

郑州科技学院

本科毕业设计（论文）

题 目	<u>二手书买卖系统的设计与实现</u>
姓 名	<u>代春辉</u>
专 业	<u>计算机科学与技术</u>
学 号	<u>202022300189</u>
指导教师	<u>刘芳云</u>

郑州科技学院信息工程学院

二〇二二年五月

目 录

摘 要	I
Abstract	II
1 绪论	1
1.1 选题背景和意义	1
1.2 发展现状	2
1.3 可行性分析	2
1.4 论文组织结构	3
2 关键技术介绍	4
2.1 Spring boot 框架	4
2.2 Mybatis 框架	5
2.3 MySQL 数据库	5
2.4 系统架构	6
3 需求分析	7
3.1 系统设计目标	7
3.2 功能需求分析	7
3.3 非功能需求分析	11
3.4 系统用例图	12
3.5 开发环境	12
4 系统的设计	14
4.1 系统功能结构	14
4.2 系统流程图	15
4.3 安全性设计	16
4.4 详细设计	17
4.4.1 前台功能模块	17
4.4.2 后台功能模块	18
5 数据库的设计	19
5.1 数据库和设计概述	19

5.2 概念模型和设计（E-R 图）	19
5.3 数据库逻辑设计	21
6 系统的实现	25
6.1 网站前台页面	25
6.1.1 系统首页	25
6.1.2 用户登录和注册	26
6.1.3 用户发布图书	29
6.1.4 购物车	30
6.1.5 留言功能	31
6.1.6 轮播图展示	33
6.2 后台管理页面	33
6.2.1 管理员用户登录	33
6.2.2 用户管理	34
6.2.3 图书信息管理	35
6.2.4 留言信息管理	36
6.2.5 图书类别信息管理	37
6.2.6 管理员个人信息管理	38
6.2.7 轮播图信息管理	38
7 系统测试	40
7.1 测试用例	40
7.1.1 用户注册功能测试用例	40
7.1.2 发布图书测试用例	40
7.1.3 图书分类测试用例	40
7.1.4 留言功能测试用例	40
7.2 测试用例的执行	44
7.3 缺陷报告	45
结 论	49
致 谢	50
参考文献	51

二手书买卖系统的设计与实现

摘 要

本文介绍了二手书买卖系统的设计与实现过程，介绍了项目二手书买卖系统的各个功能之间的联系，将软件工程的基本原理和方法应用到整个系统，对其进行需求分析，提出了解决问题的具体方法，在系统实现过程中，利用Java语言为基础，使用Spring Boot、Mybatis、MySQL等框架技术结合，主要分为前台系统和后台系统，普通用户登录前台系统后能够查看图书商品，发布二手书商品，发表自己对二手书评价，购买二手书商品，加入二手书进入购物车等功能，管理员用户登录后可以实现用户管理，二手书管理，地址管理，用户评价管理等。

二手书买卖系统具有界面简单，能够一定程度上满足用户的二手图书交易需求，用户学习成本低，系统维护成本低，且用户受众多的特点，具有一定的开发价值。

关键词：二手书买卖；Java；SpringBoot；MySQL

Design and implementation of second-hand book trading system

Abstract

This paper introduces the second-hand book sales system design and implementation process, this paper introduces the project of second-hand book sales system and the connection between the various functional, apply the basic principle and method of software engineering to the whole system, carries on the demand analysis, put forward the concrete methods to solve the problem, in the process of system implementation, using Java language as the foundation, With the combination of Spring Boot, Mybatis, MySQL and other framework technologies, it is mainly divided into the front desk system and the background system. After logging in to the front desk system, ordinary users can view the book products, release the book products, publish their own book evaluation, buy the book products, add to the shopping cart, and administrators can manage users after logging in. Library management, address management, user evaluation management, etc.

Second-hand book trading system has a simple interface, to a certain extent to meet the user's book trading needs, low user learning cost, low system maintenance cost, and the user by many characteristics, has a certain development value.

Keywords: Second-hand book sales; Java; SpringBoot; MySQL

1 绪论

1.1 选题背景和意义

在倡导节能减排的大环境下，人们对于资源的重复利用也越来越看重，随着教育的普及，书籍是人们获取知识的一大途径，越来越多的人购买书籍，这也造成了大量书籍的闲置和浪费，而资源问题是未来将要面对的主要问题，作为人口的基数庞大，教育人数多的国家，每年二手书的产生是非常巨大的，这些二手书的浪费不仅是浪费了有限的资源，还不利于自然环境的保护。面对这些问题，我们需要一个能够让这些二手书能够重新焕发价值的途径，而二手书买卖系统就是在这种条件下的而开发的，互联网技术已经突飞猛进，各种商品都在网上交易^[1]，通过互联网，将各个地区人们手中的二手书通过交易系统联系起来，能够促进二手书之间的流通，人们可以在二手书买卖系统上寻找自己需要的书籍，也可以把自己不需要的书籍通过二手书买卖系统交易给需要的人。这样不仅能够有效的避免二手书的产生，也能通过二手书买卖系统实现按书籍的循环利用，节约了有限的资源保护了环境。

大学生群体相对于其他用户而言在二手书购买上更加的活跃，随着教学政策的改变，由以前的学生必须购买学校的教材书籍，到而今可以依据自己的需求购买自己需要的教材。由于学校课程编辑更加严格，每一版课程的内容相对较之前变动不大，这也推动了二手书贸易的蓬勃发展，学生通过选择二手适宜的课程，往往能够在符合自身教学需要的情况下降低部分成本，目前而言各个学校的学生进行二手书的买卖主要是通过学校论坛、校园跳蚤市场等线上或者线下的方式，但这些方式由于是由学校组织或者发起的有很大的局限性。论坛的用户很多，但是一个校园论坛并不是只为学生提供简单的二手书交易方式，并且提供的功能相对于图书交易不够全面，展示的信息十分有限，并且由于在线上有很大的不确定性等因素，想要购买特定的书籍十分困难，难以满足日益增长的二手书交易在实际中的真实需求。校内跳蚤市场则存在展示时间过于短暂，并且只有在每年毕业或者新生入校时才有，而且交易售后无法维权，结束后尚未交易的物品无法

处理。而且跳蚤市场上，书籍只是其中很小部分，商品分类模糊。此外就现在形势看来，更多的用户青睐于方便快捷的网上购物，网购已经成为生活中不可获取的交易方式。

1.2 发展现状

据有关数据显示，近 5 年来，全国各阶段课本及教学用书平均每年约 28 亿册，金额超 200 亿元。业内人士指出，如果这些教材循环使用一年，就可以节省 200 多亿元^[2]。在当今环境保护和资源回收利用的浪潮下，二手书的交易热度也越来越高，目前为止，人们为大众所接受的二手书交易主要有两种方式：一是常规的电商平台，如京东、淘宝、拼多多、咸鱼等，利用自身优势和巨大的用户吞吐量，有特定的二手书籍模块，来完成二手书交易的各个流程服务好，技术强。另一种是更加完善，针对性更强的二手书交易平台，例如孔夫子旧书网、多抓鱼等，只做与图书相关的交易，针对性更强，有更好的解决方案。

美国的亚马逊公司针对广大学生群体专门的提供了具有二手买卖功能的苹果手机应用 Amazon Student。用户可以在应用上查询二手商品信息，其中包含二手书部分。美国最大的二手书交易平台是 Craigslist，该平台服务对象已经覆盖了全球多个国家和地区^[3]。

如今二手书对于个人用户来说并不是太友好，用户在使用之前需要进行用户店铺申请，这些对于很多人来说并不是必需的，而在交易发生之前需要支付一定的用户保障费用，操作麻烦，除此之外，也存在一些问题需要去考虑解决，比如系统用户的个人诚信问题，用户的售后维权等。要解决此类问题需要政府和企业培养个人用户诚信意识，打击不诚信行为，这样在可见的未来，我们会拥有让人满意的网购氛围。

1.3 可行性分析

分析系统的可行性是网站开发的第一项工作。它主要包括分析系统开发的可行性，根据需求、操作、技术和经济。通过对这些不同方面的分析，考虑整个系统达到预期的效果，以下是主要内容的可行性分析：

（1）需求可行性分析

需求分析主要是在设计客户所需的分析系统时进行的。一旦系统的可行性分析完成，一个满足消费者需求的重新设计的网站将有助于系统实施细节的系统开发；如果需要对没有结果的系统进行分析，则必须在设计之前进行必要的分析，以确保所开发系统的累积价值，以方便用户的导航。

（2）操作可行性分析

当在网站上执行操作可行性分析时，用户可以很容易地查看此消息，通过用户登录后的简单界面，所有功能都可以清楚地显示在网站上，即通过导航栏快速访问。整个网站的操作和设计将更容易。

（3）技术可行性分析

本文主要对可能的技术进行了分析，特别是在网站设计中主要采用 java 技术和数据库技术，mysql 数据库是开源的关系数据库管理系统，也是查询数据库结构最常用的语言之一。通过系统的分析，使用该技术能够较好的完成系统的开发。

（4）经济可行性分析

经济可行性主要涉及用于开发该系统的资金筹措问题，同时考虑到该系统是否会在投资于该系统后产生利润，如果没有利润，就没有必要开发该系统，并且本二手书买卖系统前台业务投入和后台数据库管理的费用并不特别高，分析显示了其在经济上的可行性。

1.4 论文组织结构

论文的组织结构如下：

第一章：绪论，介绍了系统的发展现状进行市场调研，本文的主要研究工作。

第二章：关键技术介绍，介绍了系统开发过程中使用的主要技术框架和方法

第四章：系统的设计，对系统主要功能进行设计，以及系统工作流程。

第五章：数据库的设计，详细介绍了系统所使用的各个数据库表。

第六章：系统的实现，介绍了系统设计的页面和主要代码。

第七章：系统测试，对系统功能进行测试，包括测试用例，测试结果，和缺陷报告。

2 关键技术介绍

2.1 Spring boot框架

SpringBoot 是一个用以简化 Spring 开发的框架。用来进行快捷 Spring 应用开发，采用“约定大于配置”的原则，在使用 SpringBoot 时只选择合适的框架或组件，SpringBoot 就可以生成一个企业级的 Spring 应用项目^[4]。开发人员通过这种方式可以快速开发与部署 Java EE 项目。

SpringBoot 应用系统开发中常使用的前端模板的基本架构模板引擎经常使用的有 FreeMarker、Thymeleaf 等，拿到数据后通过渲染实现页面展示，通过这种方式使界面的设计与应用后端的主要业务逻辑分离，通常在进行前端开发还会使用到 Bootstrap、AngularJS、JQuery、Vue 等前端框架；后端返回格式采用 Json 串的方式，为了更好的实现系统开发的标准化，同时提供 RestFulApi 风格接口，数据访问层常使用的持久层框架有 Hibernate、MyBatis、JPA 等；数据库常用 MySQL。

开箱即用，OutOfBox，是指在进行软件开发过程中通过项目管理工具 MAVEN 在管理项目时，在 pom 文件中添加相关依赖包，然后通过对应注解来代替系统开发中的繁琐 XML 配置文件以管理 bean 对象的生命周期。减少配置工作以及依赖管理工作，把更多的时间用在处理相关的业务逻辑上。

约定优于配置，Convention over configuration，是通过由 SpringBoot 来配置系统目标结构，由开发人员在此基础上增添所需的功能和项目机构。虽然在一定程度上使部分代码使用的灵活性有所下降，使得 BUG 查找的时间复杂度变得复杂，但是减少了大量的 XML 配置，并且可以将代码编译、测试和打包等工作自动化。

由于 SpringBoot 框架是以 Spring4.0 为基础的，因此，其适应能力相当好，同时，Spring 框架是 Java 平台上的一种开源应用框架，提供具有控制反转特性的容器^[5]。Springboot 框架继承了 Spring 框架的特性，并且在此特性上开发出在现实开发中能够提高开发效率，加快开发速度，简化开发流程。相较于使用原生 Spring 在二手书买卖系统中使用 Spring Boot 框架比使用 Spring 框架更加便捷，高效。

2.2 Mybatis框架

Mybatis 是一个优秀的持久层框架,其对 JDBC 操作数据库的过程进行封装,使开发者只需关注 SQL 本身即可^[6]。MyBatis 支持 DDL 数据查询,支持数据存储过程和高级映射的持久层框架。使用 MyBatis 就不需要 SQL 代码编写和 SQL 参数的手工设置以及数据结果集的检索。

MyBatis 的前身是著名的 IBATIS 框架,是一个数据持久层框架,在操作数据库时,会自动调用开发人员预先编写在配置文件中的 SQL 来完成,开发人员就省去了大量调用 JDBC 接口的重复性工作^[7]。

通过映射注解的方式解除 SQL 与程序代码的耦合:通过提供数据库操作接口进行分离,使系统设计更清晰,更方便。SQL 和代码的分离,提高了系统的可维护性。

我们把 Mybatis 的功能架构分为三层:

(1) API 接口层:开发人员通过提供给外部前端用于数据调用的这些本机 API 接口来操作数据库。接口层接收到来自前端的数据调用请求后,将调用数据处理层完成特定的数据处理或返回数据。

(2) 数据处理层:应 API 界面层的要求,它进行特定的 SQL 研究,进行分析,并进行结果映射。完成数据库的具体操作,特别是响应 API 接口的请求。

(3) 基础支撑层:负责系统的基本功能实现支持,通过在系统中出现重复的代码或系统配置,将它们隔离并打包成系统组件或方法。可以根据需要调用它,而不必重新配置或实现它。

2.3 MySQL数据库

MySQL 是一个关系型数据库管理系统,由瑞典 MySQL AB 公司开发,属于 Oracle 旗下产品。MySQL 是最流行的关系型数据库管理系统之一,MySQL 通过定义主键、外键和候选键使关系模式规范化^[8]。在 WEB 应用开发方面,通常 MySQL 是人们优先选择和使用人数最多并且个人免收费的数据库软件之一。

在市面上的众多数据库访问软件上 MySQL 所使用的 DDL 语言是用于访问数据库常用标准化语言之一。为不同受众,MySQL 用户数据允许双重策略版本软件

提供，分为社会和商业数据库中特别小，许多方面得到数据接入快速低成本，许多开发中小型网站优先考虑。今天，随着社会的发展，数据正在迅速增长。目前，数据主要通过计算机技术和数据库技术存储，以便在日常工作中更好地使用。

2.4 系统架构

系统架构是系统开发的核心骨架，一个好的系统架构能够更好的体现系统的功能，能够体现一个系统的功能健壮性、未来可拓展性、操作灵活性、稳定可靠性、信息安全性等，是为系统开发提供了一个结构清晰、行为明确的数据模型，由模块的描述、模块的相互作用、指导构件集成的模式以及这些模式的约束组成。在系统开发中软件架构不仅是系统需求和软件结构之间的对应关系，且指定了整个软件系统的组织和拓扑结构，提供了一些设计决策的基本原理。

二手书买卖系统使用 B/S 架构开发，使用 B/S 的主要优点就是只需通过浏览器进行页面访问操作就能使用，而不需要安装任何的软件。客户端零维护。系统的扩展非常容易。维护和升级方式简单。在进行维护时，所有客户机都只是浏览器，不需要进行任何客户机维护。无论有多少用户或分支，维护和升级的工作量都不会增加。所有操作只需在服务器端进行。如果是远程服务器，您只需将服务器接入企业网络，进行远程维护、升级和共享即可。降低成本^[9]。

按照三层体系结构的特性和功能，其结构包括表示层、功能层、数据层。

表示层：系统服务器对通过对当前系统用户进行身份认证，并通过 HTTP 协议将用户请求的主页发送给客户端。客户端接收并将其显示在用户的 Web 浏览器上^[10]。

功能层：当系统接收到用户发送的服务请求时，会先执行相对应的应用程序请求并与系统数据库，并通过数据处理应用程序的数据库服务器通过对等的方式。然后等待数据库提交数据处理结果到 Web 服务器,然后 Web 服务器将其传输回客户端。

数据层：当接收到 Web 服务器的数据操作请求时，通过对系统数据库数据信息的查询、修改和更新功能，并将运行结果提交给 Web 服务器。

3 需求分析

3.1 系统设计目标

二手书买卖系统就是为了解决人们日常生活中闲置的书籍的交易系统，人们可以在系统中处理自己不需要的书籍，和购买自己需要的书籍，提高了闲置书籍的利用，为系统用户节省一笔费用，为低碳经济做出贡献^[1]。

系统主要为 Java 语言为基础，利用 SpringBoot、Mybatis 等技术框架实现，实现了一个交易系统的基本功能，包括用户登录、用户注册、发布图书、轮播图、购物车管理。

系统分为前台系统和后台系统，前台系统用户主要普通用户，后台系统用户主要为管理员，包括用户管理、图书管理、留言管理等功能模块。

对于典型的交易系统，尤其像二手书这样的交易系统，必须要满足使用方便、操作灵活等的设计需求。本系统的设计时主要表现在以下几个方面：

- (1) 基于人机交互对话的设计模式，系统界面设计友好，功能展现灵活，数据存储安全可靠。
- (2) 详细展示系统内所有分类的二手书信息。
- (3) 为用户提供一个方便快捷的二手书查看功能。
- (4) 实现二手书的在线浏览，发布和购买。
- (5) 提供用户登录模块，主要进行用户登录和管理员登录。
- (6) 系统最大限度地实现了易维护、易操作。
- (7) 系统运行稳定安全可靠。

3.2 功能需求分析

二手书买卖系统的主要目的是使用户能够实现图书的在线交易，主要方面包括用户图书的浏览、用户图书的发布、用户进行图书购买、用户对自己喜欢的图书发表留言看法，和用户的注册和登录等功能。在主体功能上，前台系统包括系统首页、用户查看图书详细页面、用户登录页面、用户注册页面、用户发布图书页面、用户个人信息页面、用户订单页面等。系统后台包括后台系统首页页面、

后台系统用户管理页面、后台系统图书管理页面、后台地址管理页面、后台图书分类管理页面、后台轮播图管理页面等。

根据用户权限和登录状态的不同，系统用户可以分为系统游客、系统普通用户和系统管理员。

（1）游客用户

此类用户进入系统后没有进行用户登录或者用户注册的用户，也是系统的潜在用户，这一类用户大多是可能通过其他途径获得了该系统的消息，进入系统详细了解一下的用户，对于此类用户，主要给予了一部分权限，包括用户能够进行图书的浏览、图书详细信息的查看、查看图书评论浏览和进行用户注册，用户登录等功能，但是不能进行图书的购买和发布图书，想要进行图书的购买或者发布图书必须进行用户登录或者注册。

（2）普通用户

作为系统的主要使用者成员，相较于游客用户功能的部分功能，此类用户的系统权限更加完善，成为普通用户后，用户可以随意进行图书的发布和购买，可用度进一步提高，相较于游客用户此类用户的主要功能包括用户经过注册成为系统的成员，能够进行图书信息在线浏览、发布图书、购买图书、发表图书的评论、添加到购物车等功能。

（3）管理员用户

管理员用户是一个系统中权限最高的，此类用户拥有系统的最高权限，可以进行后台系统的各个操作。管理员用户通过系统登录后可以进行二手书信息管理、用户信息管理、评论信息管理、地址信息管理、二手书分类信息管理、轮播图信息管理等。

表 3.1 用户注册用例描述

用例名称	用户注册	用例编号	03001
参与者	系统用户	创建时间	2022 年 04 月 02 日
简要说明	注册是一个系统主要的功能，是进入系统的基础门户，只有在用户注册后，可以使用系统平台基础功能，才能够满足二手书交易的基本需求。		
前置条件	该用户在二手书买卖系统上无账号		
后置条件	用户成功注册后，用户信息存入数据库中，用户可以以此进行登录		
主要事件流	<ol style="list-style-type: none"> 1. 用户点击注册按钮 2. 输入手机号 3. 输入密码 4. 点击注册 		
备选事件流	<ol style="list-style-type: none"> 1. 用户已注册，提示该手机号已注册，请登录 2. 用户信息错误，提示请输入正确的用户名和密码 		
优先级	高	使用频率	高
对象	二手书买卖系统的游客用户、普通用户		

表 3.2 用户登录用例描述

用例名称	用户登录	用例编号	03002
参与者	系统用户	创建时间	2022 年 04 月 02 日
简要说明	登录是一个系统必备的接口之一，是一个系统的开始，当用户进行注册了以后，可以使用登录页面进行用户登录，不同的用户有不同的登录页面。		
前置条件	用户已经注册成为系统用户		
后置条件	用户登录后可以使用系统功能		
主要事件流	<ol style="list-style-type: none"> 1. 用户点击登录按钮 2. 输入手机号和密码 3. 密码通过加密后后台传来的数据进行对比 4. 手机号密码正确后成功登录进入系统 		
备选事件流	<ol style="list-style-type: none"> 1. 手机号输入不合法，出现特殊字符或者英文 2. 用户信息输入错误，手机号或者密码有误 		
优先级	高	使用频率	高
对象	已经通过注册成为系统用户有二手书交易需求的用户		

表 3.3 发布图书用例描述

用例名称	图书发布	用例编号	03003
参与者	系统用户	创建时间	2022 年 04 月 02 日
简要说明	用户成功登录后，拥有了发布自己图书的权限，可以进行自己闲置的二手的发布		
前置条件	用户已经登录系统，有发布图书权限		
后置条件	用户能够发布二手图书		
主要事件流	1. 用户点击发布图书 2. 输入图书名称 3. 输入图书价格 4. 输入图书简介 5. 上传图书封面 6. 点击发布图书		
备选事件流	1. 图书名称为空，提示图书名称不为空 2. 图书价格为空，提示图书价格不能为空		
优先级	高	使用频率	高
对象	已经登录系统，有二手书交易需求的系统用户		

表 3.4 查看图书信息用例描述

用例名称	图书查看	用例编号	03004
参与者	系统用户	创建时间	2022 年 04 月 02 日
简要说明	用户登录系统后可以进行图书详细信息的查看，包括图书简介、图书名称、图书价格、图书封面等		
前置条件	用户登录系统		
后置条件	用户能够购买图书		
主要事件流	1. 用户点击图书封面 2. 跳转到图书信息界面		
备选事件流	1. 图书状态不可用，图书被管理员进行下架导致不可用		
优先级	高	使用频率	高
对象	系统用户和管理员		

3.5 用户信息用例描述

用例名称	用户信息	用例编号	03005
参与者	系统用户和管理员	创建时间	2022 年 04 月 02 日
简要说明	系统用户或者管理员登录后可以对自己或者其他用户进行信息的管理		
前置条件	用户存在或者当前用户为管理员		
后置条件	用户可以使用		
主要事件流	1. 用户点击自己头像后进入个人信息页面 2. 点击个人信息		
备选事件流	1. 用户不可用，用户被管理严封禁		
优先级	高	使用频率	高
对象	系统用户和管理员用户		

3.3 非功能需求分析

为了保证系统的高可用性和安全性等，需要考虑系统在运行中的非功能需求。

高性能：要充分考虑影响性能的各种因素，如：在程序中对数据表进行良好的索引以提高检索速度，对数据的处理应采用低复杂度算法；在网络上，除了服务器环境外，可以考虑使用压缩方法来降低服务器压力；环境上，系统部署地方保证温度合适，避免温度过高影响系统性能。

高可用性：该二手书买卖系统用二手书的交易，在这个过程中，可能会有许多集中发生的事件，比如：在节点过程如公布图书的时候，可能出现瞬间大批量发送需求，可能达到系统处理顶峰不能处理。

低耦合高内聚：尽量使每个模块独立，根据功能或需求明确区分各个模块，提高了核心代码重用率，尽量避免非必要的代码重复；例如，数据库连接、操作等常见功能可以封装成系统工具类，而不是每次需要这些功能的时候都要去进行实现。

可移植性：通常利用系统的配置文件。对于一些在系统迁移过程中常使用的变量，通过配置文件集中起来，这样在下次进行系统部署或者修改的时候可以直接修改配置文件而不需要修改系统，而且效率更加高效。

安全性：如作为交易平台，用户的输入输出较多，要过滤各种 SQL 语句，防止恶意的 SQL 注入，跨站攻击等。使用好 COOKIE 与 SESSION，防止 cookie 欺骗等^[12]。充分利用 CDN 等预防 DDOS 及 CC 攻击。并且，要尽可能地在低层次处

理通用的要求,如对于 SQL 语句的过滤^[13],它可以进行封装数据库操作语句,这样只有一个设置可以开展保护整个站,除了写作的便利,在漏洞的发生也可以迅速修复,但也防止多个修改遗漏的分布。

3.4 系统用例图

对于访问系统的用户状态不同,通过对系统功能的要求共分为游客、注册用户和管理员,则系统用例图如图 3.1 系统用户用例图所示。

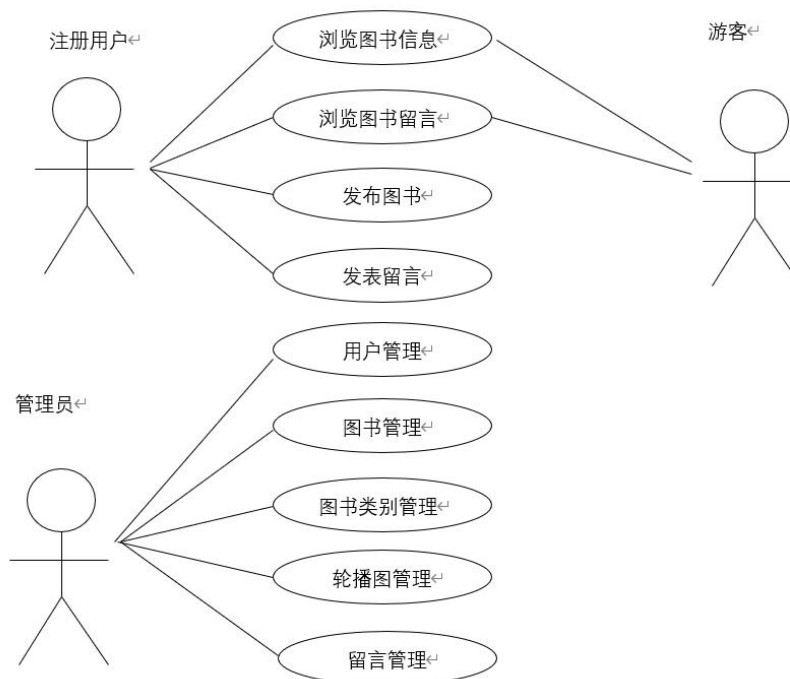


图 3.1 系统用户用例图

3.5 开发环境

运行环境配置从硬件和软件两个方面描述,所列出的都是系统开发时的软件硬件信息。更好的配置和软件环境将会带来更快的运行速度以及更稳定的运行效果。

硬件要求:

CPU: Intel (R) Core (TM) i5-8300H CPU @2.30GHz 2.30GHz

内存：8GB

磁盘安装空间：500GB（完全安装包括安装 Java 运行环境）。

软件要求：

操作系统：Microsoft Windows10

开发工具：IntelliJ IDEA 2020.1.1

数据库工具：MYSQL 5.5

4 系统的设计

4.1 系统功能结构

二手书买卖系统共分为前台和后台两个大页面，主要功能如下：

(1) 前台功能介绍：

用户访问二手书买卖系统首页面后，可进行查看各个图书类别下的图书信息、搜索图书，查看评论、选择图书类别、用户注册、用户登录等功能。用户在二手书买卖系统通过注册成为该网站的真正用户并成功登录系统后，可以进行图书的购买、发布自己的图书、对喜欢的图书进行评论等操作。

前台功能如图 4.1 系统前台功能模块图所示

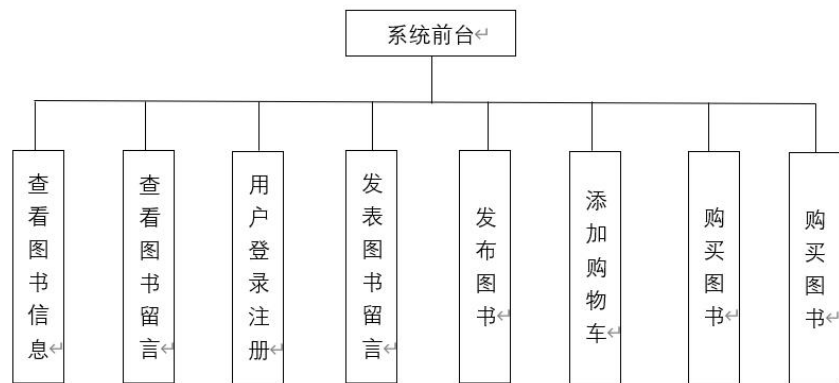


图 4.1 系统前台功能模块图

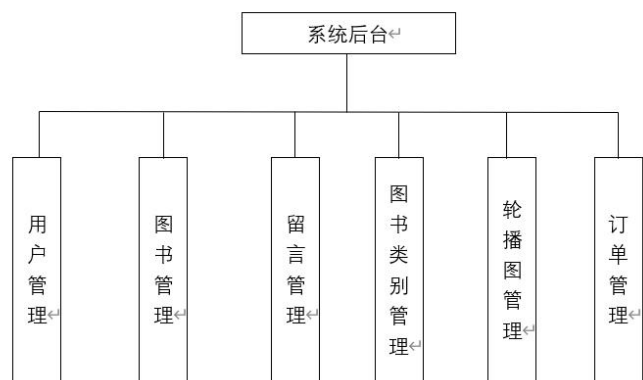


图 4.2 系统后台功能模块图

(2) 后台功能介绍:

若登录的用户权限为管理员，可以进入后台管理，可进行图书信息的管理、用户信息的管理、评论信息的管理等一些需要管理员权限的操作。

后台功能结构如图 4.2 系统后台功能模块图所示。

4.2 系统流程图

(1) 系统前台流程图如图 4.3 系统前台流程图所示。

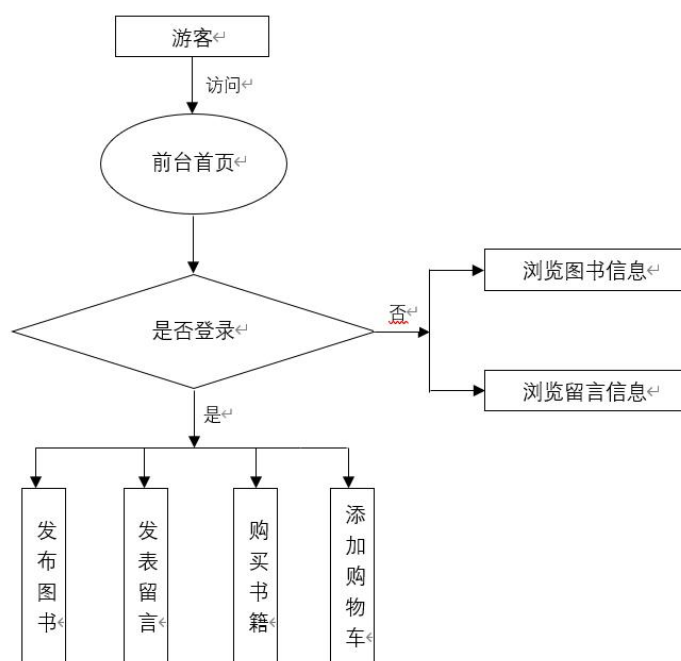


图 4.3 系统前台流程图

(2) 系统后台流程如图 4.4 系统后台流程图所示。

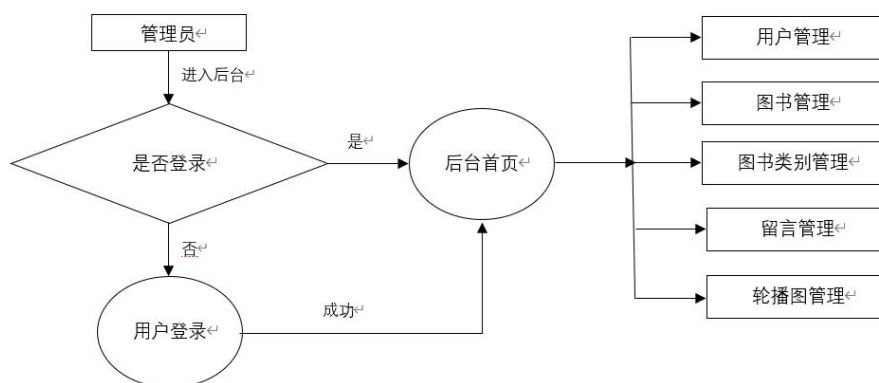


图 4.4 系统后台流程图

4.3 安全性设计

构建安全系统的目的是保护二手书买卖系统中的数据，防止数据被破坏或被盗，维护系统的安全稳定运行。因此，有必要从多个方面为二手教材网上交易系统提供一个完善的安全解决方案。本节将重点介绍数据安全、网络安全和数据备份策略用于构建安全系统。。

(1) 数据安全设计

考虑数据安全目的主要是为了避免一些突发情况，防止数据被破坏。

普通用户：对于普通系统用户，需要实现在用户进行登录时进行登录认证以及系统用户权限控制等多种安全保障方式，比如游客用户只能查看二手图书信息等。

系统管理员：管理员的主要作用是通过自己最高的权限来保护系统数据的安全，在系统出现问题能够解决系统的各种故障。提高系统数据的可靠性以及安全性。一是赋予用户适当的操作权限，二是对数据进行加密，防止密码泄露。

(2) 网络安全设计

物理措施：维护网络中的设备不出现问题，如防火墙、电源断电等问题。

访问控制：通过系统的访问控制实现用户身份认证。

数据加密：使用数据加密保护数据是最常用的数据安全方法。即使加密的内容被非法分子窃取，也不会造成数据的泄露。目前的加密方法有很多，比如 AES 加密和 MD5 加密。数据加密可以有效防止信息泄露。

4.4 详细设计

4.4.1 前台功能模块

(1) 用户注册

注册是一个系统的主要功能，注册时我们可以使用手机号进行用户注册，注册时手机号不能为空，注册时可以填写用户昵称，并且当用户成功注册后，跳转到登录页面进行登录，当注册用户的手机号存在时会提示用户已存在，注册后数据持久化到数据库中，下次登录时会直接从数据库中查询数据。

(2) 用户登录

用户使用注册时的进行登录，登录时会拿到输入的账号密码与数据库中拿到的数据进行对比，当用户账号密码出现不匹配时，会提示用户账号或者密码错误，由于密码采用的是 MD5 加密，所以在进行密码的验证时，通常会把密码先进行 MD5 加密，然后再与数据库传回的进行比较，当用户账号密码正确时，会跳转到系统首页。

(2) 图书展示功能

当用户进入系统后，可以看到系统首页下的图书信息的展示，主要展示图书信息的封面、图书的标题、图书价格等，当用户点击图书信息后，会跳转到图书详细页面，在此页面能够除了以上提到的功能能够查看图书简介等。

(3) 购物车功能

用户有自己喜欢的图书后，可以添加到购物车中，购物车中主要显示当前用户购物车中的信心。主要显示图书封面、图书价格、图书名称等，在此页面用户点击图书封面后能够跳转到图书详情页面进行图书的购买。

(4) 订单生产功能

当用户选择图书，并使用交易地址后，并确认订单信息后，点击立即购买后会跳转到订单生成页面，在此会生成一条订单信息，包括图书 id、用户 id、交易地址 id 等，并存放到数据库中。

(5) 轮播图展示

进入系统首页后，能够看到滚动的轮播图信息，并且用户能够查看下一张轮播

图信息

（6）图书发布功能

图书发布是系统不可少的功能，用户登录后获得发布图书的权限后，能够发布自己的图书，主要包括图书的标题、价格、简介、类别、图书封面等。

4.4.2 后台功能模块

（1）用户管理模块

管理员用户能够对系统的其他用户进行信息管理、在此页面管理员用户可以查看用户信息、可以编辑用户信息、查询用户信息、删除用户信息等。主要展示用户名称、性别、出生日期、用户状态等。

（2）图书管理模块

管理员用户可以对用户发布的图书进行图书管理，包括图书的上架、下架、图书的删除、图书搜索、图书信息的展示、管理员用户能够在添加图书页面进行图书信息的添加。

（3）留言管理模块

管理员用户可以对平常用户对自己喜欢的图书发表的留言进行管理、包括留言信息的删除、留言信息的查询、留言信息的展示等。

（4）图书类别模块

管理员可以对图书类别进行管理，包括对图书类的展示、查询、修改、删除、新增等。

（5）个人中心模块

主要功能是管理员对自己个人信息进行管理，包括个人信息的展示、个人信息的修改、登录密码的修改，用户头像的修改等。

（6）轮播图管理模块

管理员用户可以对系统的轮播图信息进行管理，在此页面包括轮播图信息的展示、轮播图的禁用和开启、轮播图的删除、和新增轮播图等功能。

（7）订单管理模块

管理员用户可以对系统的订单进行管理，包括订单的购买者、订单的发布者、商品图片、商品价格、商品交易地址、订单的状态，订单的删除等功能。

5 数据库的设计

5.1 数据库和设计概述

当一个系统开发经过系统分析就需要根据需求功能对数据库进行设计。在进行数据库设计的过程中，需要我们对需求功能进行理解，充分考虑到各个功能之间的潜在关系设计出合理的数据库表，通过需求分析和功能研究，把自己预期的想法融入到系统设计方案中，进行系统整体的数据库设计。在给定的应用环境中，需要构建最符合生产环境的数据库存储模式，构建相应的数据库，从而达到生产环境能够高效地存储数据，能够方便地实现用户提出的各种使用功能需求，这就是数据库设计。

这种环境的最主要部分是数据库系统，因为所有的页面数据处理最后都在数据库中执行的。本系统使用的数据库是 MySQL。一个合理的数据库设计能够决定一个系统的建设，作为系统和站点的数据中心。从需求分析到系统的逻辑设计，最后到系统的完整实现，是一个符合系统功能设计的数据库，用于整个系统实现设计过程的体现。数据库系统的设计关系数据库模型逻辑层面，根据前面分析的结果，然后设计出人际关系模式为参考，找到或建立主起点为物理数据库设计最合适的物理环境。一般来说，选择一组使用比较频繁的数据库软件，相当于使用根据前面数据模型逻辑设计的数据库软件来构建相应的数据库。

在本系统中的数据库中存储了八张数据表：book 表、bookCategory 表、collection 表、leave_word 表、order_form 表、place_transaction 表、receiving_address 表、user 表、banner 表。

5.2 概念模型和设计（E-R图）

数据库模型是对现实中相同对象的数据特征的抽象，从抽象层次上描述了系统的静态特征、动态行为和约束条件，为数据库系统的信息表与操作提供一个抽象的框架。描述了数据结构、数据操作及数据约束。

E-R 模型的基本概念

ER 图是实体之间的概念模型关系，这将它们与数据库模型图区分开来。一个

ER 图由不同的实体类型、关系、属性和类型组成。实体代表真实的对象，如用户，有时更抽象，但必须有意义。属性用于描述实体，实体之间使用关系。

- (1) 实体：现实世界中的事物；
- (2) 属性：事物的特性；
- (3) 联系：现实世界中的事物间的关系。实体集的关系有一对一、一对多、多对多的联系。

本系统 E-R 图如图 5.1 系统 E-R 图所示。

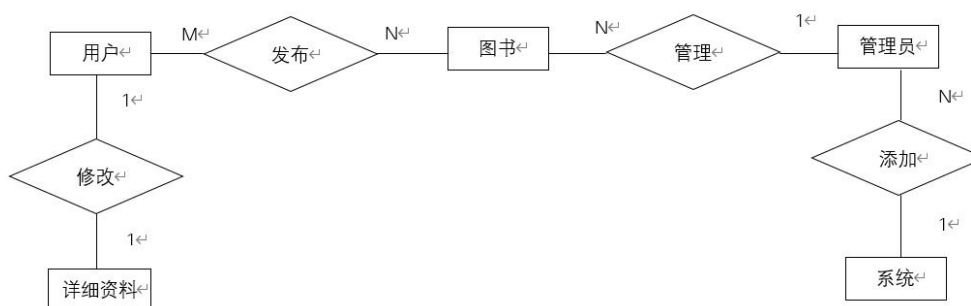


图 5.1 系统 E-R 图

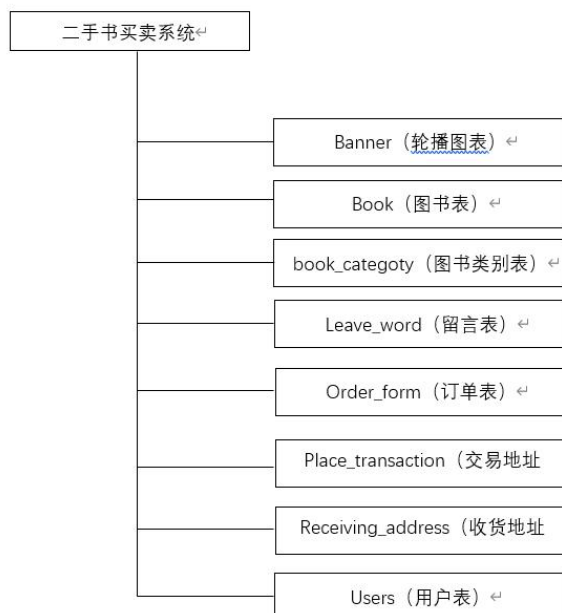


图 5.2 数据库包含数据表结构

5.3 数据库逻辑设计

根据数据库概念设计，可以创建与实体对应的数据表。本系统包含数据库表的数据结构图如图 5.2 数据库包含数据表结构图所示。

本系统共包含个 9 数据表，图 5.2 是表的结构。

- (1) 用户表 `user` 用来存储系统注册的用户和管理员的基本信息，其结构如表 5.1 所示。

表 5.1 用户表结构

字段名称	数据类型	长度	是否主键	是否空值	描述
<code>Id</code>	<code>Int</code>	11	是	否	用户 id
<code>Mobile</code>	<code>Varchar</code>	11	否	否	用户手机号
<code>User_name</code>	<code>Varchar</code>	30	否	否	用户名
<code>User_student_id</code>	<code>Varchar</code>	255	否	是	学号
<code>User_real_name</code>	<code>Varchar</code>	30	否	是	用户真实姓名
<code>User_sex</code>	<code>Char</code>	2	否	是	用户性别
<code>User_birthday</code>	<code>Date</code>		否	是	出生年月
<code>User_pwd</code>	<code>Varchae</code>	50	否	否	用户登录密码
<code>User_tel</code>	<code>Varchar</code>	50	否	是	用户电话
<code>User_email</code>	<code>Varchar</code>	50	否	是	用户邮箱
<code>User_pic</code>	<code>Varchar</code>	300	否	是	用户头像
<code>User_role</code>	<code>Int</code>	2	否	否	用户权限
<code>Is_usable</code>	<code>Tinyint</code>	1	否	否	数据是否可用
<code>Cst_create</code>	<code>Datetime</code>		否	否	数据创建时间
<code>Cst_modify</code>	<code>Datetime</code>		否	否	数据修改时间

- (2) 图书表 `book` 用来存储系统用户发表的图书的基本信息，其结构如表 5.2 所示。

- (3) 图书类别表 `book_category` 用来存储图书类别的基本信息，其结构如表 5.3 所示。

- (4) 购物车表 `collection` 用来存储用户购物车中的基本信息，其结构如表 5.4 所示。

- (5) 留言表 `leave_word` 用来存储用户对自己喜欢的图书发表的留言，其结构如表 5.5 所示。

表 5.2 图书表结构

字段名称	数据类型	长度	是否主键	是否为空	描述
Id	Int	11	是	否	图书编号
User_id	Int	11	否	否	用户编号
Book_category_id	Int	11	否	否	图书类别编号
Book_name	Varchar	100	否	是	图书名称
Book_synopsis	Varchar	300	否	是	图书简介
Book_picture	Varchar	300	否	是	图书图片
Book_price	Varchar	7,2	否	是	图书价格
Is_usable	Tinyint	1	否	否	图书是否可用
Cst_create	Datetime		否	否	图书创建时间
Cst_modify	Datetime		否	否	图书修改时间

表 5.3 图书类别表

字段名称	数据类型	是否为空	长度	主键	描述
Id	Int	否	11	是	图书类别编号
Book_category_name	Varchar	否	50	否	图书类别名
Book_category_level	Int	否	2	否	图书类别级别
Is_usable	Tinyint	否	1	否	图书类别是否可用
Book_category_id	Int	否	Int	否	图书类别父类编号
Cst_create	Datetime	否		否	数据创建时间
Cst_modify	Datetime	否		否	数据修改时间

表 5.4 购物车信息表

字段名称	数据类型	是否主键	是否为空	长度	描述
Id	Int	是	否	11	购物车编号
Book_id	Int	否	否	11	图书编号
Is_usable	Tinyint	否	否	1	数据是否可用
User_id	Int	否	否	11	用户编号
Cst_create	Datetime	否	否		数据创建时间
Cst_modify	Datetime	否	否		数据修改时间

表 5.5 留言信息表

字段名称	数据类型	是否主键	是否为空	长度	描述
Id	Int	是	否	11	留言编号
Leave_content	Varchar	否	否	300	留言内容
Leave_father_id	Int	否	否	11	回复留言编号
Book_id	Int	否	否	11	图书编号
Is_usable	Tinyint	否	否	1	数据是否可用
User_id	Int	否	否	11	用户编号
Cst_create	Datetime	否	否		数据创建时间
Cst_modify	Datetime	否	否		数据修改时间

(6) 订单表 order_form 用来存储用户购买的图书订单信息，其结构如表 5.6 所示。

表 5.6 订单信息表

字段名称	数据类型	是否主键	是否为空	长度	描述
Id	Int	是	否	11	订单编号
Pay_user	Int	否	否	11	购买用户编号
Receiver_id	Int	否	否	11	收货地址编号
Order_state	2	否	否		订单状态
Order_remark	Varchar	否	否	300	订单备注
User_id	Int	否	否	11	用户编号
Is_usable	Tinyint	否	否	1	数据是否可用
Book_id	Int	否	否	11	图书编号
Cst_create	Datetime	否	否		数据创建时间
Cst_modify	Datetime	否	否		数据修改时间

(7) 交易地点表 place_transaction 用来存储图书的交易地点信息，其结构如表 5.7 所示。

表 5.7 交易地点表

字段名称	数据类型	是否主键	是否为空	长度	描述
Id	Int	是	否	11	交易点编号
Place_level	Int	否	否	2	交易地点级别
Place_father_id	Int	否	否	11	交易地点父类编号
Is_usable	Tinyint	否	否	4	数据是否可用
Place_name	Varchar	否	否	50	交易地点名称
Cst_create	Datetime	否	否		数据创建时间
Cst_modify	Datetime	否	否		数据修改时间

(8) 收货地址表 `receiving_address` 用来存储图书收货地址的基本信息，其结构如表 5.8 所示。

表 5.8 收货地址表

字段名称	数据类型	是否主键	是否为空	长度	描述
Id	Int	是	否	11	收货地址编号
Place_name	Varchar	否	否	50	交易地点
Receiver_tel	Varcahr	否	否	30	收货人电话
Is_usable	Tinyint	否	否	1	数据是否可用
Cst_create	Datetime	否	否		数据创建时间
User_id	Int	否	否	11	用户编号
Cst_modify	Datetime	否	否		数据修改时间
Receiver_name	Varchar	否	否	30	收货人姓名
Is_default	Int	否	否		是否为默认地址

(9) 轮播图信息表 `banner` 用来存储图书收货地址的基本信息，其结构如表 5.9 所示。

表 5.9 轮播图信息表

字段名称	数据类型	长度	是否主键	是否为空	描述
Id	Char	19	是	否	轮播图编号
Title	Varchar	20	否	是	轮播图标题
Image_url	Varchar	500	否	否	图片地址
Link_url	Varchart	500	否	是	连接地址
Sort	Int	10	否	是	排序
Is_usable	Tinyint	1	否	否	数据是否可用
Cst_create	Datetime		否	否	数据创建时间
Cst_modify	Datetime		否	否	数据修改时间

6 系统的实现

6.1 网站前台页面

6.1.1 系统首页

首页是一个系统的最主要的组成部分，是一个系统功能的最主要的体现，二手书买卖系统的首页主要有系统轮播、图书类别展示、图书分页展示、用户登录展示、用户登录后可以展示用户的头像和用户名信息、图书搜索、发布图书、个人信息、我的购物车等组成，界面简单优雅，系统功能完善，系统首页图如图 6.1 所示。



图 6.1 系统首页图

实现代码：

```
public R getAllBookList(@RequestBody QueryPageBean queryPageBean) {  
    Integer currentPage = queryPageBean.getCurrentPage();  
    Integer pageSize = queryPageBean.getPageSize();  
    String queryString = queryPageBean.getQueryString();  
    Book book = new Book().setBookName(queryString);
```

```
String bookName = book.getBookName();
wrapper.like("book_name",bookName);
wrapper.eq("is_usable",1);
Page<Book> page = new Page<>(currentPage, pageSize);
bookService.page(page, wrapper);
List<Book> bookList = page.getRecords();
long total = page.getTotal();
return R.ok().data("bookList", bookList).data("total", total);
}
```

6.1.2 用户登录和注册

二手书买卖系统分为普通用户和管理员用户，普通用户登录后进入前台页面，管理员用户登录后进入后台系统，普通用户登录时可以通过注册的手机号进行登录，管理员用户可以通过自己注册的手机号进行登录。用户登录详细如图 6.2 系统登录图所示，用户注册如图 6.3 用户注册图所示。

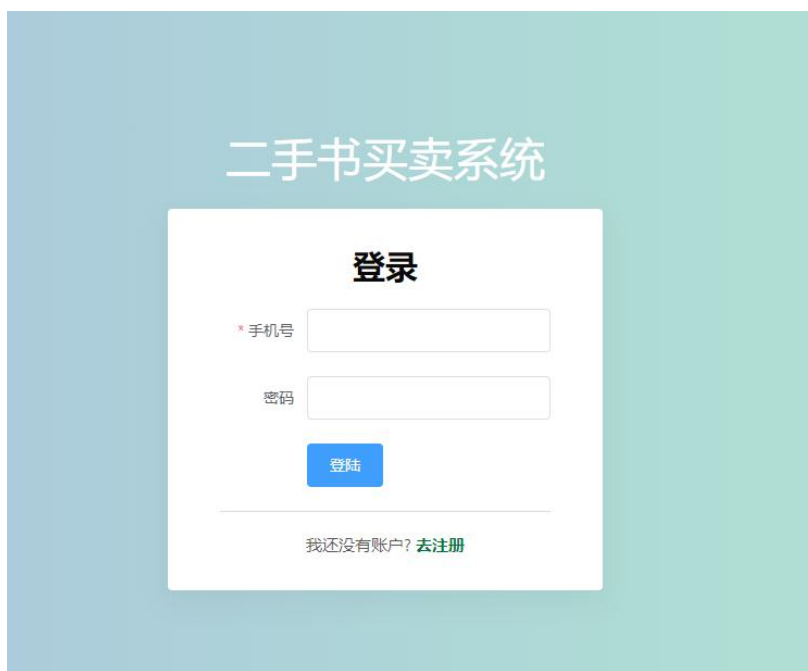


图 6.2 系统登录

用户登录成功后，后台会返回一个 token 字符串，前端拿到 token 后把 token 保存在 cookie 中，根据 token 获取用户信息后，把用户信息存到 cookie 中，这样当在其他页面需要获取用户信息时，就不再需要发送请求，直接在 cookie 拿到用

户信息，而在进行其他操作的还可以根据 token 来判断当前用户的状态信息。

图 6.3 系统注册

当用户进行注册时，后台拿到手机号后，会先查询当前手机号是否已经是系统用户，当手机号已经是系统用户，会提示用户已注册并跳转到登录页面。当用户还未注册时则会把输入的密码进行 MD5 加密，并把用户信息存储到数据库里。

实现代码：

//用户登录

```
public String login(loginVo loginVo) {  
    String password = loginVo.getPassword();  
    String mobile = loginVo.getMobile();  
    //验空  
    if (StringUtils.isEmpty(mobile) || StringUtils.isEmpty(password)) {  
        throw new BookShopException(20001, "error");  
    }  
    //验证用户是否存在  
    wrapper.eq("mobile", mobile);  
    Users users = baseMapper.selectOne(wrapper);  
    if (users == null) {  
        throw new BookShopException(20001, "请注册后再登录！");  
    }  
}
```



```
    }
    if (!MD5.encrypt(password).equals(users.getUserPwd())) {
        throw new BookShopException(20001, "密码错误");
    }
    //检验是否被禁用
    if (!users.getIsUsable()) {
        throw new BookShopException(20001, "error");
    }
    //使用 JWT 生成 token 字符串
    String token = JwtUtils.getJwtToken(users.getId(), users.getUserName());
    return token;
}

public void register(registerVo registerVo)
//    String userName = registerVo.getUserName();
    String mobile = registerVo.getMobile();
    String password = registerVo.getPassword();
    String username = registerVo.getUserName();
//    StringUtils.isEmpty(userName) ||
    //数据检验
    if (
        StringUtils.isEmpty(mobile) ||
        StringUtils.isEmpty(password)) {
        throw new BookShopException(20001, "不能为空");
        //查询条件构造器，查询手机号
        wrapper.eq("mobile", mobile);
        Users users = baseMapper.selectOne(wrapper);
        if (users != null) {
            throw new BookShopException(20001, "用户已存在，请登录");
        }
        Users users1 = new Users();
        users1.setMobile(mobile);
        String encrypt = MD5.encrypt(password);
        users1.setUserPwd(encrypt);
        users1.setIsUsable(true);
        users1.setUserName(username);
        baseMapper.insert(users1);
    }
```

}

6.1.3 用户发布图书

游客用户登录系统后可以使用系统的发布图书功能，发布自己闲置的二手书，此页面用户可以设置自己发布书籍的名称、价格、简介、图书封面、图书类别等，详细的介绍图书信息，在此页面用户选择图书封面后，系统会把封面上传到阿里云 OSS，并返回一个地址，当用户点击发布图书时，就会把封面地址保存到数据库中。详细如图 6.4 发布图书所示。

图 6.4 发布图书

实现代码：

//添加图书

```
public Integer addBook(BookVo bookVo) {  
    bookVo.setIsUsable(true);  
    Book book = new Book();  
    BeanUtils.copyProperties(bookVo,book);
```

```
int insert = baseMapper.insert(book);
return insert;
}
public R updatePass(@RequestBody PassWordVo passWordVo) {
    Integer usersId = passWordVo.getId();
    String oldUserPwd = passWordVo.getOldUserPwd();
    Users byId = usersService.getById(usersId);
    String userPwd = byId.getUserPwd();
    String encrypt = MD5.encrypt(oldUserPwd);
    if (!encrypt.equals(userPwd)){
        throw new BookShopException(20001,"请输入原正确密码");
    }
    String newUserPwd = passWordVo.getNewUserPwd();
    String newEncrypt1 = MD5.encrypt(newUserPwd);
    byId.setUserPwd(newEncrypt1);
    usersService.updateById(byId);
    return R.ok();
}
```

6.1.4 购物车

游客用户登录系统成为系统用户后，浏览自己需要的书籍时，可以把自己喜欢的书籍加入自己的购物车，等到确定购买书籍之后，可以在购物车找到该书籍，继续购买。在此页面主要展示图书封面、图书价格、图书简介等。详细如图 6.5 购物车页面所示。



图 6.5 购物车页面

当用户进入到购物车页面时先查询当前用户的信息，根据用户信息查询当前用户购物车里的图书信息，并返回到前端进行页面渲染。**Collection** 是购物车的数据模型，记录购物车中的图书订单信息，**collectionVo** 是购物车的视图模型。

实现代码

```
public R getAllCollection(@PathVariable Integer id){
    QueryWrapper<Collection> wrapper = new QueryWrapper<>();
    wrapper.eq("user_id",id);
    List<Collection> list = collectionService.list(wrapper);
    List<CollectionVo> collectionVoList = new ArrayList<>();
    for (int i = 0; i < list.size(); i++) {
        Collection collection = list.get(i);
        Integer bookId = collection.getBookId();
        Book book = bookService.getById(bookId);
        CollectionVo collectionVo = new CollectionVo();
        BeanUtils.copyProperties(book,collectionVo);
        collectionVoList.add(collectionVo);
    }
    return R.ok().data("collectionVoList",collectionVoList);
}
```

6.1.5 留言功能

当用户登录系统后，可以获得发表图书留言的权限，用户可以在自己喜欢的图书下与其他用户一起讨论书籍，发表自己的看法，可以输入文本，插入图片，添加表情等，详细如图 6.6 图书详细信息图所示。



图 6.6 图书详细信息

```
public R addLeaveWord(@RequestBody leaveWordVo leaveWordVo) {
    LeaveWord leaveWord = new LeaveWord();
    BeanUtils.copyProperties(leaveWordVo, leaveWord);
    leaveWordService.save(leaveWord);
    return R.ok();
}

public R findLeaveWorld(@RequestBody QueryPageBean queryPageBean){
    Integer currentPage = queryPageBean.getCurrentPage();
    Integer pageSize = queryPageBean.getPageSize();
    String queryString = queryPageBean.getQueryString();
    QueryWrapper<LeaveWord> wrapper = new QueryWrapper<>();
    wrapper.like("leave_content",queryString);
    Page<LeaveWord> page = new Page<>(currentPage,pageSize);
    leaveWordService.page(page,wrapper);
    List<LeaveWord> records = page.getRecords();
    long total = page.getTotal();
    List<leaveWordVo> leaveWordVos = new ArrayList<>();
    for (int i = 0; i < records.size(); i++) {
        leaveWordVo leaveWordVo = new leaveWordVo();
        LeaveWord leaveWord = records.get(i);
        BeanUtils.copyProperties(leaveWord,leaveWordVo);
        Integer userId = leaveWord.getUserId();
    }
}
```

```
Users users = userService.getById(userId);
leaveWordVo.setUserName(users.getUserName());
leaveWordVos.add(leaveWordVo);
}
return R.ok().data("leaveWordVos",leaveWordVos).data("total",total);
}
```

6.1.6 轮播图展示

用户进入系统后能够在首页查看系统的轮播图信息，并且能够操作轮播图进行图片的切换等，详细如图 6.7 轮播图信息所示。



图 6.7 轮播图信息

6.2 后台管理页面

6.2.1 管理员用户登录

二手书买卖系统是一个前后台分离的一个系统，管理员用户可以通过自己的登录门户登录后进入后台管理页面。详细如图所示 6.8 管理员登录所示。

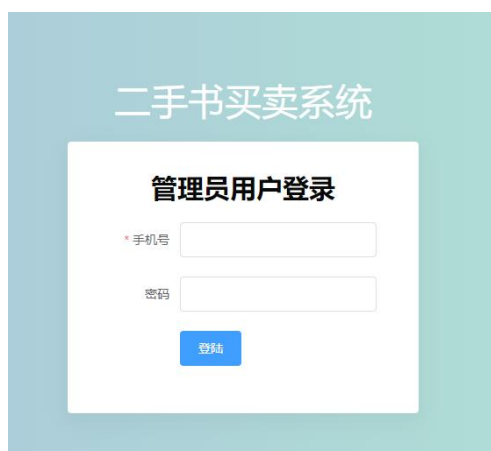


图 6.8 管理员登录图

管理员用户进行登录时，会校验用户权限，当用户权限为管理员时才会跳转到后台页面。

代码实现

```
public String adminLogin(loginVo loginVo) {
    String password = loginVo.getPassword();
    String mobile = loginVo.getMobile();
    //验空
    if (StringUtils.isEmpty(mobile) || StringUtils.isEmpty(password)) {
        throw new BookShopException(20001, "error");
    }
    //验证用户是否存在
    QueryWrapper<Users> wrapper = new QueryWrapper<>();
    wrapper.eq("mobile", mobile);
    Users users = baseMapper.selectOne(wrapper);
    if (users == null) {
        throw new BookShopException(20001, "请注册后再登录！");
    }
    if (users.getUserRole() != 1) {
        throw new BookShopException(2001, "请使用管理员账号登录！");
    }
    if (!MD5.encrypt(password).equals(users.getUserPwd())) {
        throw new BookShopException(20001, "密码错误");
    }
    //检验是否被禁用
    if (!users.getIsUsable()) {
        throw new BookShopException(20001, "error");
    }
    //使用 JWT 生成 token 字符串
    String token = JwtUtils.getJwtToken(users.getId(), users.getUserName());
    return token;
}
```

6.2.2 用户管理

当管理员登录系统后可以进入用户管理页面，在此页面可以进行用户查询修改、删除等操作，还可以查询管理员用户信息等。主要展示用户名称、用户性别、

出生日期、用户邮箱、用户状态等。详细界面如图所示 6.9 用户管理页面所示。

用户名称		查询				
序号	会员名	性别	出生日期	用户邮箱	用户状态	操作
1	张三11	男	1996-06-21 00:00:00	123456789@qq.com	可用	编辑 删除
2	李四111	男		1562543587@qq.com	可用	编辑 删除
3	王五	男		4544578878@qq.com	可用	编辑 删除
4	杨柳	女		123542584@qq.com	可用	编辑 删除
共 4 条 < 1 > 前往 1 页						

图 6.9 用户管理页面

代码实现

```
public R getAllUserList(@RequestBody QueryPageBean queryPageBean){
    Integer currentPage = queryPageBean.getCurrentPage();
    Integer pageSize = queryPageBean.getPageSize();
    String queryString = queryPageBean.getQueryString();
    Users users = new Users();
    QueryWrapper<Users> wrapper = new QueryWrapper<>();
    wrapper.like("user_name",queryString);
    wrapper.ne("user_role",1);
    Page<Users> page = new Page<>(currentPage, pageSize);
    usersService.page(page,wrapper);
    List<Users> userList = page.getRecords();
    long total = page.getTotal();
    return R.ok().data("usersList",userList).data("total",total);
}
```

6.2.3 图书信息管理

管理员用户登录后台管理系统后，可以进入图书管理页面，包括图书的添加、图书信息的查询、修改图书状态等一些列操作，主要展示图书名称、图书封面、图书价格、图书状态等主要页面信息。具体详情如图所示 6.10 图书管理页面所示

6	企业网站建设及应用实训教程		57.5	可用	下载图书 删除
7	建筑CAD基础教程(中等职业教育“十二五”规划教材)		29.5	可用	下载图书 删除
8	“十三五”职业教育规划教材 计算机组装与维护项目化教程		29.1	可用	下载图书 删除
9	南明史		129	可用	下载图书 删除
10	思维模型		34	可用	下载图书 删除
11	我的第一套中国神话		154.4	可用	下载图书 删除

图 6.10 图书管理页面

代码实现

```
public R getAllBook() {
    List<Book> bookList = bookService.list(null);
    return R.ok().data("bookList", bookList);
}
```

6.2.4 留言信息管理

管理员用户进入此页面后，可以查看所有的留言信息、包括搜索留言信息、留言的发布者、发布时间、对不合规留言信息进行删除等操作，主要展示留言内容、留言发布人、发布时间等页面信息。详细信息如图 6.11 留言信息管理页面所示。

用户名称留言内容		查询			
序号	留言内容	用户名	留言时间	操作	
1	图片清晰吗，是正版吗	张三11	2019-04-28 12:11:01	删除	
2	这个对考教师资格证有用吗	张三11	2019-04-28 12:11:29	删除	
3	这个适合零基础的人学吗	张三11	2019-04-28 12:12:25	删除	
4	你是不是有病	张三11	2019-04-28 12:12:47	删除	
5	你好	李四111	2019-05-05 04:18:54	删除	
6	图书怎么样	柳柳	2019-05-05 04:20:43	删除	
7	发货单看	李四111	2019-05-05 13:35:10	删除	
8	很快就到饭	王五	2019-05-05 14:08:45	删除	
9	黄金客户即可	柳柳	2019-05-05 15:09:37	删除	
10		张三11	2019-05-06 15:17:26	删除	

图 6.11 留言信息管理页面

代码实现

```
public R getLeaveWorldByBookId(@PathVariable int id) {  
    wordQueryWrapper.eq("book_id", id);  
    List<LeaveWord> leaveWordList = leaveWordService.list(wordQueryWrapper);  
    List<leaveWordVo> leaveWordVos = new ArrayList<>();  
    for (int i = 0; i < leaveWordList.size(); i++) {  
        LeaveWord leaveWord = leaveWordList.get(i);  
        leaveWordVo leaveWordVo = new leaveWordVo();  
        Integer userId = leaveWord.getUserId();  
        Users user = userService.getById(userId);  
        BeanUtils.copyProperties(leaveWord, leaveWordVo);  
        BeanUtils.copyProperties(user, leaveWordVo);  
        leaveWordVos.add(leaveWordVo);  
    }  
    return R.ok().data("leaveWordVos", leaveWordVos);  
}
```

6.2.5 图书类别信息管理

管理员用户有权限对图书管理进行修改，包括图书类别的查看、新增、修改、删除等操作。详细信息如图 6.12 图书类别管理页面所示。

图书类别			
<input type="text" value="图书类别"/>		<input type="button" value="查询"/>	<input type="button" value="新建"/>
序号	图书类别	类别级别	操作
1	考试教育	一级	<input type="button" value="编辑"/> <input type="button" value="删除"/>
2	经济管理	一级	<input type="button" value="编辑"/> <input type="button" value="删除"/>
3	文学艺术	一级	<input type="button" value="编辑"/> <input type="button" value="删除"/>
4	人文社科	一级	<input type="button" value="编辑"/> <input type="button" value="删除"/>
5	科学技术	一级	<input type="button" value="编辑"/> <input type="button" value="删除"/>
6	生活休闲	一级	<input type="button" value="编辑"/> <input type="button" value="删除"/>
7	公务员考试	二级	<input type="button" value="编辑"/> <input type="button" value="删除"/>
8	英语考试	二级	<input type="button" value="编辑"/> <input type="button" value="删除"/>
9	考研	二级	<input type="button" value="编辑"/> <input type="button" value="删除"/>
10	职称考试	二级	<input type="button" value="编辑"/> <input type="button" value="删除"/>
共 37 条 < 1 2 3 4 > 前往 1 页			

图 6.12 图书类别管理页面

代码实现：

```
public R findAllSubject() {
```

```

        List<oneCategoryVo>categoryList=bookCategoryService.getAllCateGory();
        return R.ok().data("categoryList", categoryList);
    }

```

6.2.6 管理员个人信息管理

管理员用户在此页面可以进行个人信息的查看、修改、还可以对用户登录密码进行修改，主要展示用户头像、用户名称、用户电话、用户邮箱、出生年月等页面信息。详细页面如图 6.13 用户信息详细信息页面所示

图 6.13 用户详细信息页面

代码实现

```

public R updateUserById(@RequestBody usersVo usersVo) {
    Users users = new Users();
    BeanUtils.copyProperties(usersVo, users);
    boolean b = userService.updateById(users);
    if (!b) {
        throw new BookShopException(20001, "修改失败");
    }
    return R.ok();
}

```

6.2.7 轮播图信息管理

管理员用户在此页面可以对系统首页轮播图图片信息进行管理，包括对轮播

图的禁用、添加等功能，详细页面如图 6.14 轮播图管理页面所示。


新建		
序号	轮播图图片	操作
1		<button>禁用</button> <button>删除</button>
2		<button>禁用</button> <button>删除</button>
3		<button>禁用</button> <button>删除</button>
4		<button>禁用</button> <button>删除</button>
共 4 条 < 1 > 前往 1 页		

图 6.14 轮播图管理页面

代码实现

```
public R getAllBanners(@RequestBody QueryPageBean queryPageBean){
    Integer pageSize = queryPageBean.getPageSize();
    Integer currentPage = queryPageBean.getCurrentPage();
    Banner banner = new Banner();
    Page<Banner> page =new Page<>(currentPage,pageSize);
    bannerService.page(page,null);
    List<Banner> records = page.getRecords();
    long total = page.getTotal();
    return R.ok().data("records",records).data("total",total);
}
```

7 系统测试

7.1 测试用例

在系统开发过程中对系统进行了一些功能测试，主要采用黑盒测试方法，对系统已有功能进行测试。

7.1.1 用户注册功能测试用例

用户管理模块选择用户注册进行测试展示，用户注册是系统最基础的模块，也是成为系统用户的唯一途径，未注册情况下，用户只能通过游客身份进行图书信息的浏览，注册成功以后就可以进行图书的购买，图书发布、留言的发布等正常的操作，管理员登录后可以进行用户信心的管理，图书信息的管理、轮播图的管理、留言信息的管理等，本系统对用户输入手机号、密码等各种情况进行了详细的测试，测试用例如下表 7.1 所示。

7.1.2 发布图书测试用例

图书管理模块选择图书发布功能进行测试展示，图书发布是用户在本系统中的核心功能，方便用户发布自己的二手书籍，本系统可以通过图书标题、图书价格、图书封面、图书简介等信息发布图书，对于图书信息本系统进行了严格的匹配，不允许图书名称、图书价格、等出现空的情况，而图书封面在用户没有上传的情况下，系统既定有默认封面。其测试用例如表 7.2 所示。

7.1.3 图书分类测试用例

图书分类使用户在自己上传图书或者搜索图书时用的功能，用户可以根据图书分类功能，在发布时选择相应的分类，或者在搜索图书时选择相应的分类下的图书。测试用例表如下表 7.3 所示。

7.1.4 留言功能测试用例

图书留言是为了方便用户之间更好的交流的一种方式，主要是为了让用户能

够知道图书的详细信息，本系统的图书留言功能只能是系统登录后的用户和管理员才能使用的，游客用户只能查看图书留言信息。详细用例如表 7.4 所示。

表 7.1 用户注册功能测试用例表

项目编号	A01		项目名称		系统用户注册测试	
测试类型	功能测试		测试人员	代春辉	测试日期	2022-04-05
模块编号	模块功能：注册新用户					
	编号	测试用例描述	操作过程及数据		预期结果	实际结果
用户注册	AO1O1	输入使用正常用户数据	1.用户名： 15225764423, 2.用户名为：张三 3.密码：123456 4.点击注册		正常注册用户	无错误
	AO1O2	添加用户账号填写是否为空	1.用户名：空 2.用户名为：张三 3.密码：123456 4.点击注册		不能注册用户	无错误
	A0103	添加重复用户	1.账号 15225764423 2.用户名为：张三 3.密码：123456 4.点击注册		不能注册用户	无错误
	A0104	添加账户为中文	1.用户名：张三， 2.用户名为：张三 3.密码：123456 4.点击注册		不能注册用户	无错误
	A0105	输入密码为空	1.账号：15225764424 2.用户名为张三 3.密码为空 4.点击注册		不能注册用户	无错误
	A0106	账户输入为非法字符	1.账号：1522576442# 2.用户名张三 3.密码：123456 4 点击注册		不能注册用户	缺陷报告 A00-1
	A0107	删除用户	1.选择需要删除的用户 2.点击删除用户		用户被删除	无错误
	A0108	修改用户	1.选择要修改的用户 2.修改用户信息 3.点击保存		用户信息修改成功	无错误

表 7.2 图书发布功能测试用例表

项目编号	A02		项目名称		发布图书测试	
测试类型	功能测试		测试人员	代春辉	测试日期	2022-04-05
模块编号	模块功能：图书发布					
	编号	测试用例描述	操作过程及数据		预期结果	实际结果
帖子发表	A0201	正常数据	1.图书名称为：活着 2.价格输入为：29.9 3.类别选择文学艺术 4.图书简介输入正常数据 5.选择图书封面 6.立即添加		能发布图书	无错误
	A0202	图书标题为空	1.图书名称为空 2.价格为：29.9 3.类别选择文学艺术 4.图书简介输入正常数据 5.选择图书封面 6.立即添加		不能发布图书	缺陷报告 A001-2
	A0203	图书价格为空	1.图书名称输入为活着 2.价格而输入为空 3.类别选择文学艺术 4.图书简介输入正常数据 5.选择图书封面 6.立即添加		不能发布图书	无错误
	A0204	图书的查询	1.输入框输入图书的名称 2.点击查询		查询成功	无错误
	A0205	图书上架	1.选择要上架的图书 2.点击上架图书		上架成功	无错误
	A0206	图书的下架	1.选择要下架的图书 2.点击下架图书		下架成功	无错误
	A0207	图书详细信息查看	1.选择要查看的图书 2.点击图书进入图书详情页面		图书信息出现	无错误

表 7.3 图书类别测试用例表

项目编号	A03		项目名称		图书类别功能测试	
测试类型	功能测试		测试人员	代春辉	测试日期	2022-04-05
模块编号	模块功能：图书类别管理					
	编号	测试用例描述	操作过程及数据		预期结果	实际结果
板块功能	A0301	正常数据	1.输入图书类别名称 程序开发 2.输入板块介绍 3.点击确定		能添加新的图书类别	无错误
	A0302	图书分类标题为空	1.图书分类标题输入为空 2.点击确定		不能添加新的图书类别	缺陷报告 A001-3
	A0303	图书类别的修改	1.选择修改的图书类别 2.输入修改后的图书类别标题 3.点击修改		能修改图书类别	无错误
	A0304	图书类别的删除	1.选择要删除的图书类别 2.点击删除图书类别		图书类被删除	无错误

表 7.4 图书留言功能测试用例表

项目编号	A04		项目名称		留言测试	
测试类型	功能测试		测试人员	代春辉	测试日期	2022-04-05
模块编号	模块功能：图书留言信息的查看、删除					
	编号	测试用例描述	操作过程及数据		预期结果	实际结果
留言功能	A0401	正常数据	1.输入留言信息 2.点击发表留言		能够进行图书信息留言功能	无错误
	A0402	输入留言信息为空	1. 输入留言信息内容为空 2. 点击发表留言		不能进行留言反馈	缺陷报告 A001-4
	A0403	删除留言信息	1.选择要删除的留言信息 2.点击删除		能删除留言信息	无错误

7.2 测试用例的执行

测试用例执行是系统功能完善的体现，详细如表 7.5 所示。

表 7.5 测试执行表

用例编号	测试通过情况	缺陷编号
A0101	通过	
A0102	通过	
A0103	通过	
A0104	通过	
A0105	通过	
A0106	失败	A001-1
A0107	通过	
A0108	通过	
A0201	通过	
A0202	失败	A001-2
A0203	通过	
A0204	通过	
A0205	通过	
A0207	通过	
A0301	通过	
A0302	失败	A001-3
A0303	通过	
A0304	通过	
A0401	通过	
A0402	失败	A001-4
A0403	通过	

7.3 缺陷报告

表 7.6 缺陷报告 A001-1

缺陷编号	A001-1	缺陷类型	界面缺陷
缺陷位置	用户注册功能	缺陷状态	新建
严重性	严重错误	优先级	高
缺陷概述	用户账号出现特殊字符		
缺陷描述	<p>操作步骤：</p> <p>1.账号：1522576442#</p> <p>2.用户名：张三</p> <p>3.密码：123456</p> <p>4.点击注册</p> <p>预期结果：</p> <p>提示：不能添加用户提示账号错误</p> <p>实际结果：</p> <p>提示：能注册用户</p> <div data-bbox="710 978 1145 1417"></div> <p>图 7.1</p>		

表 7.7 缺陷报告 A001-2

缺陷编号	A001-2	缺陷类型	界面缺陷
缺陷位置	图书发布功能	缺陷状态	新建
严重性	严重错误	优先级	高
缺陷概述	发布图书时，图书标题可以为空		
缺陷描述	<p>操作步骤：</p> <ol style="list-style-type: none">1.图书名称输入为空2.价格而输入为：29.93.类别选择文学艺术4.图书简介输入正常数据5.选择图书封面6.立即添加 <p>预期结果：</p> <p>提示：不能发布图书，提示请输入图书标题</p> <p>实际结果：</p> <p>提示：能够发布图书</p> <div><div>图书名称:</div><div></div><div>价格:</div><div>29.9</div><div>类别:</div><div>文学艺术</div><div>图书简介:</div><div>文学艺术</div><div>图书封面</div><div></div><div>立即添加</div></div> <p>图 7.2</p>		

表 7.8 缺陷报告 A003-1

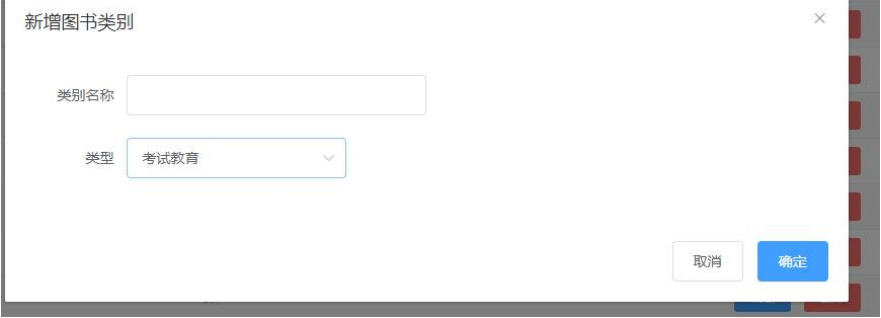
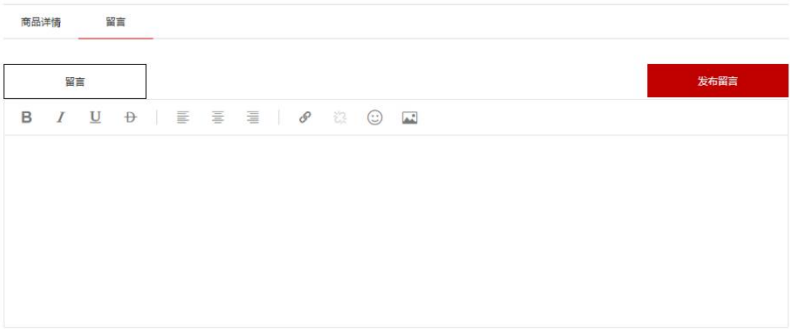
缺陷编号	A003-1	缺陷类型	界面缺陷
缺陷位置	图书类别管理功能	缺陷状态	新建
严重性	严重错误	优先级	高
缺陷概述	新建图书类别时，图书类别名称可以为空		
缺陷描述	<p>操作步骤：</p> <p>1.点击新增按钮</p> <p>2.图书类别名称输入为空</p> <p>3.点击确定按钮</p> <p>预期结果：</p> <p>提示：不能添加图书类别提示图书类别名称不能为空</p> <p>实际结果：</p> <p>提示：能够添加图书类别</p> <div></div>		

图 7.3

表 7.9 缺陷报告 A004-1

缺陷编号	A004-1	缺陷类型	界面缺陷
缺陷位置	留言管理功能	缺陷状态	新建
严重性	严重错误	优先级	高
缺陷概述	用户对图书留言功能可以为空		
缺陷描述	<p>操作步骤：</p> <p>1. 输入留言信息内容为空</p> <p>2. 点击发表留言</p> <p>预期结果：</p> <p>提示：用户不能发表图书留言信息</p> <p>实际结果：</p> <p>提示：用户能发表图书留言信息</p> <div></div> <p>图 7.4</p>		

结 论

随着教育水平的提高，书籍是获取知识的一大途径，市场上的书籍种类日渐丰富，书籍需求庞大，树木作为书纸的主要制造来源，制作纸张需要大量的纸浆，对资源的需求是非常大，而对待使用过的书籍，很多人都是丢弃，造成资源的浪费。

该设计实现对于二手书回收利用的途径上有重大意义。本文研究的关键技术主要有，怎样建立一个用户与用户之间，能够有效交流，交易书籍的系统，主要使用 SpringBoot 框架技术和 Mybatis 框架技术对系统进行实现，并且系统采用 B/S 系统架构，使图书交易系统在用户使用上更加的方便。其成果对于促进资源的回收利用具有一定的理论意义。同时对二手书的交易回收有一定的实用价值。

本文主要的研究工作如下：

针对用户出售闲置书籍的问题，在系统中用户可以发布自己的书籍，以供其他用户购买交易。结果表明提供给用户发布书籍的方法，能够提高二手书籍的流动性。

在完成以上研究工作的过程中取得了如下的创新性成果：采用用户与用户之间直接进行交易，解决了用户比较关心的图书来源问题，让用户能够放心购买需要的书籍，极大地提高图书交易的自由性，用户可以选择自己喜欢的书籍，提高了用户书籍交易的积极性。达到了二手书买卖系统设计的最初目的。

同时在本文研究工作的基础上，还可以在以下方面进行进一步的研究：可以把用户不需要书籍回收到一个地区固定的仓库，在其他用户需要的时候，我们可以快速的把书籍送到需要书籍的用户身边，这样极大地缩短了书籍交易的时间周期，使书籍交易更加快速高效，也提高了书籍交易的安全性，进一步使二手书交易更加规范化，高效化。

致 谢

在此论文完成之际，首先要感谢刘芳云老师给予我论文写作期间的帮助，正是由于您孜孜不倦的态度和实事求是的精神时刻感染着我，让我能够发现问题，并解决问题，能够顺利的完成这篇论文，能遇见这样的老师是我的荣幸。

其次，感谢在我论文完成期间帮助过我的曹同学、魏同学、崔同学、毛同学、黄同学，是他们在我们遇见问题时给予我帮助，让我有了解决问题的思路和方法。同时也感谢我的大学同学和大学老师，是他们让我的大学生活丰富多彩充满阳光，积极向上，是他们给予了我前进的方向和动力。感谢我的母校，在此求学的时光里让我学习了很多，成长了很多。

最后，感谢我的父母，在过去的求学时光里，是他们无私的奉献和支持，让我能够完成如今的学业，让我能有一个充满希望的未来。

行文至此，心绪万千，再回首仿若隔日，两年时光，匆匆而过，依然记得踏进校门那一刻的喜悦，还记得认识新同学时心中的高兴，不会忘记和室友坐在一起谈论学习和未来。如今即将离开校园，心中既有对未来的期盼也有离别的忧伤，期盼的是我们经过大学的学习，习得一身本领，可以去闯荡属于自己的一片天地，忧伤的是和一起生活两年的同学分别，心中难免有些不舍。快乐的时光总是短暂的，是故事总会有结局，结局有好有坏，但是人生只有向前。在未来的日子里，我会努力奋斗，奋起拼搏，完成自己的人生目标。

在此，衷心祝福我们，愿我们能在以后的日子里熠熠生辉。

参考文献

- [1] Yan H H , Jian-Lin Y E . Design and Implementation of Android side of Second-hand Trading Platform of "Two Goods" Campus[J]. Computer Knowledge and Technology, 2019.
- [2] 段欣园,覃一崙,刘 巧,闫艺如,王承宇,黄学军.大学生二手书购买共享平台的建立及调查分析[D]. 乐山师范学院, 2022.02.25
- [3] Liu Y , Huang H , Hu X . Design and implementation of a campus second-hand trading platform based on ThinkPhp[J]. Computer Era, 2018.
- [4] 朱运乔. 基于 SpringBoot+SSM 框架的 Web 应用系统搭建与实现[D]. 湖北职业技术学院, 2019.10.
- [5] 邱丹萍. 应用 SpringBoot 的食堂订餐管理系统设计[D]. 广东白云学院大数据与计算机学院, 2020.06.
- [6] 李孟津, 杨丹. 基于 SpringBoot 的在线招聘网站的设计与实现[J].科学技术创新, 2020.
- [7] 刘子凡,郭昱君. 基于 SpringBoot+Mybatis 的个人博客系统设计与实现[J]. 四川大学锦城学院计算机与软件学院,2021.08.
- [8] 吴晓珊, 曹旭东, 王森, 魏文龙. 基于 B/S 架构的管理系统软件开发[J]. 中国石油大学(北京)地球物理与信息工程学院, 2019.02.
- [9] 袁申.基于 B/S 三层架构的图书馆值班抽取系统的设计与实现[J]. 陕西师范大学图书馆, 2021.06.10.
- [10] 曹瑜琳, 周春晖, 陈云燕,等. 关于高校二手书交易平台"易书屋"的分析与设计[J]. 科技与创新.
- [11] 高春庚. 基于 Cookie 的会话跟踪技术及应用[J]. 信息记录材料, 2021, 22(10):2.
- [12] 韩宸望. SQL 注入攻击过滤技术研究[D]. 福建师范大学 2018.06.01.
- [13] 贾睿.业务管理系统数据库模型设计[D]. 长春工业大学人文信息学院 2019.04.23.