**河南科技学院**

**2023届本科毕业论文（设计）**

**题目：基于gin+vue的本地生活服务类商城**

|  |  |  |
| --- | --- | --- |
| **姓 名：** |  | **王雅晴** |
|  |  |  |
| **专 业：** |  | **数据科学与大数据技术** |
|  |  |  |
| **学 院：** |  | **信息工程学院** |
|  |  |  |
| **指导教师：** |  | **王新法（高级工程师）** |
|  |  |  |
| **完成时间：** |  | **2023年5月** |

**摘 要**

**本文介绍了一种基于Gin和Vue框架的本地生活服务类商城的设计与实现。本地生活服务类商城是一个线上购物平台，旨在为用户提供便捷的本地生活服务和商品购买体验。商城的后端使用Gin框架实现，前端使用Vue框架开发。商城实现了用户注册、登录、购物车、订单管理等功能。同时，商城还具有商品展示、商品搜索、商品分类等功能，以提高用户的购物体验。**

**商城的后端实现了RESTfulAPI，使用JWT实现了用户认证和授权。同时，商城还使用MySQL数据库存储用户信息、商品信息、订单信息等数据。商城的前端使用Vue框架开发，实现了响应式布局、SPA(单页面应用)等功能。商城前端还使用了Element-UI、Axios等第三方库，以提高开发效率和用户体验。**

**商城的测试结果表明，商城的性能稳定，具有较高的用户体验。商城的后端响应速度快，前端交互流畅，能够满足用户购物的需求。**

**本地生活服务类商城的开发实现，对于提升本地生活服务业的现代化水平，推动本地经济发展具有一定的意义。**

**关键词：购物平台，Gin, Vue**

**ABSTRACT**

**This article introduces the design and implementation of a local life service mall based on the Gin and Vue frameworks. Local Life Service Mall is an online shopping platform designed to provide users with convenient local life services and product purchase experiences. The backend of the mall is implemented using the Gin framework, while the front-end is developed using the Vue framework. The mall has implemented functions such as user registration, login, shopping cart, and order management. At the same time, the mall also has functions such as product display, product search, and product classification to improve users' shopping experience.**

**The backend of the mall has implemented RESTful API and implemented user authentication and authorization using JWT. At the same time, the mall also uses MySQL databases to store user information, product information, order information, and other data. The front-end of the mall is developed using the Vue framework, achieving functions such as responsive layout and SPA (single page application). The front-end of the mall also uses third-party libraries such as Element-UI and Axios to improve development efficiency and user experience.**

**The test results of the mall indicate that its performance is stable and has a high user experience. The backend of the mall has fast response speed and smooth front-end interaction, which can meet the shopping needs of users.**

**The development and implementation of local life service malls has certain significance for improving the modernization level of local life service industry and promoting local economic development.**

**Keywords: shopping;Gin;Vue**

**目 录**

[1 绪论 1](#_Toc25858)

[1.1 选题的背景及意义 1](#_Toc9685)

[1.2 国内外研究现状 1](#_Toc16518)

[1.2.1 国内研究现状 1](#_Toc15594)

[1.2.2 国外研究现状 1](#_Toc30345)

[1.3 课题研究内容 1](#_Toc16572)

[2 系统核心技术 3](#_Toc12957)

[2.1 Web后端框架 Gin 3](#_Toc2683)

[2.2数据存储MySQL 3](#_Toc23956)

[2.3 Web前端框架Vue 3](#_Toc27296)

[2.4 登录认证 JWT 3](#_Toc11556)

[2.5 容器化部署方式 Docker 3](#_Toc11556)

[3 系统需求分析 4](#_Toc29323)

[3.1 可行性分析 4](#_Toc23983)

[3.1.1 技术可行性 4](#_Toc939)

[3.1.2 操作可行性 4](#_Toc20269)

[3.1.3 经济可行性 4](#_Toc6457)

[3.2 用例分析 5](#_Toc18060)

[3.3 功能性需求分析 5](#_Toc10394)

[3.4 系统非功能性分析 5](#_Toc10394)

[4 系统设计 6](#_Toc24332)

[4.1 主要功能模块设计 6](#_Toc5587)

[4.2 数据库设计 6](#_Toc21332)

[4.3 数据库表设计 7](#_Toc21332)

[5 系统实现与测试 1](#_Toc25795)2

[5.1 用户功能实现 1](#_Toc17492)2

[5.1.1 用户注册登录模块 1](#_Toc21751)2

[5.1.2 商品展示模块 1](#_Toc5179)3

[5.1.3 商品搜索模块 1](#_Toc17750)5

[5.1.4 商品分类模块 1](#_Toc21751)6

[5.1.5 个人信息模块 17](#_Toc5179)

[5.2 管理员功能实现 1](#_Toc24066)9

[5.2.1 管理员登录模块 1](#_Toc26104)9

[5.2.2 管理员添加商品模块 1](#_Toc5719)9

[5.2.3 商城商品配置模块 2](#_Toc12771)0

[5.2.4 商品分类模块 2](#_Toc2295)0

[5.2.5 商品详情模块 2](#_Toc12771)0

[5.2.6 订单管理模块 2](#_Toc2295)1

[5.3 系统测试 19](#_Toc24066)

[5.3.1 测试概念 19](#_Toc26104)

[5.3.2 测试结果 22](#_Toc5719)

[5.3.3 性能测试 19](#_Toc26104)

[5.3.4 测试结论 2](#_Toc5719)4

[6 结论 26](#_Toc1923)

[参考文献 27](#_Toc1438)

[致谢 30](#_Toc26845)

1 绪论

* 1. 选题的背景及意义

在当今数字化时代，人们越来越习惯于在线购物，同时也逐渐开始关注本地生活服务。尤其是在疫情期间，人们更加重视本地生活服务，例如餐饮、快递、家政等服务。因此，本地生活服务类商城的研究具有非常重要的意义。首先，本地生活服务类商城可以提供一个方便、快捷的购物平台，让用户可以在家中轻松购买所需的服务和商品，减少了用户外出的时间和风险。其次，商家可以通过这样的平台展示自己的产品和服务，吸引更多潜在客户。商家可以通过平台实现在线支付、配送等服务，提高了商家的销售效率和用户满意度。最后，本地生活服务类商城的研究也可以为相关领域的研究提供参考和借鉴。在构建平台的过程中，需要考虑到用户体验、安全性、效率等因素，这些因素对于其他电商平台的研发也具有重要意义。因此，本地生活服务类商城的研究具有实际应用价值和理论意义，可以推动本地服务业的发展和数字化升级，也能够为电商平台的研究提供新的思路和方法[1]。

1.2 国内外研究现状

1.2.1 国内研究现状

在国内，本地生活服务类商城的研究和发展呈现出快速增长的趋势。其中，淘宝、京东等大型电商平台通过线上购物和本地服务的结合，为用户提供更加便捷的本地生活服务。同时，新兴的本地服务平台如饿了么、美团外卖等也在积极开发和运营本地生活服务类商城，为用户提供餐饮、快递、家政等服务。此外，近年来，一些新型的本地生活服务类商城开始出现。这些商城通过结合线上和线下的优势，实现更加深入的本地生活服务。例如，可以提供本地特色商品和服务，并通过线下门店、服务站等形式提供快速便捷的配送和服务[2]。总的来说，本地生活服务类商城在国内的研究和发展正处于蓬勃发展的阶段，各大平台和商家都在积极探索和实践新的商业模式和技术架构，以提高用户的体验和满意度，推动本地服务业的数字化升级和发展。

1.2.2 国外研究现状

在国外，本地生活服务类商城的研究和发展同样呈现出快速增长的趋势。一些知名的本地服务平台，如Uber Eats、Deliveroo、Postmates等，已经在全球范围内推广其本地服务类商城，并且取得了良好的市场反响。这些平台通过结合线上和线下的优势，为用户提供了包括餐饮、超市购物、药品配送等各类本地生活服务。此外，一些新兴的本地服务平台，如TaskRabbit、Handy等，也在积极探索和实践新的本地生活服务模式，如家政服务、修理服务等。Gin是一款轻量级的Go语言Web框架，具有出色的性能和高并发处理能力。而Vue是一款前端框架，可以实现响应式的UI设计，同时也具有高效的数据渲染能力。使用Gin+Vue技术栈的本地生活服务类商城具有高效、可扩展、良好的用户体验和安全性等优势，同时也具有创新的技术实现方式。这样的技术架构可以为本地生活服务行业的数字化升级提供新思路和技术支持[3]。

1.3 课题研究内容

本课题是面向大众服务的线上购物系统，本系统由后台服务器、微信小程序购物客户端两部分组成。后端开发为基于Gin框架进行后端API接口的开发，实现用户注册、登录、商家入驻、商品管理、订单管理等功能。前端开发为基于Vue框架进行前端页面的开发，实现商城的商品展示、购物车、订单结算、支付等功能，并考虑用户体验和响应速度的优化。数据库设计上包括用户信息、商家信息、商品信息、订单信息等。最后，对商城系统进行测试，包括功能测试、性能测试、安全测试等，确保系统能够稳定运行并保障用户数据安全。

2 系统核心技术

2.1 Web后端框架 Gin

Gin是一个基于Go语言的Web框架，其特点是高性能和简洁易用。Gin使用了类似于Martini的API设计，但是性能比Martini更好。Gin还提供了许多中间件，可以方便地扩展功能，例如日志记录、JWT验证、CORS等。在本地生活服务类商城的开发中，采用Gin框架作为后端框架，能够提供高效、稳定的API服务和可扩展的功能，同时能够与Vue框架的前端页面实现良好的协作和数据交互[5]。

2.2 数据存储MySQL

Mysql是一个流行的关系型数据库管理系统，具有开源、跨平台、高性能、可扩展性、支持SQL的特性，作为后端数据库系统能够提供稳定、可靠的数据存储和管理功能，同时支持复杂的SQL查询和事务处理，满足商城应用程序的各种数据管理需求。

2.3 Web前端框架Vue

Vue是一个流行的JavaScript前端框架，用于构建交互式的Web用户界面。Vue采用了组件化的开发方式，能够轻松实现数据驱动、模块化、可复用的Web应用程序,也能够提供灵活、高效的UI开发和组件化的开发方式。

2.4 登录认证JWT

JWT（JSON Web Token）是一种用于身份验证和授权的开放标准，它可以在网络应用程序之间安全地传输声明。JWT的优点在于它可以在不需要在服务器端存储会话信息的情况下，实现无状态的身份验证和授权。这使得JWT成为了现代Web应用程序中广泛使用的身份验证和授权机制。JWT的出现，为Web应用程序的身份验证和授权带来了更加安全和便捷的解决方案，成为了现代Web应用程序开发中不可或缺的一部分[4]。

2.5 容器化部署方式 Docker

使用Docker进行容器化部署的过程包括安装Docker引擎、创建Docker镜像、运行Docker容器、管理Docker容器、使用Docker Compose和使用Docker Swarm等步骤。通过使用Docker，开发人员和运维人员可以更加轻松地管理和扩展应用程序，从而提高应用程序的可靠性和可维护性[5]。

3 系统需求分析

3.1 可行性分析

### 3.1.1 技术可行性

本项目采用B/S架构，相比于C/S架构，维护和升级方式简单，成本较低，选择更多，用户只需通过浏览器进行站点访问。

系统采用市面上流行的架构，前端为Vue框架，后端为Gin 框架，使用JWT进行登录验证，使用RabbitMq 进行消息发送，大大提升消息的处理能力和消息的及时性[8]。在查询时，采用Redis进行内存存储，增加访问的快速性。本系统基于eladmin 系统，模块解耦独立，使得系统得到更好的维护[9]。

### 3.1.2 操作可行性

本系统界面简洁，功能明确，用户在使用过程中只需极短的时间，便可以了解系统的功能，对用户十分友好。app端为商品信息展示页面，用户可以通过商品信息列表，来获取自己想要的商品的信息，通过信息，了解商品介绍 。后台采用管理系统形式，管理员可以通过管理系统，对成员，角色等进行控制，当有新的商品时，管理员可以上架商品。

在操作方面，用户只需一部手机或者电脑，便可以轻松登录系统，实时查阅当前商品情况等，方便快捷。

### 3.1.3 经济可行性

本系统收益来源主要是通过平台进行商品的售卖，会获得一定的利润等。本系统开发周期短，开发成本低，符合经济可行性。

综上所述，本系统从技术上、操作上、经济上都具有很高的可行性，是可以进行实施的。

3.2 用例分析

本系统的系统用户有系统浏览者和管理员两类，它们的描述和功能如表3.1所示。

**表3.1 系统用户角色描述**

|  |  |  |
| --- | --- | --- |
| 角色名 | 角色功能 | 角色功能 |
| 系统管理员 | 系统主要参与者 | 对商品.类别、订单等进行管理 |
| 浏览者 | 系统主要参与者 | 查看商品、购买商品、地址配置 |

3.3 功能性需求分析

经过对待解决问题的综合与分析，首先，商城应该提供用户注册和登录功能，允许用户使用用户名和密码的方式进行登录和注册，方便用户进行交易和查看订单。本管理系统商品在类型管理模块要实现查询、添加、编辑和删除商品类型功能，也可对某个分类进行下级修改；在商品管理模块要实现增加、删除、编辑和查找商品功能，可以对热销商品、新品上线、为你推荐分别进行配置，也可以对商城首页轮播图进行配置，用户可以根据商品类别查找商品也可以使用关键词搜索商品。同时，商城应具有购物车管理功能，可以添加、删除、修改商品，也应具备订单管理功能，对订单添加、删除、修改、查询、结算的功能。在对商品进行支付时，也应支持支付宝和微信两种支付方式。

3.4 系统非功能需求

商城系统的设计时还应该考虑系统的非功能需求。一般来说，系统非功能性需求包括兼容性、易操作性、可扩展性和可靠性。

（1）兼容性

从使用角度来看系统可分为客户端与服务器两部分。本系统采用的B/S结构，在设计前端代码时考虑了不同浏览器兼容的样式脚本的问题，因此，本系统具有较好的兼容性，用户在不同浏览器仍能查看相同的系统页面。

（2）易操作性

在系统功能实现过程中应尽可能地简化业务操作功能，以最低的运行成本完成事件流程。考虑到多数使用者对计算机专业知识掌握不够，系统操作难度高会降低人们的使用率。本系统操作简单易上手，适合计算机知识低的人群使用。

（3）可扩展性

事情总是不停变化着，软件系统亦不例外，久而久之，人们对商城管理系统的要求也会不断变化，相应人员需要对系统的功能进行调整。开发者使用面向对象程序设计方法，严格遵循开发规范开发管理系统，以系统可扩展性。因此，该系统在开发过程中满足软件流程规范、面向对象思想的发展、善用接口编程，以便后期新增其他功能。

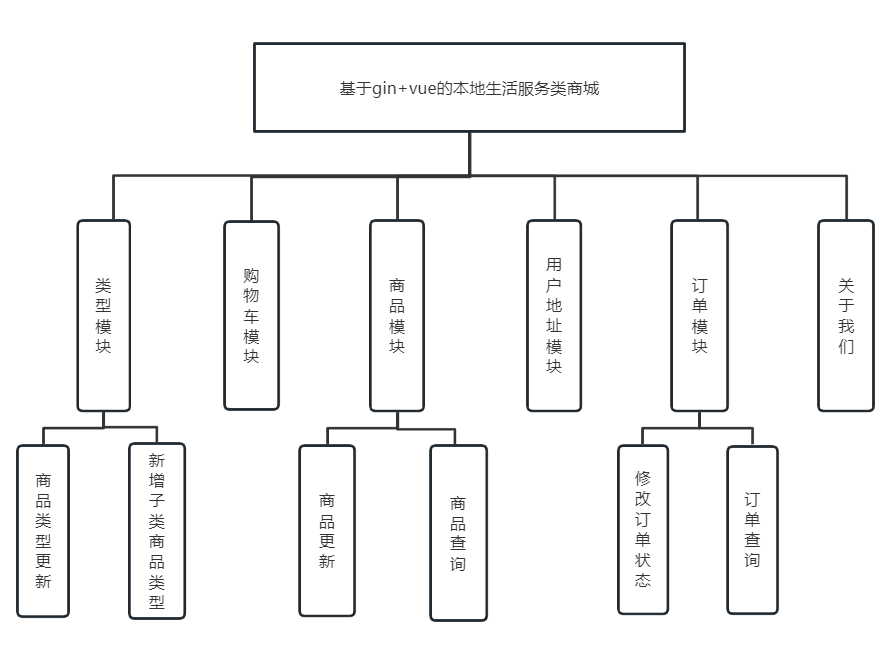
（4）可靠性

软件系统得以大规模应用，其中一个重要原因是它有很好的可靠性，如果系统内的各项数据没有可靠性，就会在实际工作中丧失管理意义，使得其作用不能完全被发挥。系统可靠度越高，使用价值越大，用户体验越好。

4 系统设计

4.1 主要功能模块设计

本系统主要分六大模块，商品类型模块包含了更新和新增子类商品类型等功能；购物车模块用来将商品添加到购物车中；商品模块包含商品更新、查询等功能；用户地址模块配置了用户地址信息；订单模块包含了对订单的查询、修改等功能；关于我们模块包含了更新公司简介的功能；统计分析模块包含了个人信息。基于gin+vue的本地生活服务类商城的功能模块图如图4-1所示。

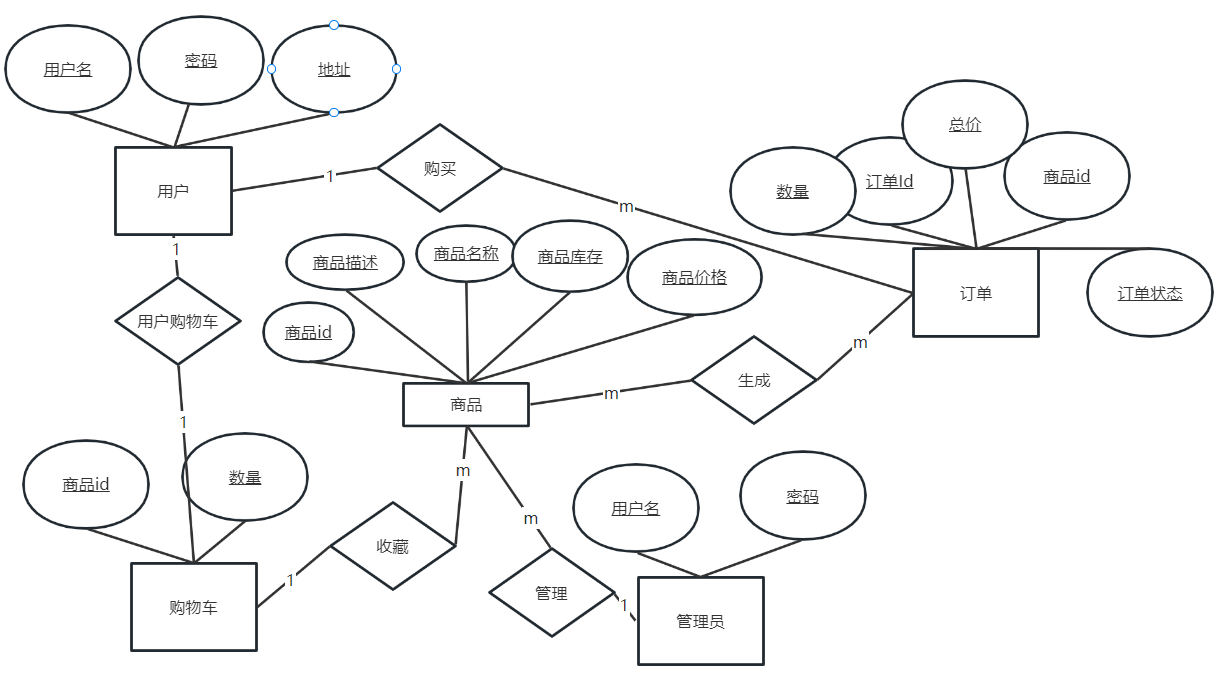


**图4-1 本地生活服务类商城功能模块图**

4.2 数据库设计

在数据库设计时，有五个实体构成，分别是商品实体、用户实体、管理员实体、购物车实体、订单实体。

其中管理员实体代表可以发布商品信息的用户，商品实体代表商品信息，购物车实体代表用户收藏的商品，订单实体代表商品被购买信息，这五个实体的E-R图4-2所示。



**图 4-2 系统E-R图**

4.3 数据库表设计

通过E-R图设计关系数据模型，本系统共设计12张数据表，如表4.1-4.12所示。

在设计数据库表的过程中,需要遵循以下原则：

（1）使用合理的数据类型及合理的数据长度，数据长度较长会影响查询效率。

（2）尽量避免在SQL语句中进行函数、计算等操作，否则将影响MySQL引擎使用索引进行查询。

**表4.1 管理员信息表 —— tb\_newbee\_mall\_admin\_user**

| 名称 | 数据类型 | 允许空值 | 长度 | 说明 |
| --- | --- | --- | --- | --- |
| admin\_user\_id | bigint | NOT NULL | 默认值 | 管理员id |
| login\_user\_name | varchar | NOT NULL | 默认值 | 管理员登录名称 |
| login\_password | varchar | NOT NULL | 20 | 管理员登录密码 |
| nick\_name | varchar |  | 11 | 管理员显示昵称 |
| locked | tinyint | NOT NULL | 10 | 是否锁定 |

**表4.2 token表 —— tb\_newbee\_mall\_admin\_user\_token**

| 名称 | 数据类型 | 允许空值 | 长度 | 说明 |
| --- | --- | --- | --- | --- |
| admin\_user\_id | bigint | NOT NULL | 0 | 主键id |
| token | varchar | NOT NULL | 32 | token值 |
| update\_time | datetime | NOT NULL | 0 | 修改时间 |
| expire\_time | datetime | NOT NULL | 0 | token过期时间 |

**表4.3 轮播图配置表 —— tb\_newbee\_mall\_carousel**

| 名称 | 数据类型 | 允许空值 | 长度 | 说明 |
| --- | --- | --- | --- | --- |
| carousel\_id | bigint | NOT NULL |  | 首页轮播图主键id |
| carsousel\_url | varchar | NOT NULL | 255 | 轮播图 |
| redirect\_url | varchar | NOT NULL | 1024 | 点击后的跳转链接 |
| carousel\_rank | varchar | NOT NULL | 255 | 排序值 |
| create\_by | bigint | NOT NULL |  | 创建人id |
| update\_by | bigint | NOT NULL |  | 更新人id |
| create\_time | datetime |  |  | 创建时间 |
| update\_time | datetime |  |  | 更新时间 |
| delete\_time | datetime |  |  | 为空代表未被删除 |
| is\_deleted | bit | NOT NULL | 1 | 状态：1启用（默认）、0禁用 |

**表4.4 商品类别表 —— tb\_newbee\_mall\_goods\_category**

| 名称 | 数据类型 | 允许空值 | 长度 | 说明 |
| --- | --- | --- | --- | --- |
| category\_id | bigint | NOT NULL |  | 分类id |
| category\_level | varchar | NOT NULL | 255 | 分类级别 |
| parent\_id | varchar | NOT NULL | 1024 | 父分类id |
| category\_name | varchar | NOT NULL | 255 | 分类名称 |
| category\_rank | bigint | NOT NULL |  | 排序值 |
| update\_by | bigint | NOT NULL |  | 更新人 |
| create\_time | datetime |  |  | 创建时间 |
| update\_time | datetime |  |  | 更新时间 |
| delete\_time | datetime |  |  | 为空代表未被删除 |

**表4.5 商品详情表 —— tb\_newbee\_mall\_goods\_info**

| 名称 | 数据类型 | 允许空值 | 长度 | 说明 |
| --- | --- | --- | --- | --- |
| goods\_id | bigint | NOT NULL |  | 商品主键id |
| goods\_name | varchar | NOT NULL | 255 | 商品名 |
| goods\_intro | varchar | NOT NULL | 1024 | 商品简介 |
| goods\_category\_id | varchar | NOT NULL | 255 | 关联分类id |
| goods\_cover\_img | bigint | NOT NULL |  | 商品主图 |
| goods\_carousel | bigint | NOT NULL |  | 商品轮播图 |
| goods\_detail\_content | varchar |  |  | 商品详情 |
| original\_price | bigint |  |  | 商品价格 |
| stock\_num | bigint |  |  | 商品库存数量 |
| tag | varchar |  |  | 商品标签 |
| goods\_sell\_status | bit |  |  | 商品上架状态 |
| create\_user | varchar |  |  | 添加者 |
| update\_user | varchar |  |  | 更新者 |
| create\_time | datetime |  |  | 创建时间 |
| update\_time | datetime |  |  | 更新时间 |
| delete\_time | datetime |  |  | 为空代表未被删除 |
| is\_deleted | bit | NOT NULL | 1 | 状态：1启用（默认）、0禁用 |

**表4.6 商品配置表 —— tb\_newbee\_mall\_index\_config**

| 名称 | 数据类型 | 允许空值 | 长度 | 说明 |
| --- | --- | --- | --- | --- |
| config\_id | bigint | NOT NULL |  | 商品主键id |
| config\_name | varchar | NOT NULL | 255 | 商品名 |
| config\_type | varchar | NOT NULL | 1024 | 商品简介 |
| goods\_id | varchar | NOT NULL | 255 | 关联分类id |
| redirect\_url | bigint | NOT NULL |  | 商品主图 |
| config\_rank | bigint | NOT NULL |  | 商品轮播图 |
| create\_user | varchar |  |  | 添加者 |
| update\_user | varchar |  |  | 更新者 |
| create\_time | datetime |  |  | 创建时间 |
| update\_time | datetime |  |  | 更新时间 |
| delete\_time | datetime |  |  | 为空代表未被删除 |
| is\_deleted | bit | NOT NULL | 1 | 状态：1启用（默认）、0禁用 |

**表4.7 商品订单表 —— tb\_newbee\_mall\_order**

| 名称 | 数据类型 | 允许空值 | 长度 | 说明 |
| --- | --- | --- | --- | --- |
| order\_id | bigint | NOT NULL |  | 订单表主键id |
| order\_no | varchar | NOT NULL | 255 | 订单号 |
| user\_id | varchar | NOT NULL | 1024 | 用户主键id |

**续表4.7**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| total\_price | varchar | NOT NULL | 255 | 订单总价 |
| pay\_time | datetime |  |  | 支付时间 |
| order\_status | bit |  |  | 订单状态：0.待支付1.已支付2.配货完成3.出库成功4.交付完成 |
| create\_time | datetime |  |  | 创建时间 |
| update\_time | datetime |  |  | 更新时间 |
| extra\_info | varchar |  |  | 订单body |
| is\_deleted | bit | NOT NULL | 1 | 状态：1启用（默认）、0禁用 |

**表4.8 收获地址表 —— tb\_newbee\_mall\_order\_address**

| 名称 | 数据类型 | 允许空值 | 长度 | 说明 |
| --- | --- | --- | --- | --- |
| order\_id | bigint | NOT NULL |  | 分类id |
| user\_name | varchar | NOT NULL | 255 | 收货人姓名 |
| user\_phone | varchar | NOT NULL | 1024 | 收货人手机号 |
| province\_name | varchar | NOT NULL | 255 | 省 |
| city\_name | bigint | NOT NULL |  | 城 |
| region\_name | bigint | NOT NULL |  | 区 |
| detail\_address | varchar |  |  | 收件详细地址 |
| update\_time | datetime |  |  | 更新时间 |
| delete\_time | datetime |  |  | 为空代表未被删除 |

**表4.9 商品订单详情表 —— tb\_newbee\_mall\_order\_item**

| 名称 | 数据类型 | 允许空值 | 长度 | 说明 |
| --- | --- | --- | --- | --- |
| order\_item\_id | bigint | NOT NULL |  | 订单关联购物项主键id |
| order\_id | varchar | NOT NULL | 255 | 订单主键id |
| goods\_id | varchar | NOT NULL | 1024 | 关联商品id |
| goods\_name | varchar | NOT NULL | 255 | 下单时商品的名称（订单快照） |
| goods\_cover\_img | bigint | NOT NULL |  | 下单时商品的主图（订单快照） |
| selling\_price | bigint | NOT NULL |  | 下单时商品的价格 |
| goods\_count | varchar |  |  | 数量（订单快照） |
| update\_time | datetime |  |  | 更新时间 |
| delete\_time | datetime |  |  | 为空代表未被删除 |

**表4.10 商品购物车表 —— tb\_newbee\_mall\_shopping\_cart\_item**

| 名称 | 数据类型 | 允许空值 | 长度 | 说明 |
| --- | --- | --- | --- | --- |
| cart\_item\_id | bigint | NOT NULL |  | 订单关联购物项主键id |
| user\_id | varchar | NOT NULL | 255 | 订单主键id |
| goods\_id | varchar | NOT NULL | 1024 | 关联商品id |
| goods\_count | varchar | NOT NULL | 255 | 下单时商品的名称（订单快照） |
| update\_time | datetime |  |  | 更新时间 |
| delete\_time | datetime |  |  | 为空代表未被删除 |
| is\_deleted | bit | NOT NULL | 1 | 状态：1启用（默认）、0禁用 |

**表4.11 用户表 —— tb\_newbee\_mall\_user**

| 名称 | 数据类型 | 允许空值 | 长度 | 说明 |
| --- | --- | --- | --- | --- |
| user\_id | varchar | NOT NULL | 255 | 用户主键id |
| nick\_name | varchar | NOT NULL | 1024 | 用户昵称 |
| login\_name | varchar | NOT NULL | 255 | 登录名称 |
| password\_md5 | varchar |  |  | Md5加密后的密码 |
| introduce\_sign | varchar |  |  | 个性签名 |
| is\_deleted | bit | NOT NULL | 1 | 状态：1启用（默认）、0禁用 |
| locked\_flag | bit |  |  | 注销标识字段 |
| create\_time | datetime |  |  | 注册时间 |

**表4.12 用户地址表 —— tb\_newbee\_mall\_user\_address**

| 名称 | 数据类型 | 允许空值 | 长度 | 说明 |
| --- | --- | --- | --- | --- |
| address\_id | bigint | NOT NULL |  | 订单关联购物项主键id |
| user\_id | varchar | NOT NULL | 255 | 用户主键id |
| user\_name | varchar | NOT NULL | 1024 | 收货人姓名 |
| user\_phone | varchar | NOT NULL | 255 | 收货人手机号 |
| default\_flag | bigint | NOT NULL |  | 是否为默认0-非默认 1-是默认 |
| province\_name | varchar | NOT NULL |  | 省 |
| city\_name | varchar |  |  | 城 |
| region\_name | varchar |  |  | 区 |
| detail\_address | varchar |  |  | 收货详细地址 |
| is\_deleted | bit | NOT NULL | 1 | 状态：1启用（默认）、0禁用 |

5 系统实现与测试

5.1 用户功能实现

### 5.1.1 用户注册登录模块

用户端注册，输入用户名、密码、验证码进行注册，同时需要进行账号验证和安全认证，登录时通过用户名和密码进行匹配校验。

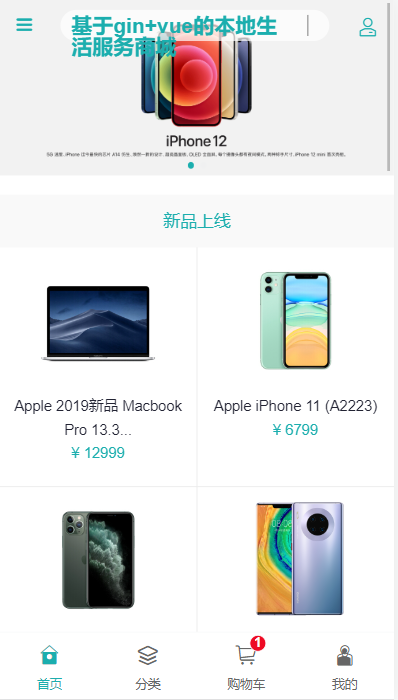
用户端注册页面如5-1所示。



**图5-1 注册页面**

**5.1.2 商品展示模块**

在登录后，跳到商城首页，首页展示商品的信息。首先是商品轮播图展示如图5-2所示。



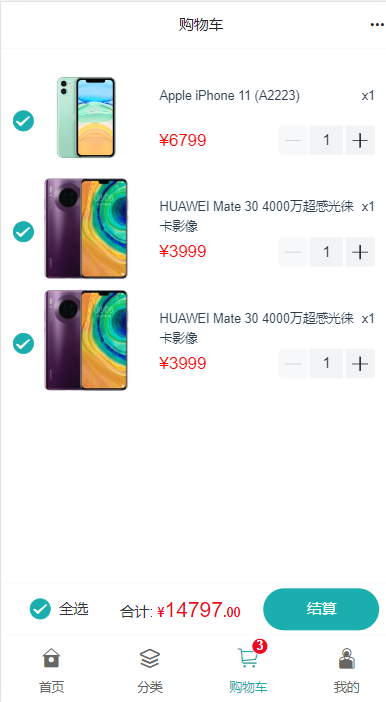
**图5-2 商品轮播图**

之后首页展示商品依次为热门商品，新品上线，最新推荐。点击具体商品，可进入商品详情页，可看到商品详情信息，如图5-3。



**图 5-3 商品详情**

点击加入购物车，会显示添加成功，同时修改购物车数量，并在购物车菜单下显示商品信息详情，如图5-4所示。之后可点击结算跳转至生成订单模块，根据选择的商品内容计算商品总价，结算时需要选择address即商品地址，默认使用个人地址中心默认地址。最后需要选择支付方式为支付宝支付还是微信支付，如图5-5所示。



**图 5-4 购物车菜单详情页**



**图 5-5 支付方式**

**5.1.3 商品搜索模块**

在搜索框中输入关键词，可根据关键词匹配搜索展示相关商品。如图5-6所示。也可以根据默认推荐，是否为最新商品以及商品价格进行排序。



**图 5-6 商品搜索**

**5.1.4 商品分类模块**

在分类菜单栏下可以查找到所有商品的分类信息，并根据当前大类还可展示所属该大类的商品小类别。如图5-7所示。



**图 5-7 商品分类**

点击进入某个具体分类后，可以查找到该分类下所有的商品。如图5-8所示。



**图 5-8 具体分类下所有商品**

**5.1.5 个人信息模块**

点击菜单栏我的，可进入个人信息模块，包含个人基本信息，昵称、登录名、个性签名如图5-9所示。



**图5-9 我的信息**

我的订单信息，全部、待付款、待确认、待发货、已发货、交易完成。如图5-10所示。



**图 5-10 我的订单**

地址管理，可以新增地址，包含姓名、电话、地区、详细地址、是否设为默认收货地址。如图5-11。

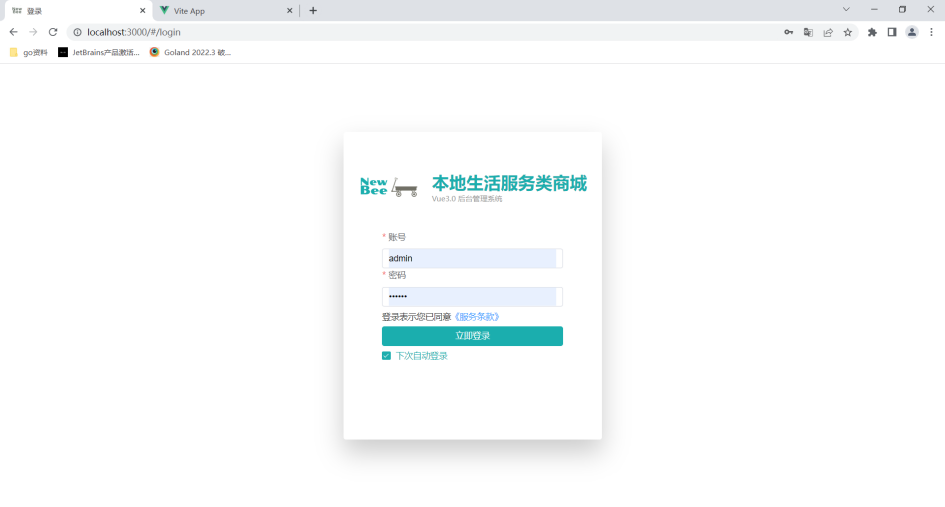


**图 5-11 新增地址**

5.2 管理员功能实现

### 5.2.1 管理员登录模块

管理员通过账号和密码进行登录。在输入正确验证码后，进行登录。如图5-12所示。



**图 5-12 管理员登录**

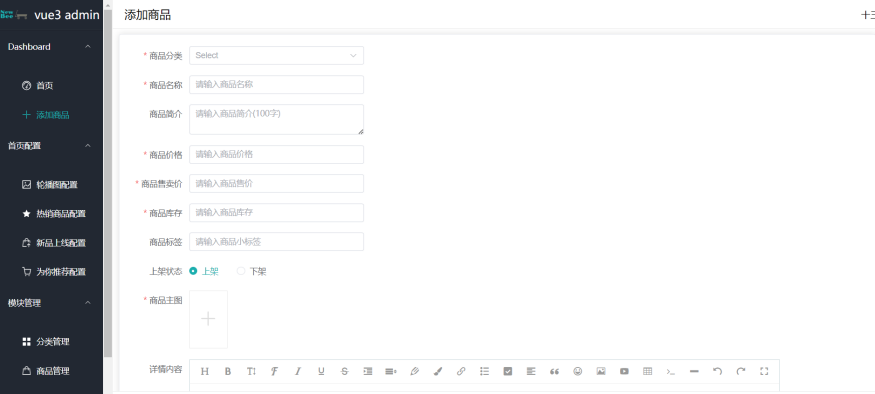
用户输入账号和密码，并且填写正确的验证码，验证码每隔60秒过期，需要刷新验证码。

管理员登录实现原理：

在管理员登录页面，输入username和password，前端请求/api/auth/login，调用AuthorizationController下的login方法，通过Rsa对账号密码进行解密，redis查询验证码，通过userService对象通过username获取用户信息，通过JwtUserDto对象生成令牌，利用foreach遍历用户角色roles，判断是否有用户权限，如果有，验证是否设置为单点登录，如果是，那么踢掉之前的token。

### 5.2.2 管理员添加商品模块

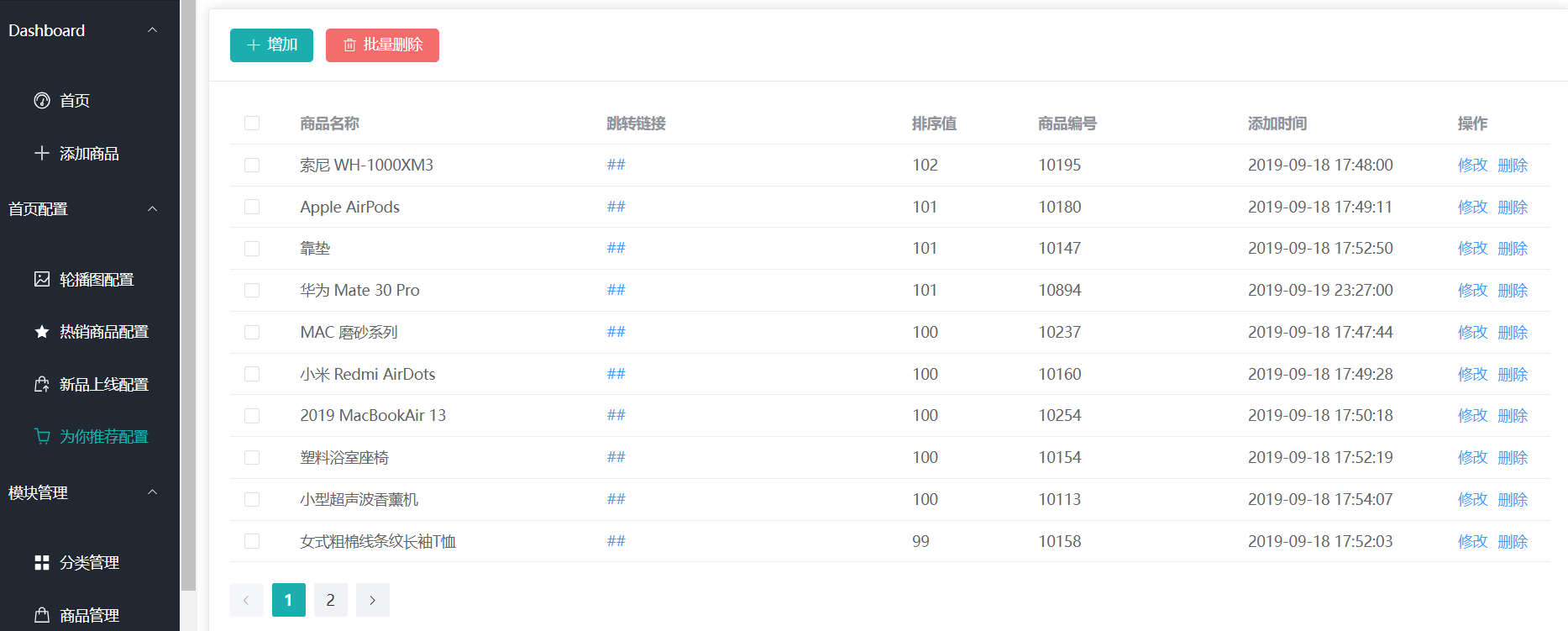
管理员添加商品，包含商品分类（商品分类根据配置树状显示大类及其下的小类）、商品名称、商品简介、商品价格、商品受卖价、商品库存、商品标签、上架状态、商品主图、详情内容。如图5-13所示。



**图5-13 添加商品**

### 5.2.3 商城商品配置模块

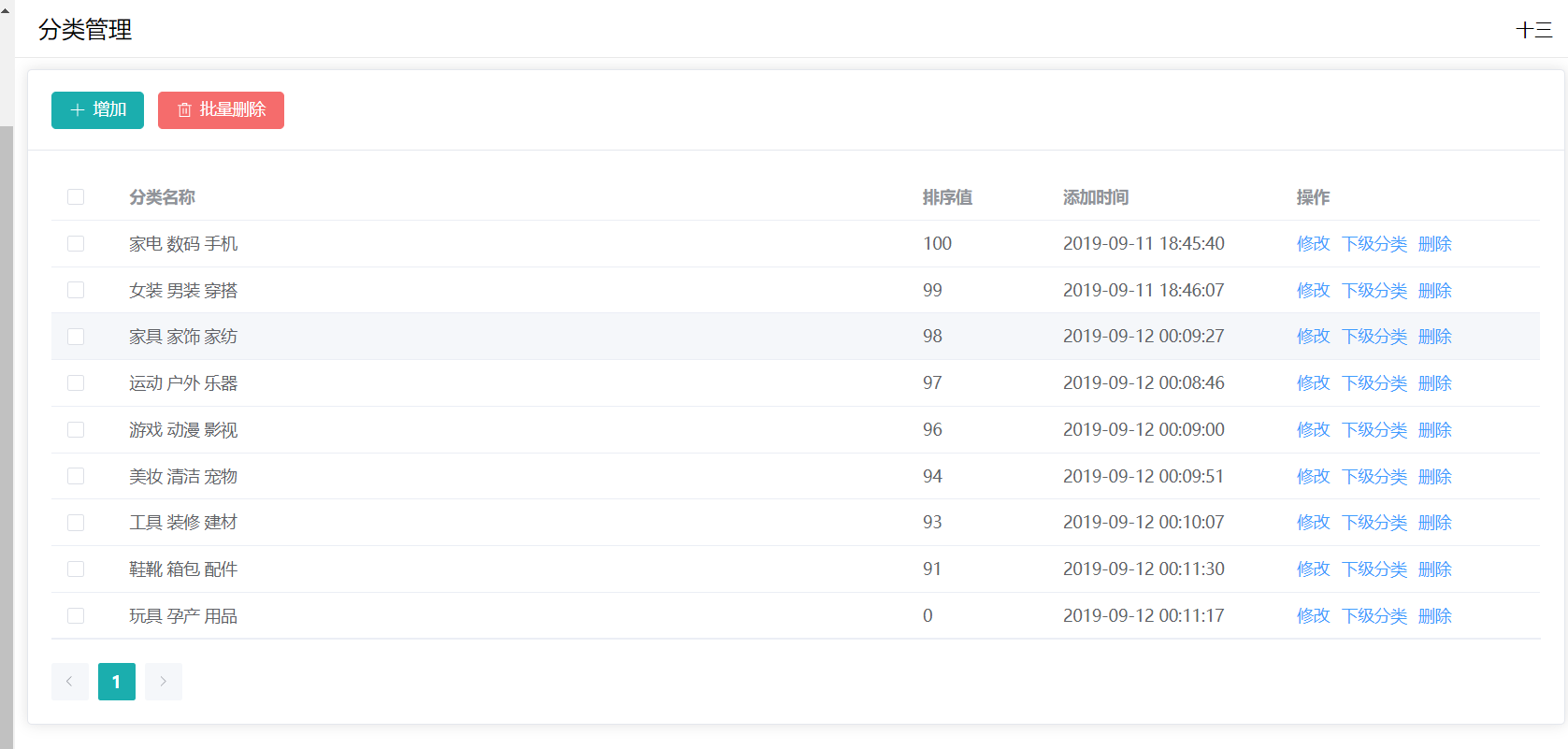
主要分为四大部分，轮播图配置，热销商品配置，新品上线配置，为你推荐配置。可以增加商品，对某条商品信息进行修改，删除商品或者批量删除商品，如图5-14所示。



**图5-14商城商品配置模块**

### 5.2.4 商品分类模块

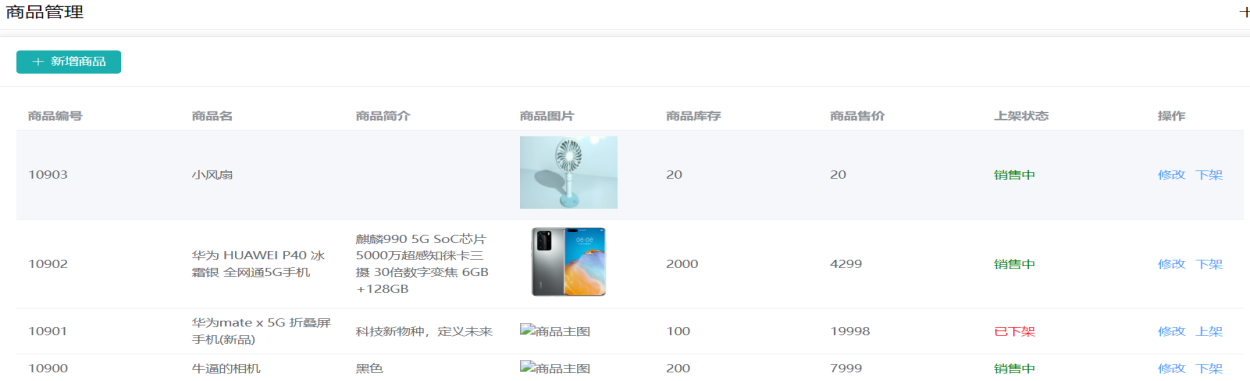
包含分类名称，排序值，添加时间，修改分类操作，为当前分类添加下级分类，单条删除分类操作，添加分类以及批量删除分类，如图5-15所示。



**图 5-15商品分类模块**

### 5.2.5 商品详情模块

管理员可以新增商品，主要用来展示商品编号，商品名，商品简介，商品图片，商品库存，商品售价，上架状态，修改商品，控制商品下架。默认每页10条，可以切换商品展示页码，如图5-16所示。



**图5-16 商品详情**

### 5.2.6 订单管理模块

管理员可以通过订单号查询订单详情，包含订单总价，订单状态，支付方式，创建时间等，也可以根据订单状态来筛选，包含待支付，已支付，配货完成，出库完成，交易成功，手动关闭，超时关闭和商家关闭，可以点击配货完成按钮、出库按钮或者关闭订单按钮来调整订单状态，如图5-17所示。



**图 5-17 订单管理**

5.3 系统测试

5.3.1 测试概念

在系统生命周期中，测试的地位十分重要。系统测试的目的是通过对新闻发布管理系统后端的不同模块的功能进行测试，尽可能地发现系统在功能上的不足，然后再将所发现的不足并修补，保障系统处于安全的、可靠的状态。系统软件测试指通过自己设计的测试用例验证系统程序操作是否正确，并把操作的结果和期望的结果相比较，如果期望的结果和操作的结果相符就会测试成功，否则此功能的测试失败。通常情况下，当检测到某个功能模块存在某一软件缺陷时，则此模块可能会出现更多缺陷。软件测试时，软件缺陷检测得越早，修复得越易，花费的代价越低。

软件测试是检验软件系统质量的重要手段，在不同环境下，软件系统的测试方法也有所差异。在软件测试过程中，同样要有一个确定的测试对象，这是因为一切测试方案、测试过程都围绕着测试对象进行。

5.3.2测试结果

本测试所有的输入都保证符合用户的使用场景，输出数据要符合系统的预期。

（1）用户模块有登录的功能，操作时需正确显示操作结果，性能要求是能够快速判断用户所填的各种值，并做相应的操作。

测试用例如表5-1所示。

**表5-1 用户登录测试用例表**

| 测试目标 | 操作步骤 | 测试用例 | 预期结果 | 执行结果 |
| --- | --- | --- | --- | --- |
| 用户登录 | （1）进入登录页面  （2）输入手机号，输入密码，获取验证码 | loginName=15090386227  passwordMd5=e10adc394ba59abbe56e057f20f883ec3949ba59abbe56e057f20f883ee10adc3949ba59abbe56e057f20f883e | 判断数据完整性，通过查询缓存是否有，如果有那么校验密码；如果没有查询数据库，数据库进行验证是否存在该用户，如果有校验密码，如果没有那么提示操作失败，给予相关提示 | 查询成功并进入首页 |

（2）用户商品首页是针对商品轮播图，热门商品，新品上线，最新推荐。可以对感兴趣的商品进行关键词搜索，浏览商品时进入某商品后还可以浏览详情，加入购物车，直接购买等。操作时需正确显示操作结果，性能要求是能够快速判断用户所填的各种值，并做相应的操作。

测试用例如表5-2所示。

**表5-2 商品模块测试用例表**

| 测试目标 | 操作步骤 | 测试用例 | 预期结果 | 执行结果 |
| --- | --- | --- | --- | --- |
| 商品首页展示 | 进入系统首页，展示商品 | 无测试用例 | 分为商品轮播图，热门商品，新品上线，最新推荐 | 包含的商品信息 |
| 商品关键词搜索 | 在搜索框页顶部有三个选项，为推荐、新品、价格，通过按钮筛选响应商品 | pageNumber:1  Keyword:%E5%8D%8E%E4%B8%BA  orderBy:price   1. Page | 根据对应的筛选项对商品进行筛选 | 查询到相应的商品列表 |
| 商品分类 | 点击分类菜单栏，选择具体分类标签 | pageNumber:1  goodsCategoryId:30  keyword:  orderBy:price | 根据对应的分类标签对商品进行筛选 | 查询到相应的商品列表 |

续表5-2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 商品加入购物车 | 在商品详情点击加入购物车按钮 | goodsCount:1  goodsId:10895 | 对相应商品进行收藏，并且在购物车列表有相应的收藏商品 | 数据表tb\_newbee\_mall\_shopping\_cart\_item成功添加该购物车记录 |
| 商品生成订单 | 点击结算按钮 | addressId:1  cartItemIds:[7,8] | 对商品进行结算，更改订单信息为已支付状态 | 购买成功 |
| 展示购物车列表 | 进入购物车菜单页面，展示当前用户对应的收藏商品 | 无测试用例 | 分页展示收藏车辆 | 查询成功对应收藏的商品 |

（3）用户个人信息

测试用例如表5-3所示。

**表5-3 用户个人信息测试用例表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试目标 | 操作步骤 | 测试用例 | 预期结果 | 执行结果 |
| 用户信息展示 | 进入个人信息菜单栏 | 无测试用例 | 返回昵称，登录名，个性签名 | 展示昵称，登录名，个性签名 |
| 个人订单 | 在个人信息页点击我的订单 | pageNumber:1  status:0   1. sPage | 根据对应的筛选项对订单进行筛选 | 查询到相应的订单列表 |
| 账号管理 | 在个人信息页点击账号管理 | 无测试用例 | 更新账号信息 | 对更改后账号信息保存即可更改 |
| 地址管理 | 在个人信息页点击地址管理 | 无测试用例 | 更改个人地址信息 | 数据表tb\_newbee\_mall\_order\_address成功添加新的地址记录 |

5.3.3 性能测试

经过性能测试，本系统表现出了良好的性能表现。在100个并发用户下，商城的响应时间平均为2秒，最大值为4秒，符合预期要求。同时，商城的最大承载量达到了5000左右，可以满足大部分用户的购物需求。在长时间运行测试中，商城没有出现任何故障，稳定性得到了保证。

针对测试结果，我们提出了一些优化建议。首先是对数据库进行优化，采用分表存储的方式减少单个表的数据量，定期清理无用数据避免数据堆积导致性能下降。其次是缓存优化，对于经常被访问的数据和计算复杂的数据进行缓存，减少数据库的访问次数。最后是代码优化，避免使用不必要的全局变量，减少内存占用。

通过以上优化措施，我们该系统的性能将会得到进一步提升，为用户提供更加优质的购物体验。

6 结论

本文设计并实现了一套商城管理系统，该系统具有完善的功能和良好的用户体验。通过对市场上已有的商城系统进行分析和比较，本文提出了一些改进和优化的方案，如增加商品推荐功能、优化购物车结算流程、提高系统安全性等。在系统实现过程中，本文采用了前后端分离的架构，使用了gin、Vue.js等技术，实现了系统的基本功能和用户界面。同时，本文还对系统进行了性能测试和安全性测试，结果表明系统具有较高的性能和安全性。在未来，可以进一步完善系统的功能和优化用户体验，提高系统的可用性和稳定性。总之，本文设计的商城管理系统具有一定的实用价值和推广价值，可以为商城的管理和运营提供一定的帮助和支持。

**参考文献**

1. 朱海洋.信息化背景下财务网上商城政府采购建设[N].财会信报,2022-12-22(007).
2. 兰志彬, 基于手机移动互联网+综合商城云平台开发及应用创新. 福建省,福建省我答答信息科技有限公司,2017-05-26.环小峰.我国商城市场的现状与前景分析[J],2021,(06):171-172.
3. 齐洋,原变青,刘颖,杨婷.基于Gin和Vue.js的作业管理系统的设计[J].信息技术与信息化,2022(10):103-105+110.Joydip Kanjilal. How to work with Redis Cache in .Net[J]. InfoWorld.com, 2017,38(2):16-35.
4. 童敏,张黎娜,梁伍七.基于JWT的分布式系统认证授权机制设计和实现[J].合肥师范学院学报,2022,40(03):7-10.
5. 孙景玉,孙浩,高婷玉,秦雯波,陈虹云.基于Redis的高并发秒杀系统,2022,23(12):45-47.DOI:10.16009/j.cnki.cn13-1295/tq.2022.12.02.
6. 谷瑞军.基于Docker容器的轻量级大数据实验平台构建研究[J].金陵科技学院学报,2019(04):1-5.
7. 杨仕昊, 叶青, 吕明,等.物联网访客数据汇聚平台的设计与实现[J]. 工业控制计算机, 2020(2):121-123.
8. 奚科芳. 数据库查询语句优化方法初探[J]. 数字技术与应用, 2016(11):247-248.
9. 赵琳.基于CentOS 7系统实现Nginx负载均衡[J].网络安全和信息化,2022,(02):96-99.
10. 王杰.Redis数据库安全研究与实践[J].电脑知识与技术,2022,18(30):60-62.DOI:10.14004/j.cnki.ckt.2022.1962.
11. 石怡.基于MySQL数据库的查询性能优化研究[J].四川职业技术学院学报,2021,31(01):164-168.
12. 范开勇;陈宇收.MySQL数据库性能优化研究[J].中国新通信,2019,(01):57.
13. 童敏,张黎娜,梁伍七.基于JWT的分布式系统认证授权机制设计和实现[J].合肥师范学院学报,2022,40(03):7-10.
14. 王巍;刘梦荞;夏明昕;马福坛;王方旭.基于HTTP的内存缓存服务的设计与实现[J].电子技术与软件工程,2022,(03):211-214.
15. 孙景玉,孙浩,高婷玉,秦雯波,陈虹云.基于Redis的高并发秒杀系统,2022,23(12):45-47.DOI:10.16009/j.cnki.cn13-1295/tq.2022.12.02.
16. 程萍.基于Vue.js的科研管理系统实现策略研究[J].电脑知识与技术,2023,19(06):31-32+50.DOI:10.14004/j.cnki.ckt.2023.0311.
17. Guyard Angeline,Martinez Jérémy,Audemar Simona. Le point de vue des rééducateurs sur les chirurgies fonctionnelles et conservatrices[J]. Hand Surgery and Rehabilitation,2022,41(6).
18. Atlantic Terminal Mall, Mobilitie Bring 5G to Brooklyn[J]. Telecomworldwire,2020.
19. Archaeology and Anthropology; Reports from Stanford University Add New Data to Findings in Archaeology and Anthropology (Multivariate Analysis of Cooking Pots From Elite and "commoner" Contexts At Yanshi Shangcheng: Implications for the Politics of Daily Life and ...)[J]. Journal of Engineering,2020.
20. Wenyan Zhao,Weiqi Wang,Lina Ma. Research on Teaching and Practice of Graduation Design from the Perspective of "Internet Plus" with "Intangible Cultural Heritage" Art as the Core[J]. Frontiers in Educational Research,2022,5(21).
21. Du Weijuan,Yang Nan. The Construction of "Internet Plus Project" Mixed Teaching Mode in the Course of Engineering Applied Mathematics[J]. Curriculum and Teaching Methodology,2022,5(5).
22. Valerii Tkachenko,Svetlana Lukianiuk. Analysis of the use of the Redis in the distributed order processing system in the restaurant network[J]. Technology audit and production reserves,2021,5(2).
23. Redis Labs Selected as a Challenger in Gartner Magic Quadrant for Cloud Database Management Systems[J]. Manufacturing Close - Up,2020.
24. Williams Cody,Stander Maria Aletta,Medvedovici Andrei,Buica Astrid. Volatile terpenoid profiling in gin and beer – A targeted approach[J]. Journal of Food Composition and Analysis,2023,118.
25. Subramani Vishal Bellie,Shi Changrong,Moghaddam Lalehvash,Atanda Luqman,Ramirez Jerome,del Río José C.,Rencoret Jorge,Bartley John,Doherty William O.S.. Cotton farming sustainability: Formation of trans-isoeugenol/ bio-aromatics, 5-chloromethylfurfural, C13–C17 liquid hydrocarbons & fertilizer from cotton gin trash[J]. Journal of Cleaner Production,2022,363.
26. Fornari Federico,Cavalli Alessandro,Cesini Daniele,Falabella Antonio,Fattibene Enrico,Morganti Lucia,Prosperini Andrea,Sapunenko Vladimir. Distributed file systems performance tests on Kubernetes/Docker clusters[J]. Journal of Physics: Conference Series,2023,2438(1).
27. You Li,Sun Hui. Research and Design of Docker Technology Based Authority Management System[J]. Computational Intelligence and Neuroscience,2022,2022.
28. Møller Pål. The Prospective Lynch Syndrome Database: background, design, main results and complete MySQL code[J]. Hereditary Cancer in Clinical Practice,2022,20(1).
29. Witsarut Sriratana,Vittaya Khagwian,Sutham Satthamsakul. Analysis of Electric Current by Using MySQL Database on Web Server for Machine Performance Evaluation: A Case Study of Air Conditioning System[J],2020.
30. I K G Sudiartha,Sudiartha I K G,Indrayana I N E,Suasnawa I W,Asri S A,Sunu Putu Wijaya. Data Structure Comparison Between MySql Relational Database and Firebase Database NoSql on Mobile Based Tourist Tracking Application[J]. Journal of Physics: Conference Series,2020,1569(3).
31. Xiamen Wangsu Co. Ltd.; "Method For Batch Processing Nginx Network Isolation Spaces And Nginx Server" in Patent Application Approval Process (USPTO 20200285506)[J]. Politics & Government Week,2020.
32. Bo She,Qiang Wang,Xiaoge Zhong,Zhe Zhang,Zunying Qin,Guodong Li. The Design and Implementation of Campus Network Streaming Media Live Video On-Demand System Based on Nginx and FFmpeg[A]. Hubei Zhongke Institute of Geology and Environment Technology.Proceedings of the 2nd International Conference on Artificial Intelligence and Computer Science (AICS 2020)[C].Hubei Zhongke Institute of Geology and Environment Technology:湖北省众科地质与环境技术服务中心,2020:7.DOI:10.26914/c.cnkihy.2020.029110.

**致谢**

在此，我要向所有在我完成这篇论文过程中给予帮助和支持的人们表示感谢。

在这里首先要感谢我的导师王新法老师。老师平日里工作繁多，但在我做毕业设计的每个阶段，都细心帮助，提出一些新颖功能，并且耐心指导，他的专业知识和经验对我的研究起到了至关重要的作用，使得我的论文可以更加顺利的完成。

然后要感谢三月小组的李学勇老师，在大学四年中关心我们的学习状况，扩展我们的知识体系，使得我们在以后的学习中接触新知识快速，容易上手，也使得我实习过程顺利，找到了一个好的工作。

此外，我还要感谢这四年来全体老师对我的培养，他们在我学习中给予了我很多有价值的建议和意见，帮助我解决了许多问题。

最后感谢信息工程学院和我的母校—河南科技学院四年来对我的大力栽培。