成 绩：

江 西 科 技 师 范 大 学

毕业论文（设计）

题目（中文）：基于 Spring Boot 的在线商城系统设计与实现

（外文）：Design and implementation of the online shopping mall system based on Spring Boot

院（系）：元宇宙产业学院

专　　业：计算机科学与技术

学生姓名：万一苇

学　　号：20213583

指导教师：

年　月　日

目录

基于 Spring Boot 的在线商城系统设计与实现

摘要：随着国民经济的稳步发展、人们收入水平的不断提高以及互联网技术的发展，我国网民数量已达 10.92 亿个人，其中网络购物用户规模达 8.54 亿。2023 年全年网上零售额 15.42 万亿元，占社会消费品零售总额的比重逐年上升，标志着网上在线购物逐步走向人们的生活之中，并且在社会节奏逐步加快的时代，越来越多的人们渴望在碎片化的时间内可以购买到自己心仪的商品，而因为网络购物的便捷，成为了成为了他们在忙碌之中购物的最好选择，因此市场的发展前景良好。本文首先介绍了在线商城的相关研究背景和现状。其次，介绍了开发在线商城过程中使用的所有技术栈，并从多个角度进行了可行性分析，然后介绍了本项目中各个实体之间的关系与数据库的设计，描述了本系统的界面设计与设计思路，最后对本系统的各个模块之间进行测试，并进行总结与展望。

关键字：网上商城；Spring Boot；Vue3；B/S

1.引言

1.1 研究背景

近年来，伴随着我国经济的稳步攀升，民众的收入水平的提高以及互联网的基础设施建设的大力推进使得网络覆盖了全国的城乡镇，根据中国互联网络信息中心（CNNNIC）第 53 次《中国互联网络发展状况统计报告》中提及我国网民数量已达 10.92 亿人[1]，这意味着全国超近 8 成国民都已经接入互联网。那么在网络普及率如此高的背景下，电子商务尤其是网络购物呈现迅速发展的势头。

根据数据报告显示，我国的网络购物用户规模达 8.54 亿，占全国网民的近 8 成，足以看见网上购物的市场之大，重要之重。2023 年全年网上零售额更是达到 15.42 万亿元，占社会消费品零售总额的比重逐年上升，网上购物已经从新型消费模式逐渐成为主流，对传统的购物方式产生了深远的影响，成为了消费市场的主流。

并且在社会节奏逐步加快的时代，越来越多的人们愈发渴望在碎片化的时间内可以购买到自己心仪的商品，而因为网络购物凭借着不受时间和空间的限制、操作方便、商品繁多和价格便宜等特点，成为了成为了广大消费者在忙碌之中购物的最好选择。不论是上班族在工作闲暇时间内浏览商品，还是家庭主妇在照顾家庭的间隙之中购买家用商品。网络购物平台都是可以给予他们更多选择，享受轻松、便捷的购物体验。

那么在面对庞大的用户基数、消费旺盛以及用户强烈的期待，开发一个对用户便捷、高效、页面简洁和友好的网上在线商城系统具有重大的社会价值和广阔的市场前景。因此为了满足用户能够方便、快捷的、轻松购买到自己心仪商品，开发一个集方便、快捷、用户友好

于一体的在线商城系统，以满足广大网民日益增长的网购需求，显得尤为必要且迫切。

1.1.1 国内外同类产品对比

在全球的电商市场中，各种网上在线商城各具特色，以下是国内外同类产品的对比。

国内：

(1).淘宝：淘宝作为中国最大的电商平台，拥有庞大的用户基础和商品种类，并且依托阿里巴巴强大的大数据技术支持，能够精准的为用户提供心仪的推荐商品。但是也应为用户数量庞大，导致商家之间的竞争激烈，为了拉取更多用户，会将商品的价格调低导致商家的成本增大，而对于小型的商家而言，更为艰难，并且由于淘宝的开店要求比较低，没有限制，那么对于商品的品质管控就低。在物流配送方面，鉴于淘宝与多家快递公司建立了合作关系，使得每家淘宝店铺可以根据自身的地理位置与业务需求灵活的选择合适的物流公司，保证商品的正常配送和给予用户良好的购买体验，然而，应当指出，尽管如此，仍无法保证所有店铺都能精准匹配到最为理想的物流合作方，那么配送服务方面就会参差不齐。

(2).京东：京东的商业模式主要以自营模式为主，商家自行采购商品和销售商品，通过设定合适的销售价格，靠着价格与采购成本之间的差值来进行盈利，这种营销方式必然伴随着压低商品价格，甚至不惜牺牲毛利率，来换取庞大的销售量。这种做法的目的是不仅是为了获取可观的利润，更是为了能够获取到庞大的现金流，以便店铺的持续运营和可持续发展奠定基础，而且由于商家是自营，那么对于品质的管控就掌握在自己手中。且京东开店门槛高，要求公司或者企业才能入驻，而且需要有注册资金。在物流方面由于京东拥有自己的物流体系，那么就能够提供快速、高效的配送服务。

国外：

(1).亚马逊：作为全球电商巨头之一，亚马逊以其跨国运营的广泛覆盖力及庞大的商品种类闻名于世，但同时，平台对于卖家的运营规范设定了极为严苛的标准。一旦卖家账户的绩效表现长期未能达到规定要求，亚马逊有权采取强制关闭店铺的措施。不仅如此，亚马逊运营的挑战并不仅限于商品上架，如何在遵循平台规则的基础上有效利用自然流量提升曝光、如何策划并执行付费流量推广策略以吸引潜在客户、以及如何将流量高效转化为实际销售，这些都是卖家在亚马逊运营过程中所面临的复杂难题。尽管亚马逊的卖家数量相对较少，约在 950 万级别[2]，竞争压力相对较小，其主要客户群体定位为欧美中高收入阶层，具有强劲的购买力和较高的利润空间。然而，入驻亚马逊需提交大量资质文

件，开店流程相对繁琐，且卖家必须严格遵循平台的高标准规则，否则将面临账户绩效不达标导致店铺被强制关闭的风险。

1.2 研究目标与内容

1.2.1 研究目标

本系统主要的研究目标是基于 Spring Boot 技术，设计并实现一个交互系统简单、页面简洁且响应迅速的网上在线商城，让用户无需花费过多的时间成本，即可轻松的浏览商品和购买商品。

1.2.2 研究内容

本系统是根据网上在线商城的发展趋势与市场需求，结合前沿技术与创新理念，研究并开发的一款旨在满足用户便捷化、个性化购物需求，显著降低商家运营成本、助力商家高效服务提升，并能创造大量就业机会的高品质网上在线商城系统。

2.相关技术简介

2.1 B/S 架构

B/S 架构是在 C/S 架构的基础上经过不断地发展和变化而来的。B/S 架构（Browse/Server，浏览器/服务器）广泛应用于现代 Web 应用程序中，该架构的核心思想是将主要的业务逻辑放在服务器端，用户在使用 B/S 架构时，无需进行相应的客户端软件安装，用户只需要具备百度、谷歌、火狐等浏览器，便可以通过对应网页进行对应的架构体系操作[3]。在这样的结构下大大优化了客户端电脑上的负担, 简化了系统的开发、维护和使用[4]。通过该框架结构以及植入于操作系统内部的浏览器，该结构已经成为了当今软件应用的主流结构模式[5]。

2.1.1 B/S 架构的优点

(1).跨平台的兼容性

B/S 架构实现了对多种操作系统和移动端的无缝支持。用户端只需要一个现代浏览器，

通过访问 Web 应用程序的网址，即可访问到该应用程序，极大的提高了兼容性

(2).客户端的易维护

由于 B/S 架构的核心思想是将主要的业务逻辑放在服务器端，因此客户端不需要负责任何的复杂业务逻辑，只需要将服务器传输过来的数据通过浏览器内核的渲染引擎将数据渲染到页面上，展示给用户。用户不需要下载或安装其他的软件，只需要确保浏览器的版本合适，就能够体验到最新的应用程序的功能和界面。

(3).安全性高

B/S 架构能够采用网络安全协议，如 HTTPS 等，为客户端和服务器端之间的数据传输交流提供了加密保护，并且由于客户端不存储敏感的数据和逻辑，即使客户端遭受攻击或被盗窃，也不会导致存储在服务器的数据泄露，提高了安全性。

2.1.2 B/S 架构的缺点

(1).服务器端压力与成本

由于 B/S 架构主要的业务逻辑和数据的处理都集中于服务器端，所以服务器端需要承担较大的计算和存储压力。而且随着访问的用户数增多，可能要不断地更新服务器硬件资源、优化业务代码，这会增加运营的成本与难度。

(2).通信开销大

在 B/S 架构中，用户和服务器之间的交互主要是通过 HTTPS 请求和响应进行，每一次用户的操作都需要一次网络请求与响应，因此当用户数量多时，这样的交互频繁的场景会消耗大量带宽，并增加服务器的负载。

2.2 Spring Boot

随着互联网的兴起，Spring 占领了 Java 领域轻量级开发的重要位置[6]。Spring Boot 框架是在 2013 年开始研发并在 2014 年发布第一个版本的一个开源的轻量级的 Java 开发框架[7]，其设计的目的是简化 Spring 应用烦琐的搭建以及开发过程，它只需要使用极少的配置，就可以快速得到一个正常运行的应用程序，开发人员可以从编写大量常规和重复的配置代码中解脱出来。

2.2.1 Spring Boot 的优点

(1).简化配置

Spring Boot 通过自动配置极大的简化了程序员在开发业务代码时的配置工作，提高了工作效率，它会根据项目中所引入的依赖自动注入相关的 Bean 配置，使得程序员们只需要将大量的精力关注到业务代码逻辑上，减少了使用 XML 文件配置。

(2).快速开发与启动

Spring Boot 拥有众多的 starter 依赖，只需要将它们导入，就可以快速的进行开发，并且由于 Spring Boot 其内置了 Tomcat 服务器，使得 Web 应用无需部署就可以直接运行，大大加快的开发速度。

(3).生态丰富

Spring Boot 是基于 Spring 框架设计的，因此能够无缝集成 Spring 全家桶及其他各种的开源库，如 Spring Data、Spring Security、Spring Cloud 等。当然，也支持非 Spring 项目的集成。

2.2.2 Spring Boot 的缺点

(1).学习周期长

尽管 Spring Boot 优化和简化了许多配置操作，但是对于第一次接触 Spring Boot 框架的人来讲，学习的周期可能会很长，因为 Spring Boot 的生态丰富，所以开发者需要投入一些时间和精力来了解各个模块如何使用。Spring Boot 通过简化配置操作，减少了手动配置的步骤，但是这也要求开发者对其工作原理要有深刻的理解。

2.3 Vue3

vue.js 是 2014 年发布的渐进式 JavaScript 框架，采用的是自底向上增量开发的方法[4]，随着开发技术的演进和社区需求的增大，Vue 团队在 2020 年正式发布了 Vue3，Vue3 作为 Vue2 版本的重大更新，在保持 Vue2 核心理念和易用性的同时，引入了一系列关键改进和新特性，旨在提升性能、增强类型安全性、优化开发体验并适应现代 Web 开发趋势。相比于其他框架，Vue 专注于视图层的处理和渲染，通过组件化的方式来实现可重用性和模块化开发，同时提供了响应式和声明式的数据绑定，让开发者可以更加方便地管理和操作应用程序状态[8]。

2.3.1 Vue3 的优点

(1). 性能提升

随着 ES2015 的标准化 JavaScript 得到了重大改进，主流浏览器终于开始为这些新特性提供良好的支持，其中最让值得关注的是 Proxy 代理对象，它允许外界访问指定的目标对象之前，先执行拦截操作，提供这样的机制允许对外界访问进行一些修改操作。Vue 框架的核心功能是实现了数据的动态绑定以及视图的响应式更新，这就意味着当用户在进行交互导致的数据变化或状态改变时，Vue 框架会自动的更新相应的 Dom 和视图。在 Vue2 中响应式的实现通常需要使用到 getter 和 setter，通过这些向外暴露的访问方法来追踪数据的变化并动态的更新视图,但是以这种方式实现响应式更新具有局限性，比如说要动态的添加某个对象或者属性的话，那么就无法实现 getter 和 setter 方法来实现对该某对象或该数据实现响应式更新，因此为了解决这种问题，在 Vue3 中引入了 Proxy 代理对象来消除 Vue2 的一些限制。

(2). 灵活的组合式 API

引入的灵活的组合式 API，允许开发者可以在同一个作用域内声明所有的相关逻辑，更加符合开发者编写代码的习惯，有助于提高代码的可读性和可维护性。

2.3.2 Vue3 的缺点

(1).兼容性较低

由于 Vue3 依赖 ES6 新增加的 Proxy 特性，因此 Vue3 无法直接支持没有拥有 Proxy 的老旧浏览器。

(2).生态较差

新版本发布初期，其他部分第三方插件需要花费一定的时间来支持 Vue3，尽管 Vue3 已经发布了一段时间，但是在特定的场景下仍有可能存在兼容性问题。

2.4 Redis 缓存技术

缓存是一种存储技术，旨在通过减少访问延迟、提高数据读取速度来优化数据的存取效率，Redis 缓存也是缓存的一种形式,它是一个开源的基于内存存储的数据库,可支持如链表、集合等多种数据类型。可以使用简单的 Redis 命令快速处理大量的数据,实现对数据高并发读

写[9]。通过 Redis 缓存可以将用户需要查询的数据存储到缓存中，降低数据库的访问次数，减少数据库压力并且还可以提高数据回显给用户的速度，特别是在查询操作远远大于修改操作中。

2.4.1 Redis 缓存的优点

(1).高性能

Redis 将数据全部存储到缓存中，提供了极低的延迟和极高的吞吐量，非常适合需要快速读写操作的场景

(2).简单易用

Redis 提供了许多命令，使用起来非常简单，特别是使用图形化界面的时候，输入命令时会给予相应的提示，这些简单的命令使得开发者可以快速上手并进行高效的数据操作。

2.4.2 Redis 缓存的缺点

(1).内存限制

由于 Redis 是基于内存的，所以存储数据的容量仅仅受限于服务器的内存大小，一旦内容耗尽，那么 Redis 可能无法再存储更多的数据。

(2).与数据库一致性的问题

在缓存和数据库进行写操作时，要保持数据的一致性就需要额外的进行设计和实施，会要求开发者拥有一定的能力。

3.系统分析

这一部分是对网上在线商城项目进行全面的系统分析，旨在明确实现该项目的可行性、项目需求和项目设计，为后面实现该项目的实施奠定坚实的基础。

3.1 系统可行性分析

3.1.1 技术可行性

当前的互联网技术、大数据技术已经非常的成熟，为搭建网上在线商城提供了坚实的技术支持，前端可以采用现代的 JavaScript 技术框架，如 Vue 等技术来实现用户交互良好的前端页面，后端采用 Spring Boot 技术框架，保证后端服务器应用的稳定。数据库采用 MySQL 等关系型数据库来存储海量的商品信息和用户信息。必要的时候还可以采用 Redis 缓存技术来提高系统的性能。

3.1.2 社会可行性

如前面所述，我国的网民数量庞大，网络购物的用户数量增多，网上零售额占社会零售总额比例增大，市场的需求旺盛，为网上在线商城提供了广阔的市场前景。并且因为电子商务是通过互联网等信息网络销售商品或者提供服务的经营活动，是数字经济和实体经济的重要组成部分，是催生数字产业化、拉动产业数字化、推进治理数字化的重要引擎，是提升人民生活品质的重要方式，是推动国民经济和社会发展的重要力量[10]，国家为保证电子商务领域的良性发展，提出了一系列相关的法律法规来保护电子商务领域。

3.2 功能需求

为了解决用户线上购买网络商品的问题，本文旨在探讨通过 B/S 架构设计与实现一款功能相对完整的网上在线商城系统。本系统涵盖了八大核心功能，分别是用户账号管理模块、商品信息管理模块、订单状态管理模块、商品类别模块、用户申请开店申请模块、商品评论与举报管理模块、联系客服功能以及平台或店铺数据统计模块。其中，用户账号管理模块是旨在为管理员提供对用户信息的增删改查、用户权限的修改以及用户的账号状态管理；商品管理模块允许管理员对全部商品增删改查、上架和下架商品，而对商家只允许对自己店铺的商品增删改查、上架和下架；订单管理模块允许管理员对全部的订单增删改查，允许修改订单的状态（未发货、已发货、退款、用户已签收、交易完成），而对商家只允许对自己店铺的订单增删改查，允许修改部分订单的状态（未发货、已发货、退款）；商品分类模块只允许管理员进行增删改查，而店家只能在这些分类中选择商品的分类；用户申请开店模块只允许用户申请，提供用户的账号、密码、店名、商店图片，提交申请，帮助用户可以快速的在

平台上建立一个属于自己的店铺；管理商品评论举报模块负责处理用户对商品的评价与举报，维护公正、透明的购物环境，并且管理员可以对举报人进行金额的奖励，以便促进更多人加入到维护公正、透明的购物环境的工作中；联系客服模块允许用户可以直接与商家进行联系，了解该商品的用处和说明自己的疑问；数据统计模块则通过收集、分析各类运营数据，为管理者提供销售报表、流量分析、用户行为洞察等决策支持信息。本系统在技术方面，前端采用 Vue3 框架结合 Element−plus 组件库，通过 Axios 进行前后端请求发送和响应，旨在实现给予用户良好的交互体验的前端页面。后端遵循 MVC 设计架构，以 Spring Boot 框架为基础，搭配 Mybatis 作为持久层框架，实现业务逻辑、数据库数据和视图显示清晰分离，保证了代码的可读性和可维护性。数据库方面选用在 WEB 应用方面最好的关系型数据库 Mysql，为用户提供稳定的数据存储。

3.2.1 系统功能图

图 1. 网上在线商城功能总体的模块图

通过对本系统功能的需求分析，如图 1 所示，本系统主要分为下面 3 个主要模块

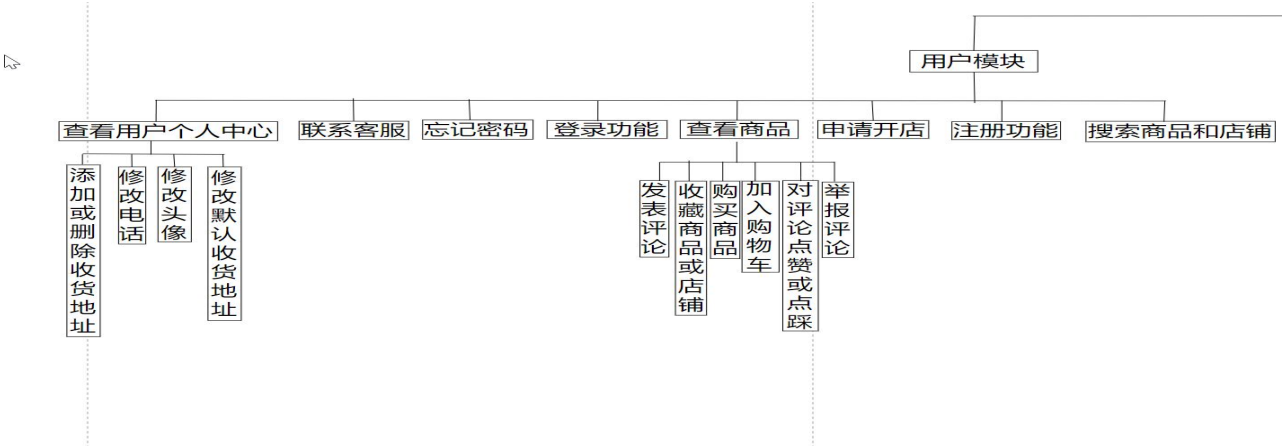

图 2. 用户模块

用户模块是整个电子商务、社交平台以及其他各类的互联网应用中不可缺少的一部分，如图 2 它主要是负责处理用户自己的账号管理、身份验证、商品浏览、搜索商品或店铺、联系客服以及开店申请等功能。这几个功能涵盖了用户从注册、登录到日常的使用的全部操作，为用户提供了安全、简洁和交互良好的操作环境。

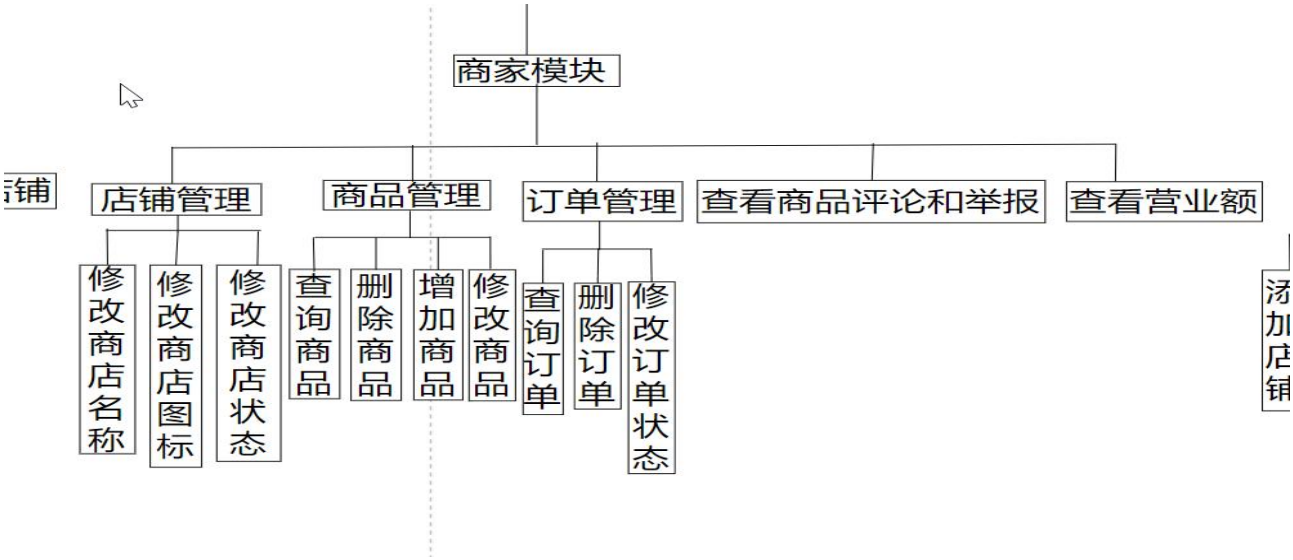

图 3. 商家模块

商家模块作为电商平台中专门为商家设计的一套后端管理系统，如图 3 主要负责商家自己的店铺管理、商品管理、订单管理、查看商品评论和举报以及查看营业额等功能。通过这些功能，商家可以高效地管理自己店铺的各项业务，提高经营效率，更好地为用户提高优质的服务。

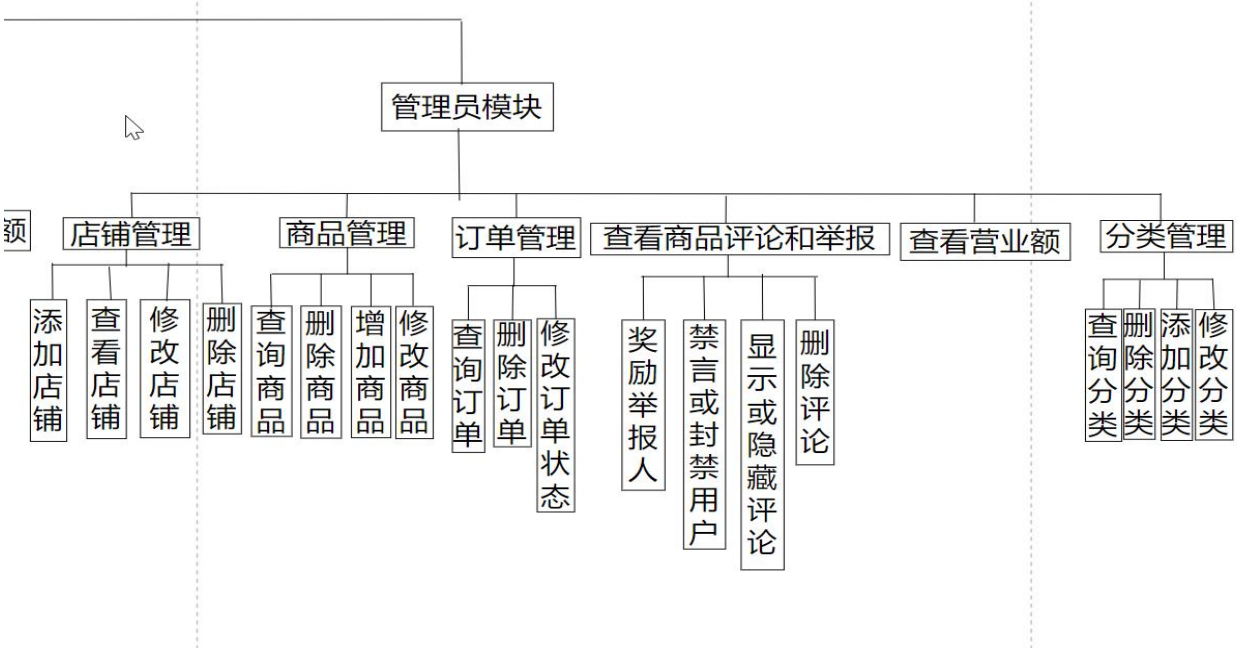

图 4. 管理员模块

管理员模块是整个电商平台的后台管理系统的核心部分，如图 4 主要拥有管理整个平台的店铺管理、商品管理、订单管理、分类管理、查看商品评论和举报以及查看营业额等功能，这些功能赋予管理员可以整个平台进行全面的管理与监控，确保平台可以规范的运行，维护市场的良好秩序，提升用户的购物体验。

3.2.2 系统数据流图

(1).顶层数据流图

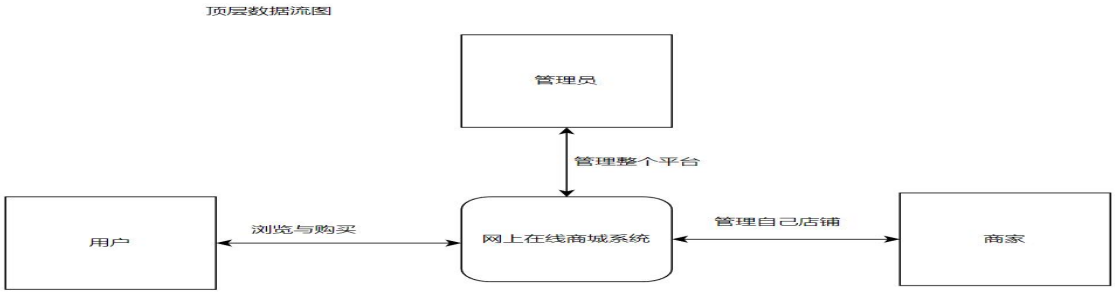

图 5. 顶层数据流图

如图 5 所示，在顶层数据流图中，管理员在网上商城系统中来管理整个平台的数据，而商家负责管理自己的店铺，用户负责在网上商城中浏览商品与购买。

(2).0 层数据流图

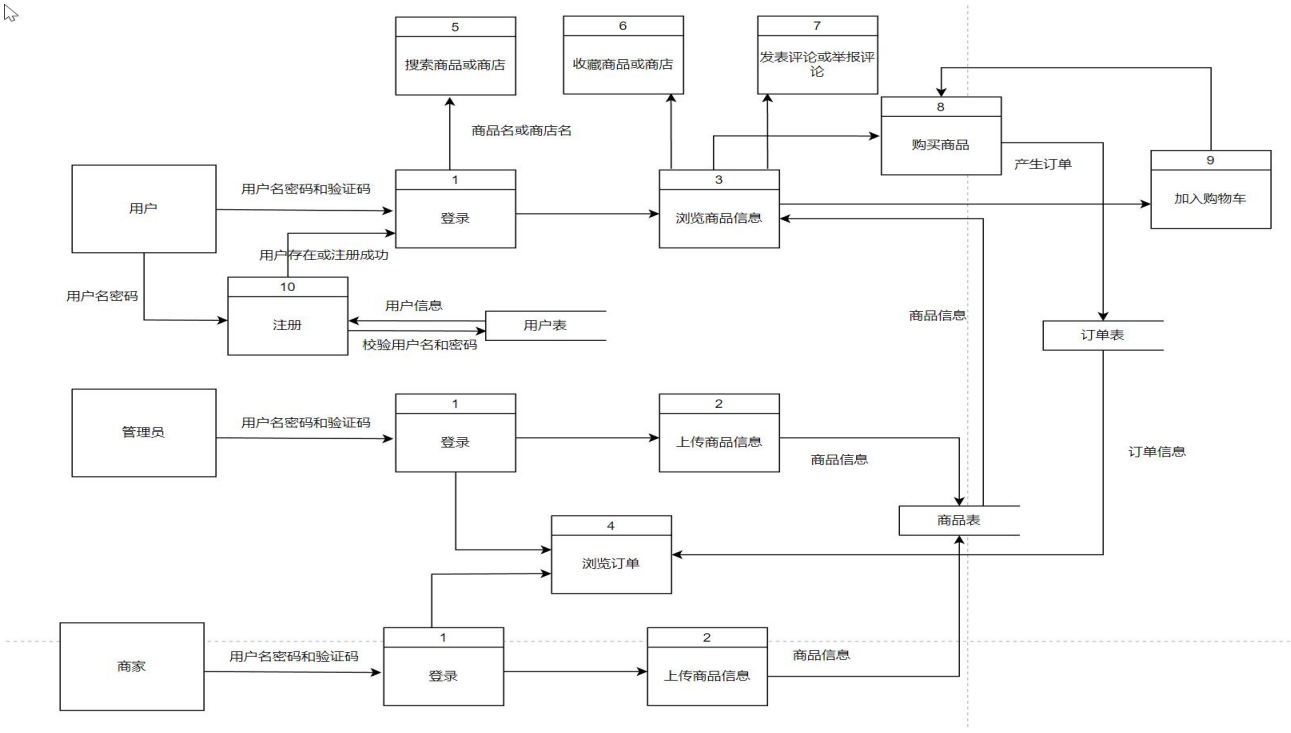

图 6. 0 层数据流图

如图 6 所示，在 0 层数据流图中，用户能够注册账号，如果说账号存在或者注册成功，都让其去登录，用户登录成功后能够搜索商品或商店，还能够浏览商品信息，在商品详情中能够收藏商店或商品、购买商品和加入购物车，还能够针对商品、商店进行发表评论或举报评论。而商家和管理员登录后能够上传商品信息给用户查看，还能够浏览店铺的订单。

(3).用户登录 1 层数据流图

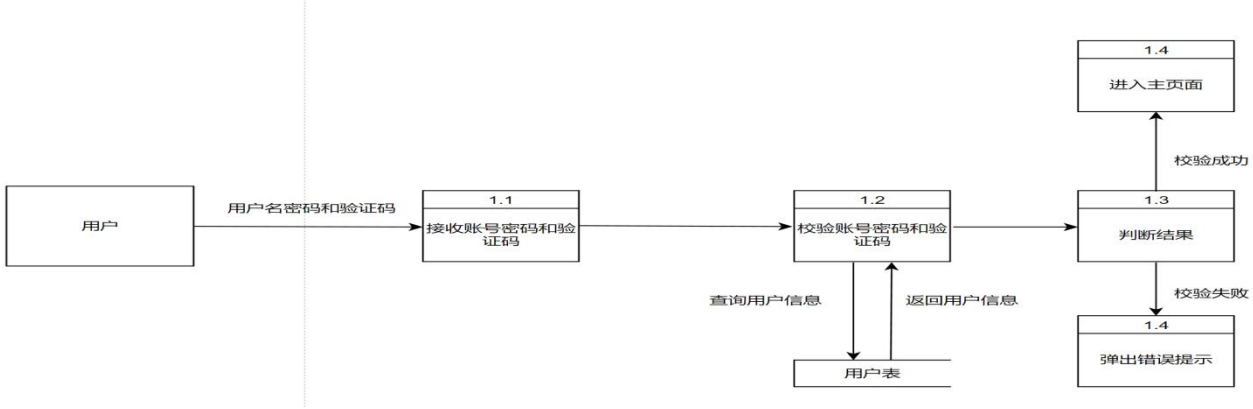

图 7. 用户登录模块的 1 层数据流图

如图 7 所示，用户在前端界面输入其专属账号、密码以及系统生成的验证码，随后将这些认证信息提交给后端服务器。后端接收到请求后，首先执行严格的校验流程，包括确认用户账号在系统用户表中的有效性以及密码与数据库中存储的加密密码的一致性。若查无此用户或密码校验失败，系统将立即向前端返回错误提示信息；反之，若验证无误，系统判定登录成功，并授权用户访问主页面。

(4).用户购买商品模块的 1 层数据流图

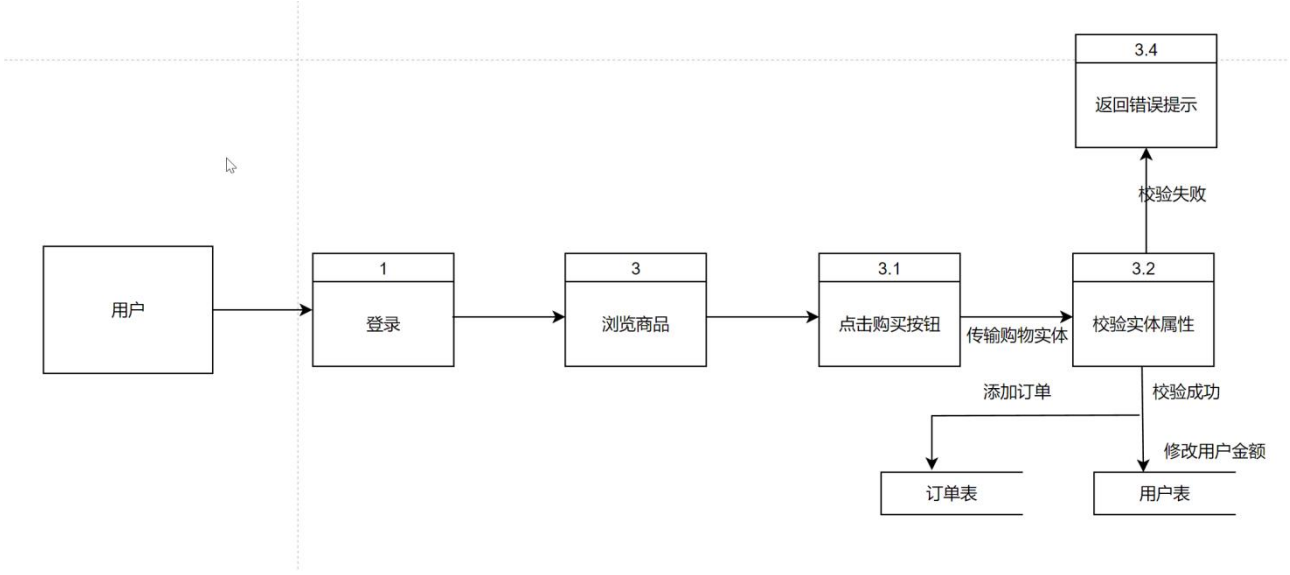

图 8. 用户购买商品模块的 1 层数据流图

如图 8 所示，当用户成功登录并浏览商品时，对心仪商品点击“购买”按钮，前端即刻

将包含商品信息及用户购买意愿的购买实体提交至后端服务器。后端接收到请求后，首先进行全面的属性合法性校验，确保购买请求中各项参数均符合预设规则。若校验通过，系统立即着手创建订单实体，详实地记录此次交易的各项细节，并将其安全地存入订单数据库。与此同时，系统精准地核算此次购买对用户钱包金额的影响，及时更新用户账户余额。但是如果购买实体属性存在非法或无效项，系统将毫不迟疑地向前端返回清晰的错误提示信息，指导用户修正购买请求。

4.系统总体设计

4.1 概述

在结构化设计中，总体设计阶段要将软件需求转化为数据结构何软件的系统结构[11], 它承担着将前期软件需求分析阶段确定的高层次需求转化为具体的、可实现的系统设计元素的任务。总结设计阶段的成果可以为后续详细设计和代码开发提供了可靠的方向，确保软件开发能够按照指定的路线进行。保证软件的质量和开发效率。

4.2 设计原则

软件设计总共有六大设计原则，分别为单一职责原则、开闭原则、里氏替换原则、迪米特法则、接口隔离原则以及依赖倒置原则，而网上在线商城应该采取如下原则：

(1).单一职责原则：在面向对象设计部分采取单一职责原则，其核心思想为，一个类最好只做一件事[12]。比如在网上在线商城中，不同模块负责不同功能，如商品管理、订单管理、用户管理等，每个模块应该只负责相应的功能，确保单一职责。

(2).接口隔离原则：各个模块应该只依赖它们需要的接口，而不需要实现全部的接口。比如说网上在线商城中用户模块应该只需要应该需要的接口，不需要的接口应该不实现。

(3).开闭原则：模块实体应该遵循对扩展开发和对修改关闭，当需要添加新功能时，应该通过添加新的代码来完成，而不是修改已有代码，保证已有的功能不受新功能影响，减少引入新功能时对系统的风险。。已有的代码在添加新功能时不应该被修改，这样可以保持系统的稳定性和可靠性。如果每次添加新功能都需要修改已有的代码，系统将变得难以维护和理解。

4.3 数据库设计

4.3.1 E-R 图设计

如图 9 所示（由于表和属性太多，故省略属性，只保留实体和实体间的关系），用户与用户收货地址是 1 对多的关系，1 个用户可以有多个收货地址，而一个收货地址只对应着一个用户，这样的设计允许用户灵活添加、编辑或选择不同的收货地址，提高了在线购物的便利性和用户体验；用户和开店申请是 1 对多的关系，1 个用户可以发送多个开店申请，但是一个开店申请只属于一个用户，这样设计是为了允许单个用户多次提交开店申请，为用户提供重试机会或申请不同类型的店铺成为可能，无需为每次申请创建新的用户账号，增强了系统的灵活性和扩展性；用户和商店是 1 对 1 的关系，1 个用户只能拥有一个店铺，而 1 个店铺也只属于 1 个用户，这样设计是为了确保商家可以更加的专注于单个品牌建设和顾客体验；用户和购物车的关系为 1 对多，1 个用户可以拥有多个购物车，而 1 个购物车只属于一个用户，这样设计用户在购物车中添加任意数量的不同商品或同一商品的不同规格，方便管理、修改购物意向，直至最终结算；用户和评论之间的关系是 1 对多，1 个用户可以发表多个评论，但是 1 个评论只能被 1 个用户发表，允许每个用户可以自由地对不同内容或者意见的反馈，增强了平台的互动性和用户的参与性；用户和收藏之间是 1 对多的关系，1 个用户可以收藏多个商品或商店，而 1 个商店或商品只能被 1 个用户收藏，这样设计为了更好地适应用户的多样化需求，促进用户互动；用户和订单之间是 1 对多的关系，1 个用户可以拥有多了订单，但是 1 个订单只能是 1 个用户所产生的，这样设计是基于电子商务中用户购买行为的自然特性，既满足了用户多样的购物需求，也便于平台进行高效管理和数据分析；1 个用户可以发送多条消息，而 1 条消息只属于 1 个用户，这样设计每个消息直接关联到唯一的发送者，减少了数据冗余；商店和商品之间是 1 对多的关系，1 个商店可以拥有多个商品，但是一个商品只能归属于一个商店，这样设计允许一个商店拥有多个商品条目，能够反映现实世界中零售或线上店铺的商品多样性；商店和店铺地址之间是 1 对 1 的关系，1 个商店只能拥有 1 个店铺地址，1 个店铺地址也只能属于 1 个店铺，这是因为一个店铺通常只有 1 个经营地址，实体店由于其物理位置的唯一性，自然对应一个具体的地址。这保证了顾客能够准确找到店铺位置，进行线下访问或作为物流配送的目标地点；商店和营业额是 1 对多的关系，这样做是为了更好地适应商业运营的动态特性，支持深入的数据分析，满足财务管理需求，并确保系统的灵活性和可扩展性。

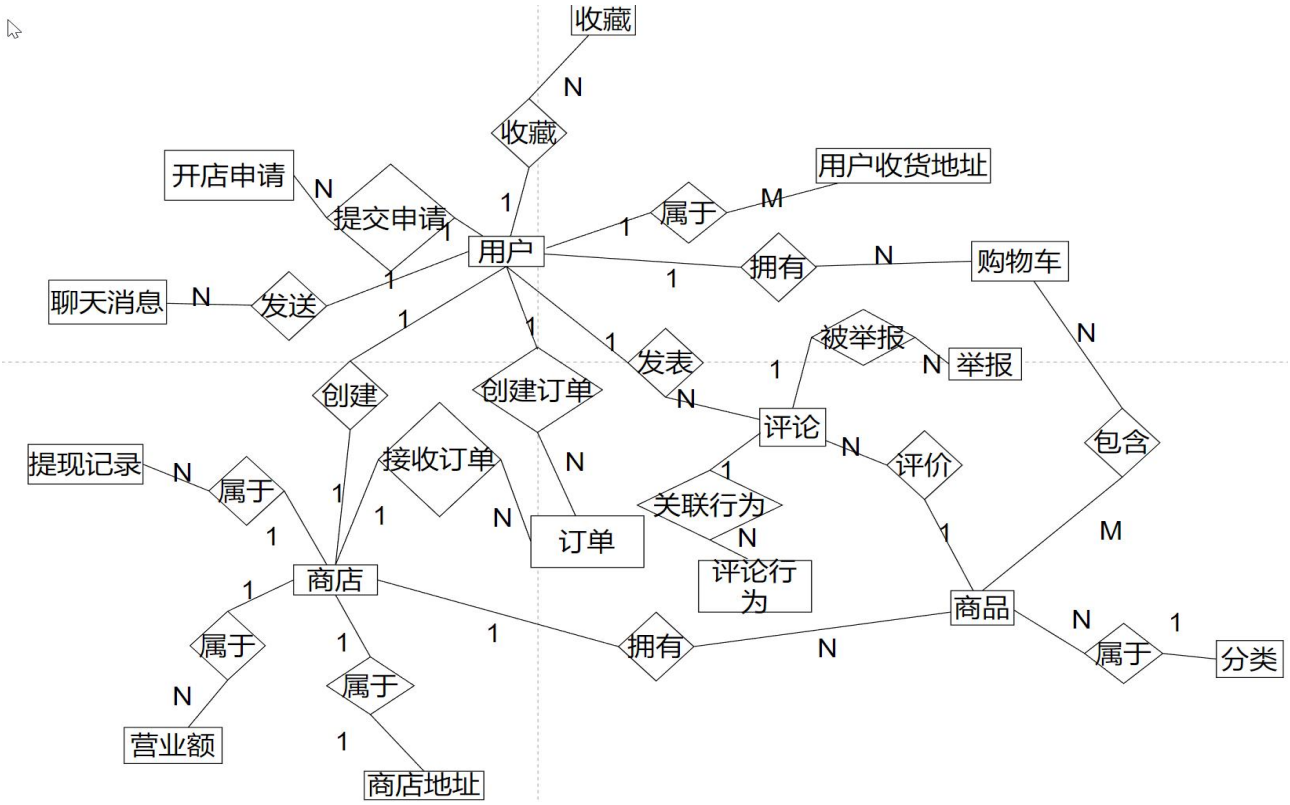

图 9. 网上在线商城 E-R 图

4.3.2 数据库表设计

通过用户类中的属性得到表 1 用户数据库表，该表的作用是用来存储用户信息的

表 1. 用户数据库表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	用户编号
username	varchar	11	NO	N	NULL	账号
password	varchar	255	NO	N	NULL	密码
phone	char	11	YES	N	NULL	电话号码
status	int		NO	N	NULL	用户身份 0：管理员 1：普通用户 2：商家
account_status	int		NO	N	NULL	账号状态 0 禁用 1 启用
ban_start_time	datetime		YES	N	NULL	封禁开始时间
ban_end_time	datetime		YES	N	NULL	封禁结束时间

forbidden_word	int		NO	N	1	禁言状态 1 不禁言 0 禁言
forbidden_start_time	datetime		YES	N	NULL	禁言开始时间
forbidden_end_time	datetime		YES	N	NULL	禁言结束时间
is_online	int		NO	N	NULL	用户在线状态 0 不在线 1 在线
avatar	varchar	255	YES	N	NULL	头像
update_time	datetime		NO	N	NULL	修改时间
money	decimal		NO	N	0.00	用户的钱包（仅模拟支付）
create_time	datetime		NO	N	NULL	创建时间

通过商店类属性得到表 2 商店的数据库表，该表的作用是用来存储商店的信息

表 2. 商店数据库表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	id
user_id	bigint		NO	N	NULL	用户 id
store_name	varchar	255	NO	N	NULL	店名
status	int		NO	N	NULL	店的状态 0 关店 1 开店
logo	varchar	255	YES	N	NULL	商店 logo
create_time	datetime		NO	N	NULL	创建时间
update_time	datetime		NO	N	NULL	修改时间

通过商品类属性得到表 3 商品的数据库表，该表关联了商店的 id 表示了该商品是属于哪个商店的，还关联了分类的 id 表明了该商品是归属于哪个类别。

表 3. 商品的数据库表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	商品 id
store_id	bigint		NO	N	NULL	店家 id
category_id	bigint		YES	N	NULL	分类 id
goods_name	varchar	255	NO	N	NULL	商品名
price	double		NO	N	NULL	价格
total	bigint		NO	N	NULL	商品总量

discount	double		NO	N	NULL	商品折扣
description	varchar	255	NO	N	NULL	商品描述
update_time	datetime		NO	N	NULL	修改时间
cover_pic	varchar	255	YES	N	NULL	商品封面
status	int		NO	N	NULL	商品状态 0 下架 1 上架
create_time	datetime		NO	N	NULL	创建时间

通过分类类属性得到表 4，该表主要存储类别的信息。

表 4. 分类数据库表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	id
category_name	varchar	255	NO	N	NULL	分类名称
category_status	int		NO	N	NULL	分类状态 1 启用 0 禁用
update_time	datetime		NO	N	NULL	修改时间
create_time	datetime		NO	N	NULL	创建时间

根据购物车类得到表 5，该表主要存储用户购物车信息，该表分别关联了用户的 id 和商品的 id，表明是某个用户将某个商品添加进了购物车。

表 5. 购物车数据库表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	购物车 id
user_id	bigint		NO	N	NULL	用户 id
goods_id	bigint		NO	N	NULL	商品 id
goods_name	varchar	255	NO	N	NULL	商品名称
number	int		NO	N	NULL	购买数量
total_price	decimal		NO	N	NULL	总价
cover_pic	varchar	255	YES	N	NULL	
goods_price	decimal		YES	N	NULL	商品单价
discount	double		YES	N	NULL	商品折扣
update_time	datetime		NO	N	NULL	修改时间
create_time	datetime		NO	N	NULL	创建时间

根据订单类属性创建该表 6，该表关联着用户 id、商品 id 和商店 id 表明着是某用户在某商店购买了某商品。

表 6. 订单数据库表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	订单 id
orders_number	bigint		YES	N	NULL	订单号
user_id	bigint		NO	N	NULL	用户 id
goods_id	bigint		NO	N	NULL	商品 id
store_id	bigint		YES	N	NULL	商店 id
total_price	decimal		NO	N	NULL	商品总价
goods_name	varchar	255	NO	N	NULL	商品名
status	int		NO	N	NULL	订单状态 1 待发货 2 已发货 3 退款
number	int		NO	N	NULL	商品数量
province_code	int		YES	N	NULL	省号
province_name	varchar	255	YES	N	NULL	省名
city_code	varchar	255	YES	N	NULL	市号

city_name	varchar	255	YES	N	NULL	市名
district_code	int		YES	N	NULL	区号
district_name	varchar	255	YES	N	NULL	区名
create_time	datetime		NO	N	NULL	创建时间
update_time	datetime		NO	N	NULL	修改时间

根据用户收货地址类属性得到表 7，该表主要是用来存储用户的收货地址，其中 user_id 就是为了关联到用户的 id，表明哪个用户的地址。

表 7. 用户收货地址表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	id
user_id	bigint		NO	N	NULL	用户 id
province_code	varchar	255	NO	N	NULL	省级区划编号
province_name	varchar	255	NO	N	NULL	省级名称
city_code	varchar	255	NO	N	NULL	市级区划编号
city_name	varchar	255	NO	N	NULL	市级名称
district_code	varchar	255	NO	N	NULL	区级区划编号
district_name	varchar	255	NO	N	NULL	区级名称
detail	varchar	255	NO	N	NULL	详细地址
create_time	datetime		NO	N	NULL	创建时间
update_time	datetime		NO	N	NULL	修改时间
is_default	int		NO	N	NULL	是否为默认地址

根据商店地址类属性得到表 8，该表主要是用来存储商店经营的地址，其中 store_id 关联着店铺的 id，是为了表示该地址是属于哪个商店的。

表 8. 商店地址表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	id
store_id	bigint		NO	N	NULL	商店 id
province_code	varchar	255	NO	N	NULL	省级区划编号
province_name	varchar	255	NO	N	NULL	省级名称
city_code	varchar	255	NO	N	NULL	市级区划编号
city_name	varchar	255	NO	N	NULL	市级名称

district_code	varchar	255	NO	N	NULL	区级区划编号
district_name	varchar	255	NO	N	NULL	区级名称
detail	varchar	255	NO	N	NULL	详细地址
is_default	int		NO	N	1	是否是默认地址
update_time	datetime		NO	N	NULL	修改时间
create_time	datetime		NO	N	NULL	创建时间

根据分类类属性得到表 9，该表主要是存储商品的类别。

表 9. 分类表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	id
category_name	varchar	255	NO	N	NULL	分类名称
category_status	int		NO	N	NULL	分类状态 1 启用 0 禁用
update_time	datetime		NO	N	NULL	修改时间
create_time	datetime		NO	N	NULL	创建时间

根据用户申请类属性得到该表 10，该表通过 username 来关联用户，表示是该用户发出的申请。

表 10. 用户申请开店表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	申请编号
username	varchar	255	NO	N	NULL	用户名
password	varchar	255	NO	N	NULL	密码
store_name	varchar	255	NO	N	NULL	商店名称
province_code	varchar	255	YES	N	NULL	省编码
province_name	varchar	255	YES	N	NULL	省名
city_code	varchar	255	YES	N	NULL	市编号
city_name	varchar	255	YES	N	NULL	市名称
district_code	varchar	255	YES	N	NULL	区编码
district_name	varchar	255	YES	N	NULL	区名称
status	int		NO	N	NULL	申请状态 0 待审核 1 成功 2 拒绝
detail	varchar	255	YES	N	NULL	详细地址
reason	varchar	255	YES	N	NULL	拒绝理由

create_time	datetime		NO	N	NULL	创建时间
update_time	datetime		NO	N	NULL	修改时间

根据收藏类属性得到该表 11，该表通过 user_id 来关联到用户表明是哪个用户收藏的，通过 goods_id 和 store_id 表明该用户收藏的是哪个商店或哪个商品。

表 11. 收藏表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	收藏夹 id
user_id	bigint		NO	N	NULL	用户 id
goods_id	bigint		YES	N	NULL	商品 id
store_id	bigint		YES	N	NULL	商店 id
update_time	datetime		NO	N	NULL	修改时间
create_time	datetime		NO	N	NULL	创建时间

根据提现记录类属性得到该表 12，该表关联了 user_id 和 seller_id 表明是买家和卖家是谁，又关联 store_id 表明了买家是在哪个商店购买的。

表 12. 提现记录表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	id
user_id	bigint		NO	N	NULL	买家 id
seller_id	bigint		NO	N	NULL	卖家 id
store_id	bigint		NO	N	NULL	商店 id
goods_name	varchar	255	NO	N	NULL	商品名
number	int		NO	N	NULL	商品数量
total_price	decimal		NO	N	NULL	商品总价
origin_money	decimal		YES	N	NULL	原先的钱
pay	int		NO	N	NULL	支付方式
update_time	datetime		NO	N	NULL	修改时间
create_time	datetime		NO	N	NULL	创建时间

根据评论类属性得到该表 13，该表关联着 user_id 表明是哪个用户发表的评论，又关联着 store_id 和 goods_id 表明是在哪个商店下的哪个商品下发表评论。

表 13. 评论表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注

id	bigint		NO	Y	NULL	评论 id
user_id	bigint		NO	N	NULL	发表人 id
username	varchar	255	NO	N	NULL	用户名
avatar	varchar	255	NO	N	NULL	用户头像
goods_id	bigint		NO	N	NULL	商品 id
store_id	bigint		NO	N	NULL	商店 id
content	text	65535	NO	N	NULL	评论内容
star	decimal		NO	N	NULL	评分
like_count	int		NO	N	0	点赞数
reply_count	int		NO	N	0	回复数
parent_comment_id	bigint		YES	N	NULL	父评论 id（若为顶级评论，则为 NULL）
comment_status	int		NO	N	NULL	评论状态 1 隐藏 0 显示
create_time	datetime		NO	N	NULL	创建时间
update_time	datetime		NO	N	NULL	修改时间
report_count	int		NO	N	0	举报数量
dislike_count	int		NO	N	0	点踩数量

根据评论行为类属性得到该表 14，该表关联着 user_id 表明是哪个用户采取的行为，又关联着 comment_id 表明用户是对哪个评论操作, action 表示用户是点赞还是点踩还是什么都不做。

表 14. 评论行为表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	id
user_id	bigint		NO	N	NULL	行动人 id
comment_id	bigint		NO	N	NULL	评论 id
action	int		NO	N	3	评论行为
create_time	datetime		NO	N	NULL	创建时间
update_time	datetime		NO	N	NULL	修改时间
goods_id	bigint		NO	N	NULL	商品 id
store_id	bigint		NO	N	NULL	商店 id

根据举报类属性得到该表 15，该表关联 user_id 表明是哪个用户举报的，又关联着 comment_id 表明用户举报的是哪个评论。

表 15. 举报表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	举报
user_id	bigint		NO	N	NULL	举报人 id
comment_id	bigint		NO	N	NULL	被举报评论的 id
reason	varchar	255	NO	N	NULL	举报原因
is_award	int		NO	N	0	是否奖励该举报用户 0 未奖励 1 已奖励
status	int		NO	N	1	举报状态 1 待处理 2 已处理
update_time	datetime		NO	N	NULL	修改时间
create_time	datetime		NO	N	NULL	创建时间

根据聊天消息类属性得到该表 16，该表关联了是用户的 id 和商家的 id，content 代表着发送消息的内容。

表 16. 聊天消息表

列名	字段类型	长度	是否为空	是否为主键	默认值	备注
id	bigint		NO	Y	NULL	主键
content	text	65535	NO	N	NULL	内容
send_id	bigint		NO	N	NULL	发送者 id
receive_id	bigint		NO	N	NULL	接收者 id
create_time	datetime		NO	N	NULL	创建时间
update_time	datetime		NO	N	NULL	修改时间

5.系统的实现

5.1 页面设计

5.1.1 登录页面设计

如图 10 所示，在用户进入系统时，首先会进入首页，这时候会校验用户是否已经登录获得 token，如果没有 token 就跳转至登陆页面并弹出提示，这时候允许用户输入账号、密码和验证码，然后点击登录按钮，如果登录成功就会将 token 放到浏览器的本地缓存中，并跳转到商城首页，如果登录失败了那么就如图 11 和图 12 所示，会给予相应的提示。

图 10. 登录页面

图 11. 用户名不存在情况

图 12. 密码错误情况

5.1.2 注册页面设计

如图 13 所示，在注册页面中需要输入账号、密码和确认密码后，点击注册按钮，会将数据封装成对象传递给后端校验，如果说注册成功，将会跳转到登陆页面，如果注册失败，那么就如图 14 和图 15 所示，会给予错误提示。

图 13. 注册页面

图 14. 账号已存在情况图

图 15. 两次密码不一致情况

5.1.3 忘记密码页面设计

如图 16 所示，用户需要输入其用户名、新密码和确认密码，点击修改按钮，向后端发送请求，校验所给的数据，如果修改成功，就提示修改成功并跳转到登录页面如图 17 所示，如

果账号不存在，也给予响应的错误提示如图 18 所示。

图 16. 忘记密码页面

图 17. 忘记密码修改成功情况

图 18. 账号不存在情况

5.1.4 首页页面设计

如图 19 所示，当用户登录成功时，就会跳转到首页，首页上可以看到最上面是导航栏，分别有用户姓名下拉框、去往首页的按钮、去往登录页面的按钮、去往注册页面的按钮、如果是商家或管理员的身份就拥有去往工作区的按钮，还有操作下拉框。用户姓名下拉框拥有如图 20 所示，下拉框里拥有用户账号的钱包、充值按钮、去往个人中心页面以及账号退出按钮。操作下拉框拥有如图 21 所示去往用户购物车页面的按钮、去往用户收藏页面的按钮以及去往订单页面的按钮。

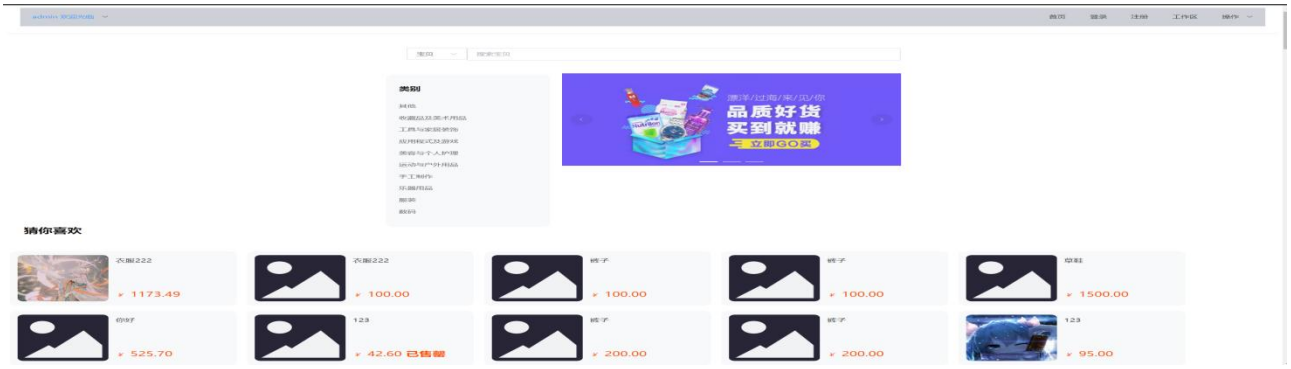

图 19. 首页页面

图 20. 用户姓名下拉框

图 21. 操作下拉框

5.1.5 个人中心页面设计

如图 22 所示，当用户点击个人中心时，就将跳转到该页面，在该页面用户可以更换头像、查看账号、手机号、钱包余额、查看收货地址以及修改收货地址、修改默认地址、删除收货地址、添加收货地址。整个页面的设计风格应与网站整体保持一致，提供清晰的视觉层次和反馈。

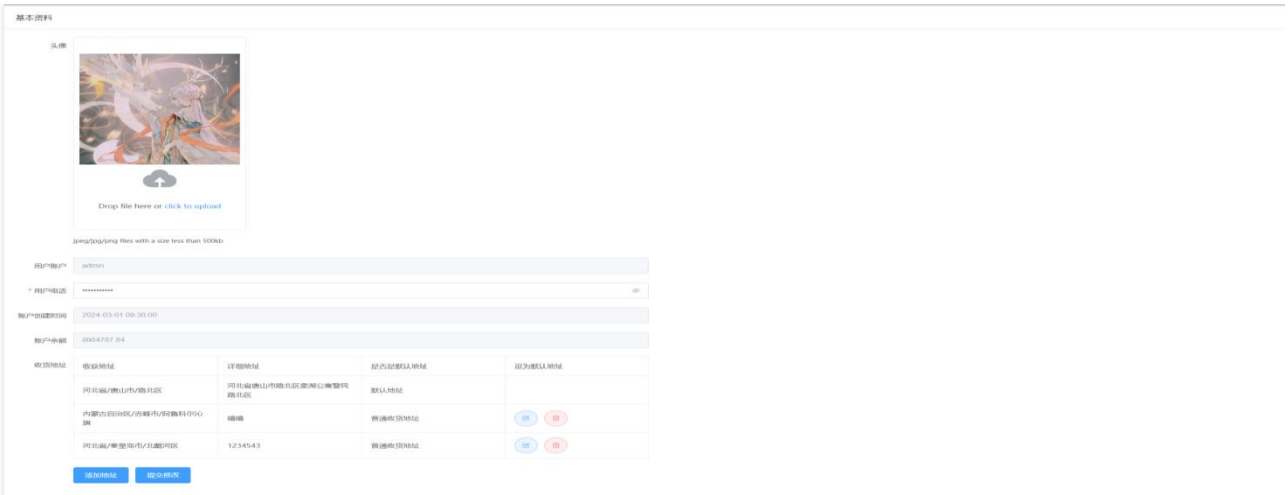

图 22. 个人中心页面

5.1.6 商品详情页面设计

如图 23 所示，商品详情页拥有一个搜索框，用户可以在搜索框中输入商品的名称然后按下回车即可搜索，下面拥有商店名字、商品的评分以及进入商店的按钮。再往下就是展示商品的详情以及购买和添加购物车按钮，之后还能对其他用户对商品的评论。如图 24 所示，用户可以在评论区中，对其他用户的评论进行点赞、点踩还有举报，点击上面一排按钮可以选择显示哪种类型的评论。

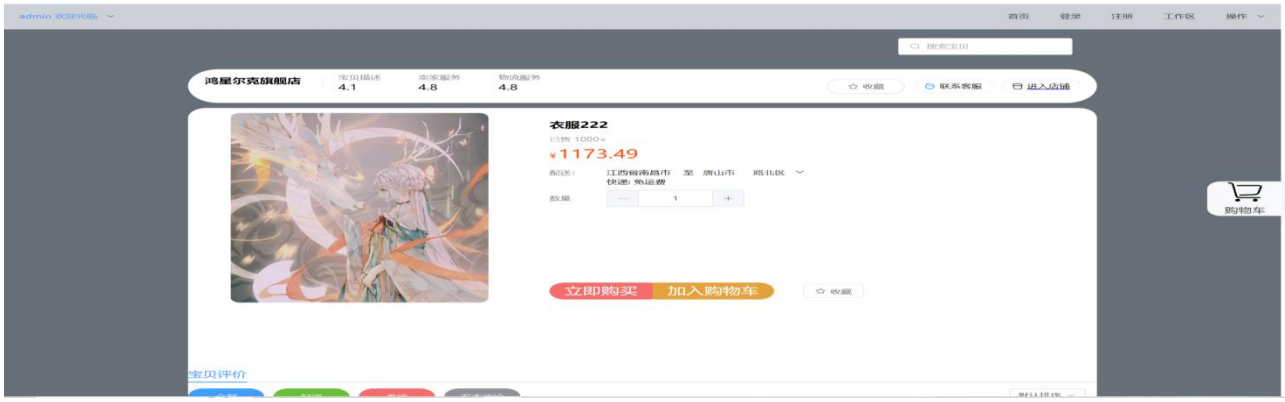

图 23. 商品详情页面

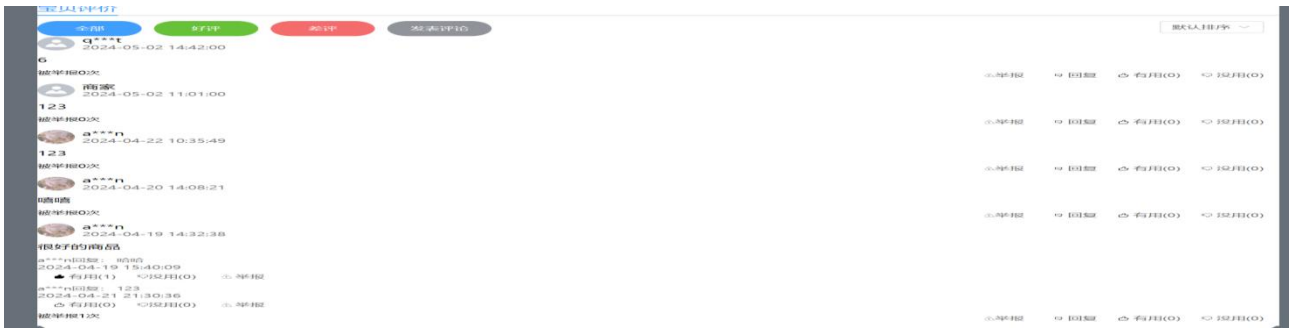

图 24. 评论区

5.1.7 购物车页面设计

如图 25 所示，在购物车页面中，用户可以在通过选择下拉框来选择搜索框中要输入的是商品名称还是店铺名，下面就是购物车项，用户可以通过勾选购物项来进行结算支付或者删除或者将商品移入收藏夹中。对于每一个购物项，用户可以调整商品的数量，结算时只看最后的被勾选的商品数量总和。

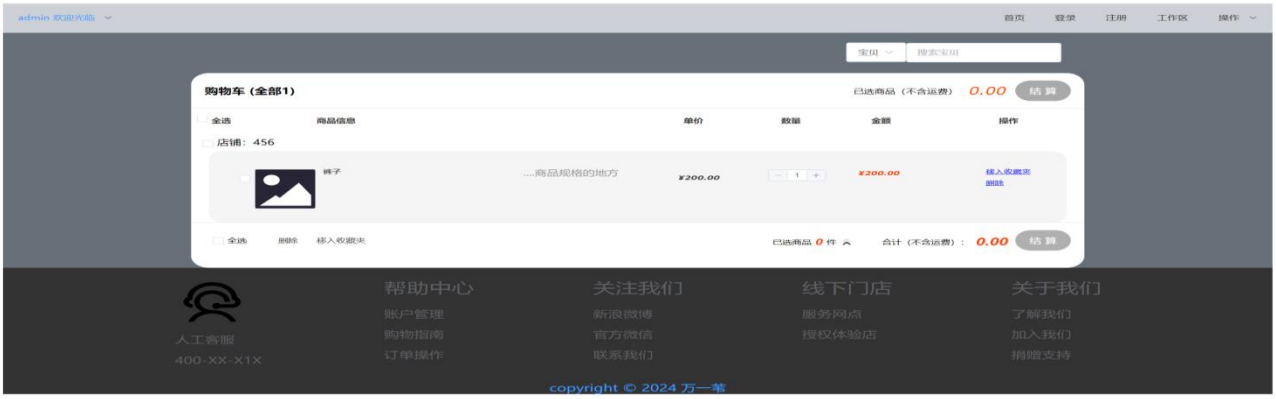

图 25. 购物车页面设计

5.1.8 用户订单页面设计

如图 26 所示，用户点击右上角操作下来框选择我的订单后，进入到用户订单页面，该页面存放着用户购买过的所有商品，用户可以选择性的查看哪种类型的订单，对于每一个订单，用户可以查看到订单号、购买时间、购买数量、花费金额、支付方式、订单当前的状态，如果说订单还未发货，用户可以点击催单，那么对应的商家会接收到催单提醒如图 27 所示。并且如果说用户接收到订单，那么用户可以在 7 天内无理由退款。

图 26. 用户订单页面

图 27. 用户催单提醒

5.1.9 管理员后台页面设计

如图 28 所示，在管理员后台页面中，管理员可以查看平台的营业额，平台抽成、平台中的用户信息、商品信息、分类管理、商店信息、用户开店申请、订单信息、评论举报信息以及管理员自己的个人信息，这些功能不仅能够高效地让管理员进行日常运营管理，还能为平台长期的发展奠定基础。

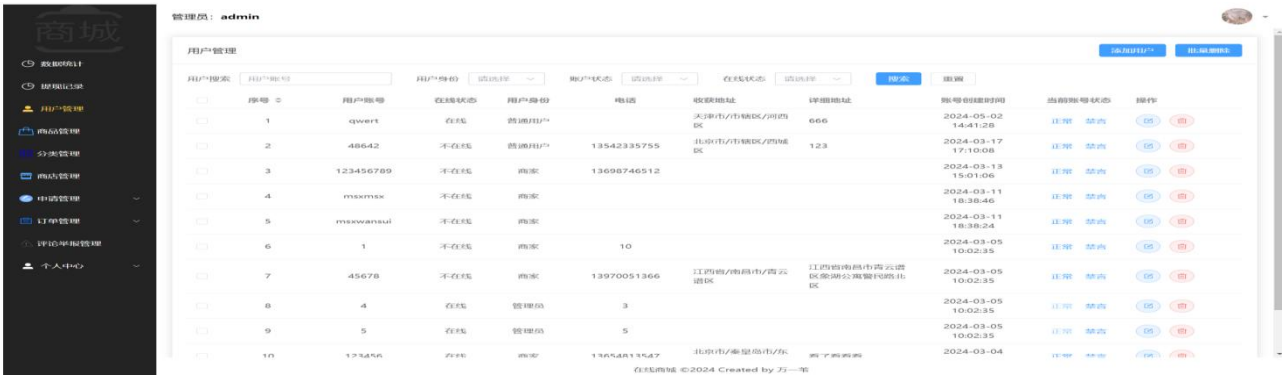

图 28. 管理员后台页面

5.1.10 商家后台页面设计

如图 29 所示，商家可以在后台查看自己店铺的营业额，查看平台抽成、管理店铺的商品、管理店铺的订单、查看店铺商品下的评论举报以及商家自己的个人信息。这样的设计不仅提升了商家的运营效率和顾客满意度，也促进了平台与商家之间的信任与合作，为双方的共赢创造了条件。

图 29. 商家后台页面

6.系统测试

6.1 测试目的

系统测试是软件开发过程中至关重要的一步，其目的是为了保证软件的质量并修复在程序中发现的错误，验证程序功能是否符合预定的需求，是否能满足用户的期望，以便提高程序的稳定性与可靠性。系统测试不仅仅只测试系统的代码是否有问题，还包括测试各个模块无缝集成，共同实现预期的系统功能以及发现并修补潜在的安全漏洞，保护软件免受各种网络攻击，维护数据的安全和隐私。系统测试是一项综合性的质量保障活动，它从多角度、多层次出发，确保软件在投入市场前，能够在功能上满足用户需求，还能够在安全、性能等其他方面都能够给予用户一个良好的体验。

6.2 测试原则

系统测试的核心在于紧密跟随需求，强调从项目初始阶段就融入的尽早测试策略，追求测试范围的广泛性和深入性，确保每个环节都能独立且公正地被检验。以用户体验为基本测试原则，从用户的正常使用角度作为出发点进行多项功能测试[13]，保证软件可以高效、安全地运行。

6.3 黑盒测试

6.3.1 注册模块测试

测试页面如图 13 所示，测试过程如表 17 所示。

表 17. 注册模块测试

测试场景	输入数据	测试结果	预期结果
输入框为空	无	用户名输入框提示用户名必须是 5-11 位的字母或数字 密码输入框提示密码必须是 6-15 位的非空字符 确认密码输入框提示密码必须是 6-15 位的非空字符	用户名输入框提示用户名必须是 5-11 位的字母或数字 密码输入框提示密码必须是 6-15 位的非空字符 确认密码输入框提示密码必须是 6-15 位的非空字符
用户名已存在	用户名:admin	提示账号已存在	提示账号已存在
密码和确认密码不一致	密码 123456 确认密码 456789	确认密码的输入框下将会用“两次输入的密码不一致”的错误提示	会有两次输入的密码不一致提示
用户名格式不对	用户名:@ldwoq	用户名输入框提示用户名必须是 5-11 位的字母或数字	会提示用户名必须是 5-11 位的字母或数字
注册成功	账号 fieqpw 密码 123456 确认密码 123456	提示注册成功，并跳转到登录页面	提示注册成功，并跳转到登录页面

6.3.2 登录模块测试

登录页面如图 10 所示，测试过程如表 18 所示。

表 18. 登录模块测试

测试场景	输入数据	测试结果	预期结果
输入框为空	无	与预期相符	用户名输入框提示用户名必须是 5-11 位的字母或数字 密码输入框提示密码必须是 6-15 位的非空字符
用户名不存在	用户名:fmidawj	与预期相符	提示账号不存在
密码错误	账号:admin 密码 123456	与预期相符	提示密码错误
验证码错误	验证码 gxjaz	与预期相符	提示验证码错误

验证码过期	验证码 gxjaz	与预期相符	会提示验证码过期，请刷新页面或刷新验证码
登录成功	账号 admin 密码 123456	与预期相符	登录成功，并跳转到首页

6.3.3 首页搜索模块

首页页面如图 19 所示，测试过程如表 19 所示。

表 19. 首页搜索模块

测试场景	输入数据	测试结果	预期结果
输入框为空	无	与预期相符	提示输入内容不能为空或全为空格
选择宝贝且输入内容不空	衣服	与预期相符	跳转到所有含有衣服二字的搜索结果页面
选择店铺且输入内容不空	鸿星尔克	与预期相符	会跳转到所有包含鸿星尔克名字的搜索结果页面

6.3.4 购买商品模块

商品详情页面如图 23 所示，测试过程如表 20 所示

表 20. 购买商品测试

测试场景	输入数据	测试结果	预期结果
购买金额不足	无	与预期相符	提示金额不足，请充值
商品数量小于购买数量	无	与预期相符	提示该商品已售罄
没有默认地址	无	与预期相符	提示请去个人中心添加收货地址
购买成功	无	与预期相符	提示购买成功消息并扣除钱包中的相应金额

6.3.5 购物车模块

购物车页面如图 25 所示，测试过程如表 21 所示

表 21. 购物车模块测试

测试场景	输入数据	测试结果	预期结果
正常添加商品至购物车	选择一个可购买的商品，点击“加入购物车”按钮	与预期相符	商品成功添加到购物车，购物车商品总数+1，提示信息正确
购物车商品数量变更	点击购物车项中，商品数量的加减号按钮变更商品数量	与预期相符	购物车商品数量增加或减少
清空购物车功能	点击全选框，然后点击最下面的删除按钮，点击确认框	与预期相符	提示删除成功
购买成功	勾选商品点击结算按钮	与预期相符	提示购买成功，并删除被勾选的商品，对应用户扣除相应的钱包金额

6.3.6 个人中心模块

页面设计如图 22 所示，测试过程如表 22 所示。

表 22. 个人中心模块测试

测试场景	输入数据	测试结果	预期结果
更换头像	点击更换头像，选择有效图片文件	与预期相符	头像更换成功并正确的回显
地址编辑	选择一个地址点击编辑，修改部分信息并点击提交按钮	与预期相符	提示修改成功
设置默认地址	点击普通收货地址的编辑按钮	与预期相符	提示修改成功
添加收货地址	点击添加收货地址按钮，填写完整的收货地址和详细地址信息并保存	与预期相符	提示添加成功

6.4 测试结果分析

在完成以上基本功能的测试后，通过对比实际测试结果与预期目标，我们得出结论：系统功能表现符合设计预期，整体运作顺畅。测试覆盖了登录、注册、搜索、购买商品、加入购物车、个人中心等多个核心环节，每一项功能均能按既定流程正确执行，未发现影响使用的重大缺陷。用户界面友好，响应迅速，确保了良好的使用体验。此外，系统在错误处理、数据一致性及安全性方面也展现出稳健性能，能够有效引导用户操作并保护用户信息安全。

7.结束语

本文设计了一款在线商城系统，能够有效实现商家及顾客进行在线商品交易的需要，并能有效提高商品交易的便捷度，减少商品流通环节，并节约交易成本[14]。选择开发网上在线商城方向的初衷是由于观察到电子商务在全球范围的迅速发展，以及人们的网上购物需求日益增长，在这个趋势下，我看到了一个巨大的市场机遇，心中就诞生了一个能够开发一个简单版淘宝的想法，想要创造一个便捷、高效以及页面简洁的网上在线商城，希望可以让用户和商家之间的距离拉近，让交易变得更加顺利。然而，在梦想照进现实的过程中，我也不得不面对自我能力与项目需求之间的差距。特别是在前端设计方面，我意识到，虽然我对 HTML 和 CSS 有一定的了解，但在实现一个既美观又实用的用户界面时，我的技能显得捉襟见肘，尤其是当我决定采用先进的 Vue 框架来实现前后端分离时，这种差距感尤为明显。我原计划

的简洁设计，在实际操作中遭遇了重重困难，未能完全达到预期的视觉效果，这让我深刻领悟到，一个成功的在线商城，前端的 UI 设计与后端的强大逻辑同等重要，它们共同决定了用户的初次印象与长期体验。

而且在实现本系统的过程中我尝试了通过 Git 来实现代码版本管理控制，大量减少了开发人员不必要的工作，可以有效解决代码冲突、事务并发、文档冗余等问题，实现对项目进行高效的管理[15]。让我深刻地领悟到掌握项目版本管理的价值，通过不断的实践逐渐地掌握了 Git 简单的管理代码方法。

这段经历还教会了我，面对技术难题时，勇于探索和持续学习的态度比任何先验知识都来得重要。每当遇到障碍，我都会主动查阅资料或者直接阅读源码来寻找答案。这一过程虽然充满挑战，但也正是这些挑战，让我从一名技术的初学者逐渐成长为能够完整开发一个项目的开发者。

随着互联网的迅速发展和电子商务的全球化趋势，电子商务已经逐渐渗透到各个行业领域，成为现代商业活动中不可或缺的一部分[16]，网上在线商城系统正步入一个前所未有的革新阶段，特别是增强现实技术和虚拟现实技术的飞速发展[17]，我相信这将逐步实现让用户可以在家里就能够亲身体验商品，无论是试穿新衣服还是预览家具布局，这种沉浸式的交互过程模糊了线上线下的区别，使得用户可以更好的聚焦于商品购物过程。而且网上在线商城正在与社交元素的深度融合，让电商不仅仅是购物平台，更成为了社交互动的场所。各大电商借助社交媒体的力量，通过直播、短视频、广告等形式，使得电商系统变成了一个充满活力的社区，促进了商品口碑传播，提高了销售额。

综上所述，网上在线商城系统正在朝着更加智能、互动的方法前行，不断地探索新技术和新方向，给予用户新的体验，推动电子商务行业不断的扩大。

参考文献

[1] 中国互联网络信息中心(CNNIC). (2024). 第 53 次《中国互联网络发展状况统计报告》[R]. 北京: 中国互联网络信息中心.

[2] Elad B. (2024). Amazon Seller Statistics 2024 by SMBs vs Enterprise, Device Traffic, Social Media Advertising, Concerns and Alternate Platforms [EB]. Enterprise Apps Today.

[3] 朱宝善, 陈光浦, 李鹏, 王深.基于 B/S 模式和 MySQL 的人力资源管理系统设计 现代科学技术[J].现代电子技术.2021(14).66

[4] 曹明昊.基于 SpringBoot 和 Vue 框架的邯郸市现代农业园区信息管理系统的研发[D].河北工程大学, 2022.DOI:10.27104/d.cnki.ghbjy.2021.000671.

[5] 王虎. 基于 B/S 模式的信息管理平台[D]. 天津: 天津大学, 2017. DOI:10.7666/d.D01670461.

[6] 马绍阳, 王伟东, 韩斌倩, 等.基于 Spring Boot+Vue 的智能远程医疗平台的设计与实现[J]. 网络安全技术与应用, 2024(01):55-57.

[7] 李昊. 基于微信小程序的智能推荐点餐系统的设计与实现[D]. 南京邮电大学, 2021.DOI:10.27251/d.cnki.gnjdc.2020.000283.

[8] 季甜甜, 刘冬冬.基于 Vue 前端性能的研究与分析[J].阜阳师范大学学报(自然科学版), 2024, 41(01):15-22.DOI:10.14096/j.cnki.cn34-1069/n/2096-9341(2024)01-0015-08.

[9] 李建华, 夏汛, 罗明全.基于 ThinkPHP 和 Redis 的高并发微信公众号开发的研究与实现[J]. 计算机应用与软件, 2019, 36(02):108-112.

[10] 中华人民共和国商务部, 中央网络安全和信息化委员会办公室, 国家发展和改革委员会. (2022). 商务部 中央网信办 发展改革委关于印发《"十四五"电子商务发展规划》的通知[Z].

[11] 吕云翔. (2020). 实用软件工程(第 2 版)[M]. 北京: 人民邮电出版社.

[12] 韩万江, 陈淑文, 韩卓言, 等. 基于微服务架构的分布式灾情管理系统设计[J]. 中国地震, 2021, 37(4): 806-818.

[13] 罗丹雯, 王振宇, 王孟博.基于微信平台的旅游助手小程序设计[J].黑龙江科学, 2022, 13(08):86-88.

[14] 杨晟, 罗奇.基于 Spring Boot 的在线商城系统设计[J].科技创新与应用, 2022, 12(19):58-61.DOI:10.19981/j.CN23-1581/G3.2022.19.013.

[15] 仇礼钦, 王鑫, 盛飞龙, 等.基于 Git 的软件项目管理配置方法及应用实践[J].机电工程技术, 2023, 52(05):223-227.

[16] 焦世奇, 王新.中国种子企业电子商务的现状及发展趋势[J].分子植物育种, 2024, 22(03):994-998.DOI:10.13271/j.mpb.022.000994.

[17] 李明伟, 薛彦卓, 耿敬, 等.工程教育专业认证背景下港航专业沉浸式教学方法与实践研究[J].中国现代教育装备, 2021(17):87-89.DOI:10.13492/j.cnki.cmee.2021.17.026.

Design and implementation of the online shopping mall system based on Spring Boot

Abstract: With the stable development of national economy, the continuous improvement of residents' income level and the development of Internet technology, the scale of Internet users in our country has reached 1.092 billion, among which the scale of online shopping users has reached 1.092 billion. 854 million · Online retail sales will reach 1.542 billion won in 2023, indicating that online shopping has become the best choice and status quo for busy shopping in the era of increasingly social rhythm of people's lives. Online shopping mall next introduces all the technologies used in the process of developing and creating online shopping mall. After taking pictures from different angles and analyzing the possibility, this introduction program introduces the interface design and core system design ideas entity and database design through the relationship between each. Summary and view of each tested module.

Key Words: Online mall**;**Spring Boot;Vue3;B/S