库数据的处理，如何做好前后台的交互工作，也是本项目开发的一个重要方面。

第 2 章 系统分析

2.1 可行性分析

2.1.1 经济可行性

在互联网技术飞速发展的今天，已经有非常多的工作都离不开计算机。计算机网络的普及，也让人们的工作更加方便。因为此次开发是以学习为目的，所以并不会涉及到资金问题。该系统使用 Java 语言开发，Java 是开源免费的。此外本系统使用的 MySQL 数据库、Maven 项目管理工具等等均为开源免费，这必然会从费用方面减少许多。所采用的 SpringCloud 框架是 Java 生态中最完整、使用最为广泛的开发技术，这不仅降低了开发成本而且提高了开发效率且在经济方面完全可行。

2.1.2 技术可行性

本次项目开发的系统是 Windows 10，开发环境是 IDEA，IDEA 作为当前市面上最好用的 Java 编译器，可以提高开发人员的工作效率。其直观简洁的 GUI、动态的错误高亮提示、灵活的排版，使开发人员在开发过程中更加轻松。此次开发选用 Java 开发语言，它是一种面向对象的程序设计语言，也是当前使用最广泛的编程语言。Java 具语言简洁的独特优势，属于面向对象的分布式应用，以及最主要的是，Java 是一种动态语言。数据库采用 MySQL5.7 构建。它是开源免费的，大大降低了用户使用成本。MySQL 的特点是简单易用，可移植性强，可运行于 Windows、Linux、Unix 等不同的平台。将这些开发工具结合起来，可以发挥出更大的作用，为整个项目的开发提供了便利，从而说明本系统在技术方面可行。

2.2 用户需求分析

现在人们的生活节奏加快，工作也变得更加繁忙，人们的自由支配时间被不断地压缩，因此，在有限的空余时间里，去完成自己想做的事情，已经变成了人们的一种迫切需求。

由于网络的迅速发展，使得网上购物变得更加普遍，购物网站也越来越为人们所熟知。网上购物让消费者不用出门就可以买到自己想要的物品，所以网上购物这种新兴事物得到了广大互联网用户的认可。网上购物这种方式是时代发展所带来的概念，是创新意识的一种体现，网上购物的需求量也在上升，因此，使得这次的开发非常有必要。

2.3 功能需求分析

在当今的互联网世界里，网页的外部形态多种多样，不但在前端的美术设计上愈加完善，而且在功能上更是充分体现了人性化。这款软件的核心并不是软件中的前端 UI，而是软件中的功能设计。

就当前的技术而言，由于各类网站的发展日趋完善，网页的功能性已经不存在任何问题。而如何为使用者提供更好的服务，则成为开发人员思考的问题。这是由于在当代技术的持续发展下，各种新的事物层出不穷，从而导致了互联网上的信息形态也呈现出一种潮流。使用者持续地从互联网上获取新的潮流资讯，并对潮流的功能性服务进行体验，这就对开发者提出了更高的要求。所以，如何让这些软件变得更容易上手，才是开发人员想要解决的问题。

开发人员首先要考虑的是用户。用户进入网站后，首先进行注册登录，进入首页后要让用户一眼能看出这是一个关于数码商品的购物网站，特色商品要放在显眼的位置，并在页面中设置推荐商品、热销商品和新品推荐分区，且每个商品上都要有该商品的详情查询和购买提醒（在没有登录的情况下无法使用购买功能）。登录后一旦发现了自己感兴趣的东西，就可以点击上面的“购买”进行交易。在此期间，所有物品都会被加入到这个用户的购物车里，使用者可以自行管理自己的购物车，在不准备再买的情况下，可以通过“删除”键将物品从购物车里移走，也可以通过“继续购物”键返回主页面进行购物。在完成购买后，用户可以点击“支付”完成交易，并在交易中输入对应的用户信息，以供系统管理人员确定以及发货。

其次，是管理员管理系统的作用。首先，管理员需要在后台使用专门的账号、密码登陆，登陆成功之后，在用户管理中可以查看注册的用户，管理员可以看到所有的订单信息并对其进行处理操作。其次是商品管理，管理员能够对商品的信息、商品的状态进行操作，最后系统还能对商品分类、商品推荐等相关功能进行管理。

根据以上想法设计出相关功能，简单说明如下：

（1）商品分类管理模块

商品分类管理模块主要体现在管理员对商品分类的查看、编辑。可以增加一个一级分类，将同类的商品关联到下面，方便用户寻找商品。

（2）用户管理模块

用户管理模块的具体内容有用户注册、用户登录、用户密码修改等。

（3）购物车模块

购物车模块是指用户可以把商品添加到购物车中方便查看和统一结算，后台会对购物车中商品的数量、总价进行统计。使用者也可以从购物车里移除项目，当商品改变时，系统会重新计算。

(4) 商品管理模块

商品属性管理模块主要是指管理员能够查看和修改已有的商品信息，并且能够对平台内的商品进行添加、删除、上下架等操作

(5) 订单管理模块

订单管理模块的功能是用户可以查看或取消订单，管理员可以进行查看、删除、修改的操作。

(6) 支付管理模块

支付管理模块主要体现在用户点击购买后，系统会产生一个订单的 ID，并且会显示一个可供选择的支付页面。付款结果将于使用者付款后通知。

(7) 商品推荐管理模块

商品推荐的管理是可以把一些热销的、新潮的或者库存积压的商品放在推荐位，以达到清理库存，提高销售量的目的。

根据功能需求分析，用例图可以分为用户用例图和管理员用例图。如图 2.1、图 2.2 所示。

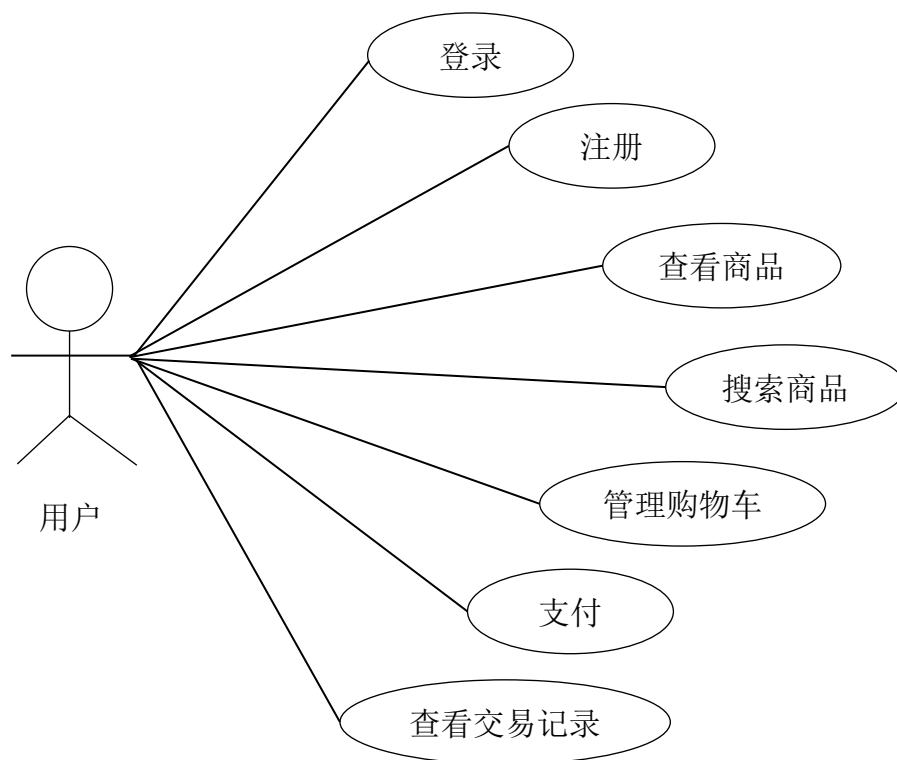


图 2.1 用户用例图

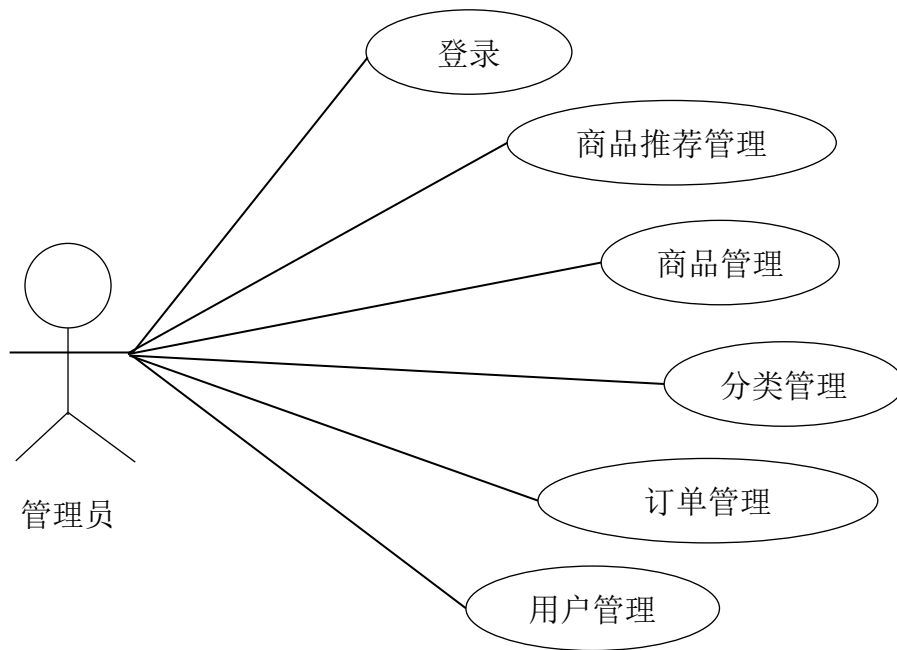


图 2.2 管理员用例图

2.4 硬件分析

本系统所需的计算机硬件最低配置，CPU: 2.0GHz ， RAM:3000MB ， 磁盘:120GB。

第 3 章 系统设计

3.1 系统总体设计思想

3.1.1 系统结构设计

在系统分析的基础上，把该系统划分为前台、后台和数据库三大部分。前台用户可查看商品及其详情等信息，还可以在注册登录后进行添加购物车，购买等操作。数据是整个系统中最重要的一部分，管理人员的工作大多都集中在数据库上，并且在前台及时地展示更新。

前台作为规模比较大的功能群之一，其所含信息量大面广，类型杂乱无序。设计成优化结构形式可有效地组织和分类平台信息内容，方便用户得到需要物品。这些部分中每个部分用尽可能少的东西展示出来，避免用户看得目不暇接，丧失耐性。在实现上述功能的同时，对前台结构形式进行设计，同时也为后面程序的编写做一个框架。

系统后台负责维护全平台信息及数据。对系统前台展示的信息，后台均应进行相应维护。并且为便于管理员进行维护，可把各管理模块组合在一个页面上，形成一个整体后台管理系统。

前台和后台结构设置完成后，在结构中加入数据库。后台将更新数据写入数据库，前台对数据库进行信息提取和显示。

3.1.2 关键性开发技术介绍

优秀的开发技术可以帮助开发人员提高开发效率、减少人力和提高质量。本系统采用的开发技术有 Java 语言、MySQL 数据库、SpringCloud 框架等，下面就关键性开发技术作一介绍。

Java 属于面向对象编程语言。是一个为应用程序开发而提供的计算平台。所以 Java 具有速度快，安全可靠等特点。考虑以 Java 为此次开发语言的原因是因为 Java 不依赖平台，它提供了在不同平台下的解释运行环境。Java 语言还具有简单、安全、分布式、多线程、可移植、开源等优点。

MySQL 是一种开放源代码的小型关联式数据库，相对于 Oracle 和 SQLServer 这样的大数据库，MySQL 的规模较小，功能也比较有限，但其体积更小，速度更快，成本更低，并且其提供的功能在稍复杂的应用中已够用，这些特性使得它成为本次项目开发的首选。

SpringCloud 是一系列框架的有序集合，它利用 SpringBoot 的开发便利性简化了分布式系统的开发，比如服务发现、服务网关、服务路由、链路追踪等。考虑 Spring Cloud 的原因有如下几方面。Spring Cloud 来源于 Spring，质量、稳定性、持续性均有保障，Spring Cloud

支持 **Spring Boot**，使其更容易实现业务的落地；使用门槛较低，可扩展性强；负载均衡：提高了跨多个计算资源的工作负载分配。

3.2 功能设计

3.2.1 整体设计描述

从总体上来看，该系统可以划分为八个主要的功能模块。如图 3.1 所示。

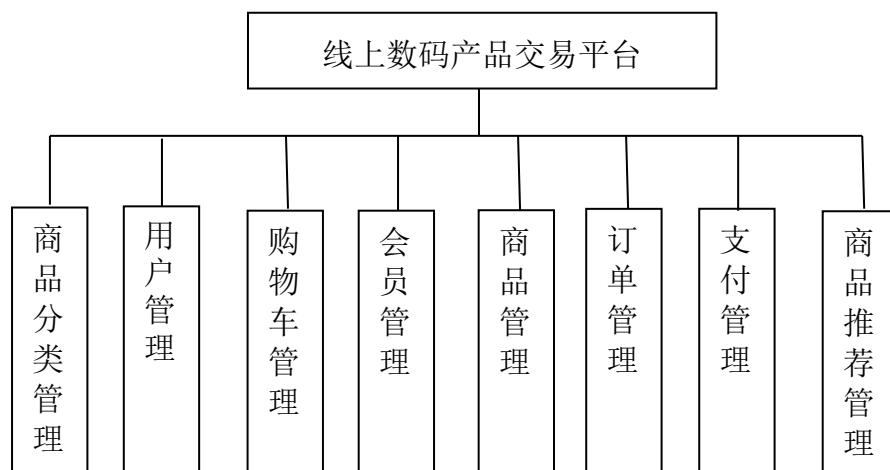


图 3.1 线上数码商品交易平台功能模块图

3.2.2 系统流程设计

当用户访问平台首页时，即可浏览商品和搜索商品。用户想购买商品的时候必须先注册登录，然后才能将商品加入购物车中进行购买，最终结算产生订单。

用户在进入首页时，需要登录才能将商品加入购物车进行购买。否则将跳转到登录界面。

用户在线上数码商品交易平台的整个购物流程如图 3.2 所示

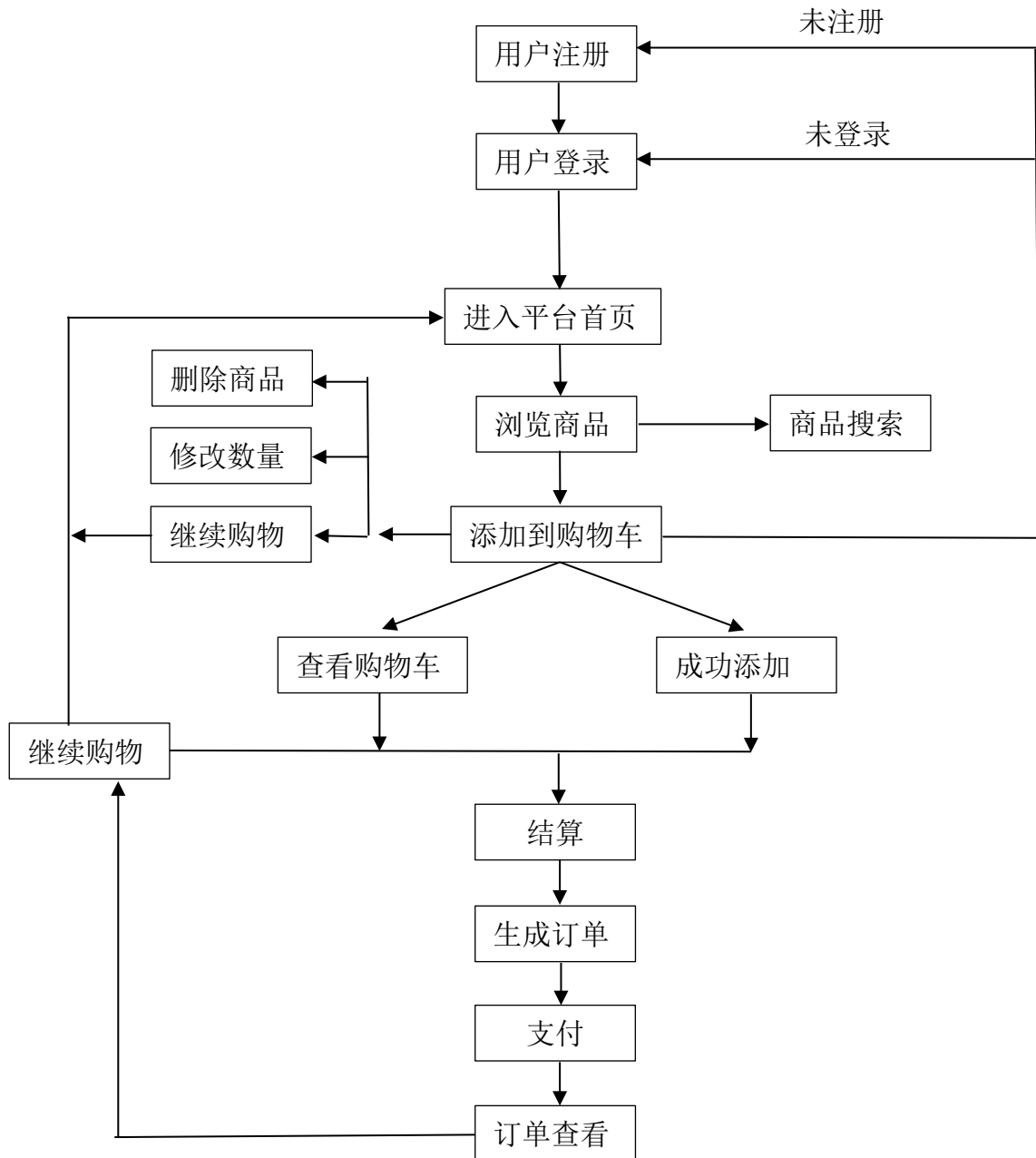


图 3.2 购物与交易流程图

管理员登录后台后，可以对商品分类、商品信息、商品推荐、用户以及订单等功能进行管理。

管理员在后台进行管理的操作流程如图 3.3 所示。

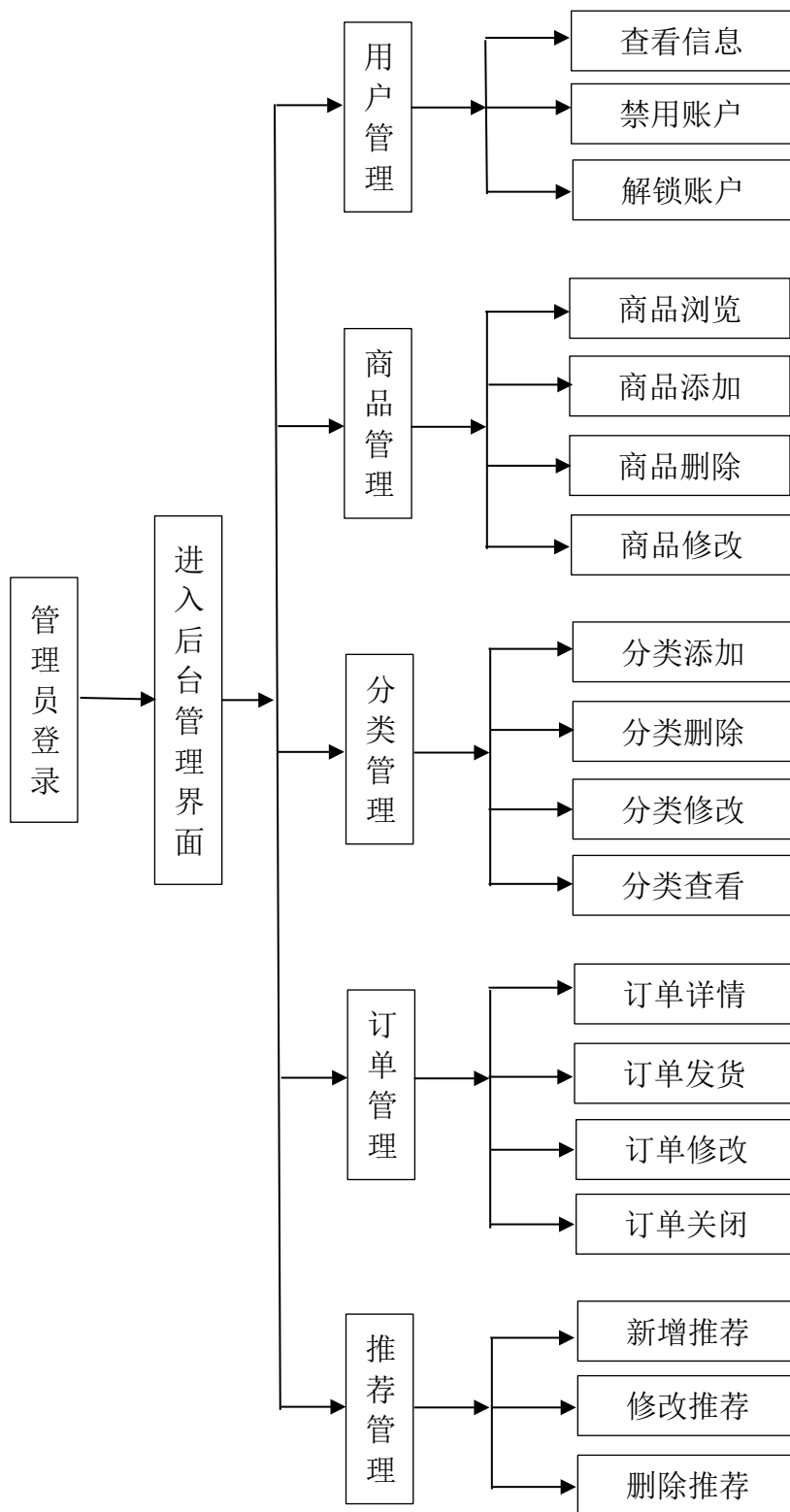


图 3.3 后台管理流程图

3.3 数据库表设计

对于大多数系统而言，数据库是整个应用系统的基石，因此数据库的表设计也就成为整个系统设计的重中之重。如果表设计的不够合理，不但会对系统性能造成影响，还会给开发和集成带来很大的麻烦，甚至留下一些潜在的隐患，最后还会引起一系列的问题，比如数据不一致等。

在进行数据库表的设计前，开发人员必须仔细地设计表结构。合适的表结构会给数据的保存提供很大的便利。这次的开发设计了 5 个数据表，分别是订单信息表、用户信息表、分类信息表、商品信息表、商品图片信息表。通过设计表结构，对表中所包含的信息进行了定义，既方便了数据的保存，又方便了数据的管理。下面详细说明了每个数据表的设计。

如表 3.1 订单信息表所示，主要储存了订单的 id、订单号、支付总金额、交易内容、支付状态、创建时间等订单信息。

表 3.1 订单信息表

名	类型	长度	null	注释
id	bigint	20		id
order_sn	char	64	√	订单号
order_id	bigint	20	√	订单 id
alipay_trade_no	varchar	50	√	交易流水号
toatal_amount	decimal	18	√	支付总金额
subject	varchar	200	√	交易内容
payment_status	varchat	20	√	支付状态
create_time	datetime		√	创建时间
confirm_time	datetime		√	确认时间
callback_content	varchar	4000	√	回调内容
callback_time	datetime		√	回调时间

如表 3.2 用户信息表所示，主要存储用户的 id，等级等详细信息。

表 3.2 用户信息表

名	类型	长度	null	注释
id	bigint	20		id
sku_id	bigint	20	√	sku_id
member_level_id	bigint	20	√	用户等级 id
member_level_name	varchar	100	√	用户等级名

如表 3.3 商品信息表所示，主要存储商品的 id、商品名、图标、所属分类等信息。

表 3.3 商品信息表

名	类型	长度	null	注释
attr_id	bigint	20		商品 id
attr_name	char	30	√	商品名
search_type	tinyint	4	√	是否需要检索
value_type	tinyint	4	√	值类型
icon	varchar	255	√	属性图标
value_select	char	255	√	可选值列表
attr_type	tinyint	4	√	属性类型
enable	bigint	20	√	启动状态
catelog_id	bigint	20	√	所属分类
show_desc	tinyint	4	√	快速展示

如表 3.4 图片信息表所示，主要存储商品的图片信息。

表 3.4 图片信息表

名	类型	长度	null	注释
id	bigint	20		id
sku_id	bigint	20	√	sku_id
img_url	varchar	255	√	图片地址
img_sort	int	11	√	排序
default_img	int	11	√	默认图

如表 3.5 分类信息表所示，主要存储分类的 id、分类名称、父分类 id、排序值、商品数量等信息。

表 3.5 分类信息表

名	类型	长度	null	注释
id	bigint	20		分类 id
name	char	50	√	分类名称
parent_cid	bigint	20	√	父分类 id
sort	int	11	√	排序
icon	char	255	√	图标地址
product_unit	char	50	√	计量单位
product_count	int	11	√	商品数量

第 4 章 系统实现

4.1 数据库连接的实现

所有的软件都必须具备与数据库交互的能力，而与数据库进行交互的技术则是其关键所在。在开发和应用中，开发人员和使用者在操作软件时，必然会连接数据库，对其进行数据的提取和处理。下面这段代码是本次开发中的一段代码，是搜索功能代码中的一部分。

```
spring:
  datasource:
    username: root
    password: root
    url: jdbc:mysql://localhost:3306/gulimall_pms
    driver-class-name: com.mysql.cj.jdbc.Driver
```

在 application.yml 中存放于数据库连接的信息

username: 数据库登陆的用户名

password: 数据库登陆的密码

url: 数据库连接的信息，localhost 是本地登录，占用 3306 端口，连接 gulimall_pms 数据库

```
@RequestMapping("/update")
public R update(@RequestBody AttrVo attr){
    attrService.updateAttr(attr);

    return R.ok();
}
```

上述是一段数据库修改处理的代码。首先建立一个 update 方法，前端发送/update 请求将会被@RequestMapping("/update")注解拦截，请求体会传递 json 参数，json 参数通过 @RequestBody 注解转换为 attr 实体类，然后在 update 方法进行数据的修改处理。获取到要修改的内容后通过 updateAttr 方法将 attr 实体类传递到 server 层进行逻辑处理。

```
@Override
public void updateAttr(AttrVo attr) {
    AttrEntity attrEntity = new AttrEntity();
    BeanUtils.copyProperties(attr,attrEntity);
```

```
this.updateById(attrEntity);  
}
```

当 server 层的 updateAttr 方法接收到 collector 层传递来的参数后，创建一个 AttrEntity 实体类，通过 BeanUtils 类中的 copyProperties 方法将 attr 中需要的数据拷贝到 attrEntity 中，在 server 层中自带的 updateById 方法，通过 attrEntity 中的 id 字段更改数据库的内容。

4.2 登录验证功能的实现

当用户登录时，首先输入账号及密码，然后按下登录按钮，这时账号和密码就会传送到判定页面，并交给判定页面进行判定。在处理时，软件系统通过连接数据库，比较用户输入的关键词与用户表中的数据，对用户输入的账户密码进行验证，并返回结果页。如图 4.1 中所示，为用户登陆页面。



图 4.1 用户登陆页面



图 4.2 管理员登陆页面

```
Dim username, password
```

```
username = Request.Form("username")
```

```
password = Request.Form(password")
```

```
if username ="" or password="" then
```

```
response.Redirect("../User/fail1.aspx")
```

上述一小段代码是从登录页面获取参数值。首先，我们用 Dim 对定义了两个变量，一个是“username”，一个是“password”，接着，我们用“Request”方法获得 Form 中的变量，它包括了从登录页面传过来的两个变量：“用户名”和“密码”。最后，判断所获取的用户名和密码是否为空，如果是空，就用 response.redirect 方法，跳转到错误提示页面。否则进行下一步的数据库确认。

4.3 购物车功能的实现

购物车功能的实现是本次开发的重点，也是目前购物平台中应用最为广泛的技术之一。本次开发的项目里“删除”、“继续购物”、“结账”三大功能是购物车功能的核心。用户可以通过购物车页面查看自己添加的商品，也可以删除、编辑商品，还可以继续购物或结账。

如图 4.3、图 4.4 所示，为购物车页面。在这里用户可以看到自己添加的所有商品。添加商品后，用户可以选择删除、编辑，或者继续购物；也可以结算。

购物车 温馨提示：产品是否购买成功，以最终下单为准哦，请尽快结算



图 4.3 空购物车页面

在商品详情页面单击“加入购物车”后,其对应商品便会加入购物车。如下图 4.4 所示。

购物车 温馨提示：产品是否购买成功，以最终下单为准哦，请尽快结算

商品名称	单价	数量	小计	操作
 Apple iPhone12 Pro (A2408) 1...	8499元	<input type="text" value="1"/>	8499元	×
 Apple AirPods 充电电盒	1246元	<input type="text" value="1"/>	1246元	×
 Apple MacBook Air 13.3	7168元	<input type="text" value="1"/>	7168元	×
 索尼 WH-1000XM3 头戴式耳机	2599元	<input type="text" value="1"/>	2599元	×
 HUAWEI Mate 30 4000万超感光徕卡影像	3999元	<input type="text" value="1"/>	3999元	×

继续购物 | 共 5 件商品

合计 (不含运费) : 23511.00元

去结算

图 4.4 添加商品后的购物车页面

点击购物车中“数量”下的数字，可对要购买商品的数量进行编辑。编辑成功后购物车下方的件数和金额合计也会发生相应的变化。而本次设计的购物车中同一商品最多可添加五件，如果超出数量限制后系统会进行相应的提示。如图 4.5 所示。

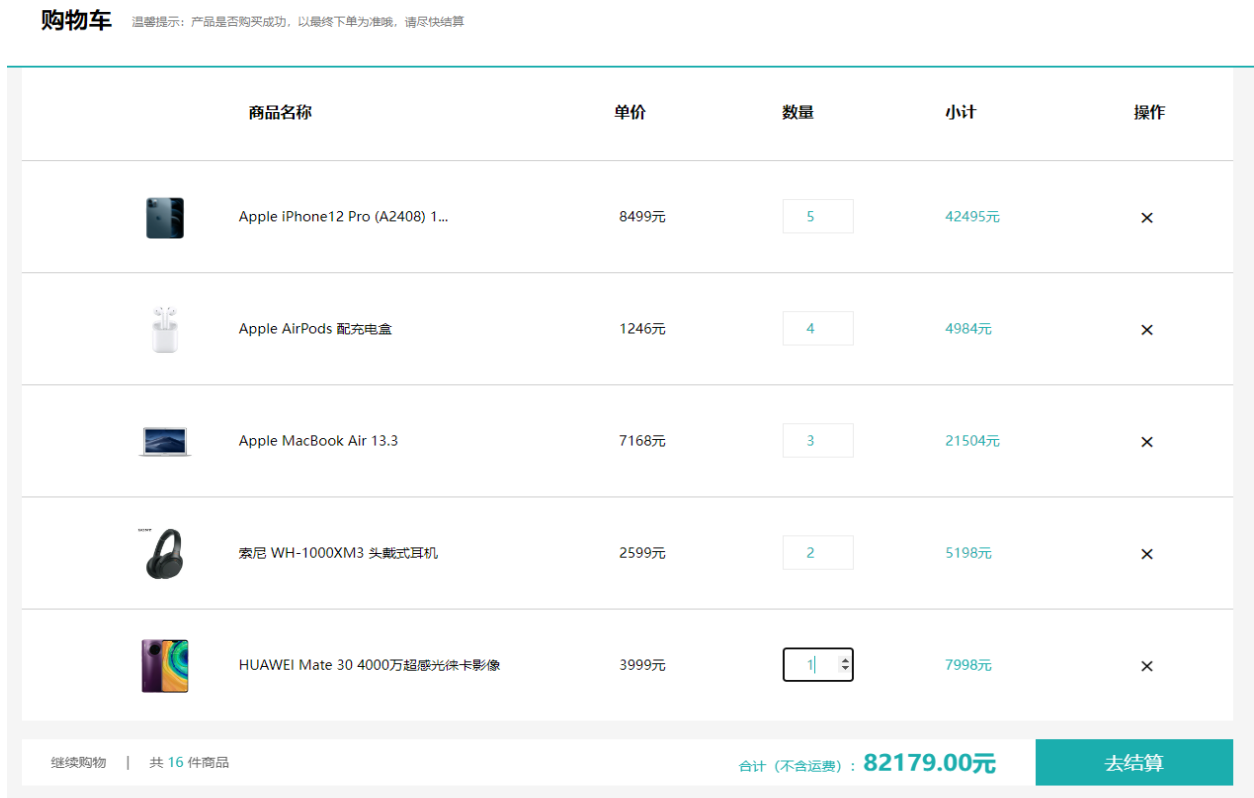


图 4.5 修改商品数量

将商品数量修改到 10 件，弹出提示框。如图 4.6 所示。



图 4.6 商品件数限制

以下代码是购物车功能中的一段核心代码。

```
private BoundHashOperations<String, Object, Object> getCartOps() {
    CurrentUser currentUser = CartInterceptor.threadLocal.get();
    String cartKey = "";
    if (currentUser.getUserId() != null) {
        cartKey = CART_PREFIX + userInfoTo.getUserId();
    } else {
        cartKey = CART_PREFIX + userInfoTo.getUserKey();
    }
    BoundHashOperations<String, Object, Object> operations =
        redisTemplate.boundHashOps(cartKey);
    return operations;
}
```

上述代码中，将购物车中的信息存放在了 redis 缓存中，使用 redis Template 通过 cartKey 绑定购物车。这里的 operations 是对 Redis 的一组操作（增删改查）的集合。

4.4 订单功能的实现

在整个系统中，订单功能起到了很大的作用，它的作用是对订单进行管理，包含了订单的生成、订单的确认、订单的支付、订单的完成、订单的取消等过程。同时，对订单状态和订单金额的计算也进行了详细的阐述。

选择购物车中的“去支付”，进入到订单创建页面。用户可核对购买的商品、价格以及收货地址是否无误。如果发现填写的收货地址错误的话还可以在此修改收货地址。

如图 4.7 所示。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/025212104212011130>