

蚌埠學院

毕业设计(论文)

基于 SpringBoot 的在线投稿系统的 设计与实现

学 院 : 计算机与信息工程学院

专业 (班级) : 软件工程 (2018 级软件工程 2 班)

作者 (学号) : 潘根 (51802042041)

指导教师 : 王磊 (讲师)

完成日期 : 2022 年 5 月 14 日

蚌埠学院教务处制

目 录

中文摘要.....	1
英文摘要.....	2
1 引言.....	3
1.1 课题背景.....	3
1.2 研究现状.....	3
1.3 系统开发目标.....	4
2 技术介绍.....	5
2.1 开发技术介绍.....	5
2.1.1 Java 介绍	5
2.1.2 B/S 架构	5
2.1.3 SpringBoot 框架	5
2.1.4 MyBatis Plus 框架	5
2.2 开发软件介绍.....	6
2.2.1 IntelliJ IDEA.....	6
2.2.2 MySQL 数据库	6
3 系统需求分析.....	7
3.1 可行性分析.....	7
3.2 需求分析	7
3.2.1 浏览人员主要功能需求分析.....	7
3.2.2 作者主要功能需求分析.....	8
3.2.3 专家主要功能需求分析.....	8
3.2.4 编辑主要功能需求分析.....	8
3.2.5 管理员主要功能需求分析.....	8
4 概要设计.....	10
4.1 系统的功能模块结构图	10
4.2 用例说明	10
4.2.1 浏览人员用例.....	10
4.2.2 作者用例.....	11

4.2.3 专家用例.....	11
4.2.4 编辑用例.....	12
4.2.5 管理员用例.....	13
4.3 数据库设计.....	13
4.3.1 数据库概念结构.....	13
4.3.2 数据库逻辑结构.....	15
5 详细设计.....	24
5.1 注册模块设计流程图.....	24
5.2 作者上传稿件模块设计流程图.....	25
5.3 修改密码模块设计流程图.....	25
5.4 添加类别信息模块设计流程图.....	26
5.5 添加公告模块设计流程图.....	27
5.6 添加编辑模块设计流程图.....	27
6 编码实现.....	29
6.1 开发环境与组织结构.....	29
6.2 登录功能实现.....	30
6.3 文件上传功能.....	31
6.4 文件下载功能.....	32
7 系统测试.....	34
7.1 测试目的.....	34
7.2 测试方法.....	34
7.3 功能模块测试.....	34
7.3.1 登录模块测试.....	35
7.3.2 作者管理模块测试.....	35
7.3.3 专家管理模块测试.....	35
7.3.4 编辑管理模块测试.....	36
7.3.5 系统压力测试.....	36
7.4 测试结论	37
8 结束语.....	38
谢辞.....	39
参考文献.....	40

基于 SpringBoot 的在线投稿系统的 设计与实现

摘要：在互联网迅速发展的时代，众多的软件被开发出来，给用户带来了很大的选择余地，人们越来越追求更加个性化的需求。在这种背景下，出版社只能以员工为中心，把稿件的不断创新作为最重要的竞争手段。基于 SpringBoot 的在线投稿系统采用 Java 技术，所有业务模块均采用浏览器交互方式，使用 MySQL 作为系统数据库，SpringBoot 作为后端开发框架，MyBatis Plus 作为数据持久层。前台页面使用 LayUI 框架，后台页面使用 Vue 框架，开发工具使用 IntelliJ IDEA。整个设计过程中，采用软件工程的“开闭”原则，选用敏捷开发方法、前后端分离方式、B/S 模型与 MVC 架构作为基础，实现了在线投稿系统的开发。经过系统测试，改进了程序的逻辑和代码。同时，确保系统中的所有功能都能正常运行，具有良好的运行体验。经过测试，软件基本达到了在线投稿系统的预期效果。

关键词：投稿系统；Java；MySQL；SpringBoot

Design and implementation of online Submission System Based On SpringBoot

Abstract: In the era of the rapid development of the Internet, a large number of software have been developed, which has brought users a lot of choices. People are increasingly pursuing more personalized needs. In this context, publishing houses can only focus on employees and take the continuous innovation of manuscripts as the most important means of competition. The online submission system based on springboot adopts Java technology, and all business modules adopt browser interaction. MySQL is used as the system database, springboot is used as the back-end development framework, and mybatis plus is used as the data persistence layer. The front page uses layui framework, the back page uses Vue framework, and the development tool uses IntelliJ idea. In the whole design process, the "opening and closing" principle of software engineering is adopted, and the development of online submission system is realized based on agile development method, front and rear end separation method, B / S model and MVC architecture. After system testing, the logic and code of the program are improved. At the same time, ensure that all functions in the system can operate normally and have a good operation experience. After testing, the software basically achieves the expected effect of the online submission system.

Keywords: Submission System; Java; MySQL; SpringBoot

基于 SpringBoot 的在线投稿系统的 设计与实现

1 引言

1.1 课题背景

传统的投稿方式是线下进行。用户需要与出版社进行线下交流。随着网络信息的不断普及，越来越多的出版社热衷于网上发展。传统的线下模式已经不能满足人们的需求。

在当今世界，随着互联网的飞速发展，如何利用互联网创造更简单、更高效的投稿方式是现在主要讨论的问题^[1]。需要在线投稿系统相关网站，一方面，使得作者可以有一个更加便捷、高效的投稿途径，另一方面，管理员可以及时更新和管理在线投稿系统网站的信息，使人们能够轻松获取所需信息。这是一个潜在的趋势，可能会有效地加速在线投稿系统的普及。

1.2 研究现状

经过调查发现，现代人的生活节奏加快，生活压力逐渐增大。网络的发展给人们带来了方便。随着稿件的不断增加，越来越多的出版社开始加入网上在线投稿的浪潮。然而，我国对于在线投稿系统的信息管理效果较低，出错率较高^[2]。因此，大家都在寻找一种更专业投稿系统。

随着稿件信息种类的不断增加，用户需求不断增多，在线投稿系统也不断的得到壮大。基于 SpringBoot 的在线投稿系统主要根据作者、专家、编辑和管理员的实际需要，方便用户利用互联网实现对在线投稿系统的了解、对比等一系列操作，同时让管理者可以通过这个系统对用户实际需求以及各稿件信息有所了解进行操作。设计该系统主要目的是为了方便用户可以有一个更好的平台体验，管理员也可以通过该系统进行更加方便的管理，实现了之前指定好的计划。

1.3 系统开发目标

对于系统的设计，必须确保主界面干净有序，能够吸引人们的眼球，不会产生视觉疲劳。更重要的是，它将给人们带来一种操作简便的直观感觉，从而留住用户去使用，并增加三分热度的持续时间。在系统的后台设计中，采用非常简单有效的技术，以便于开发和将来的维护^[3]。不仅要确保所有功能都能满足用户的需求，而且要确保用户能够通过系统主动实现预期的操作，而管理者可以通过系统简单地了解和管理用户的需求^[4]。为实现这一目的，提出了以下目标。

- (1) 用户可以进行在线投稿、查看最新的稿件信息以及相关资讯；
- (2) 用户可以对比各稿件信息，选择自己所需类型的信息；
- (3) 管理员可以在后台管理各种信息。

2 技术介绍

2.1 开发技术介绍

2.1.1 Java 介绍

Java 是一门面向对象的编程语言。Java 语言最大优势是与平台无关。一次写成，处处运行。跟 C 语言相比，Java 语言更加简洁易懂。Java 中有一个重要反射机制，利用反射，可以获得类中的构造方法、成员变量、成员方法、注解、属性等。另外，Java 还提供了丰富的类库，方便用户操作。Java 语言还具有安全性、多线程、动态性等特点^[5]。

2.1.2 B/S 架构

B/S 结构也有很多特点。可以不用安装任何软件就可进行操作。系统扩展性高，只要有一台有网络的电脑就能使用。由于软件技术的不断成熟，B/S 架构最重要的特点是与浏览器配合，这给软件开发带来了极大的便利，不仅能够降低开发成本，还可以不断增强系统的软件功能^[6]。层层相互独立和表示层是 B/S 结构完成互相连接的主要特征。

2.1.3 SpringBoot 框架

SpringBoot 是 Pivotal 团队开发的一个新框架，旨在简化 Spring 应用程序的初始化设置和开发，开箱即用。与 SSM 相比，SpringBoot 省去了很多配置，开发人员只需进行少量配置甚至无需配置就可运行。SpringBoot 最大的特点是约定大于配置，并且在 SpringBoot 官方中推荐使用 YML 和 Java 注解配置类来进行个性化配置。除此之外，在 SpringBoot 中还内置了 Tomcat，可以直接将应用打成 Jar 包部署到服务器上而无需 War 包，减少了部署时的额外配置。

2.1.4 MyBatis Plus 框架

MyBatis Plus 是由国人开发的一款数据持久层框架，它在 MyBatis 的基础上只做增强，不做修改，旨在提高开发效率。引入 MyBatis Plus 不会对现有工程产生任何影响，可以做到无侵入。MyBatis Plus 内置了通用 Mapper 和 Service，只需少量配置即可实现大部分 CRUD 操作，相比于 MyBatis，可以节省大量的开发时间。MyBatis Plus 还内置了丰富的插件，例如自动分页、逻辑删除和自动填充等。而且利用 MyBatis Plus 提供的

代码生成器或 Maven 插件可以生成更多代码，包括 Mapper、Model、Service、Controller 层代码，支持市面上大部分主流数据库。

2.2 开发软件介绍

2.2.1 IntelliJ IDEA

IntelliJ IDEA 支持广泛、兼容性高、功能强大。普遍适用于 Java 和 JavaEE 的系统开发。它支持市场上几乎所有的数据库链接工具，如 JDBC、Hibernate、AJAX、Struts、Java Servlet、Spring 等。IntelliJ IDEA 是业界知名的开发工具，可以帮助研究和开发数据库以及使用 JavaEE。此外，它还可以提高系统的操作能力，这在服务器集成过程中尤为突出。

2.2.2 MySQL 数据库

MySQL 的语言是结构化的，用户可以在数据上进行工作。与其他大型数据库相比，MySQL 复杂程度较低，易于使用。可以在不同的平台上运行工作。MySQL 全面支持 SQL 的子句查询和聚合函数。MySQL 因其速度、可靠性和适应性而备受关注。并且 MySQL 具有开源免费的版本，体积小，使用方便，易维护。由于 MySQL 的语言和结构相对简单，但功能和存储信息量都非常强大，因此得到了广泛的应用。

3 系统需求分析

需求分析是程序开发过程中非常重要的一部分，是程序开发的指南针。一份详细、精确的需求分析既能够减少程序开发过程中遇到的问题，也能降低后期程序维护的成本。

3.1 可行性分析

可行性研究是任何一个项目不可缺少的环节。在编写代码之前，分析相似系统的相似性或差异性可以为研究做好充分准备。从技术、经济、操作几个方面来看，系统是可以实现的。

技术可行性：系统采用了应用最广泛的技术支持程序。主要的技术支持是 Java 语言。Java 作为一种成熟的语言程序，在许多软件开发中发挥着巨大的作用^[7]。此外，用 Java 语言编辑的程序可以直接运行，无需其他翻译器进行翻译。因此，在技术上是完全可行的。

经济可行性：项目开发的初衷是为了节约，开发过程中使用的技术在市场上是通用的、易于操作的，所以不需要花钱请专业人士进行系统开发，在项目开发的过程中也学到了更多的知识。该软件的开发对计算机的软硬件要求不高，因此该项目价格合理，经济可行。

操作可行性：操作的可行性决定系统的可用性。在系统操作的设计中，采用了一种简单易懂的方式。整个操作菜单界面整洁有序，各项功能排列有序。不会有重叠或转换。用户可以在自己想要的任何方面直接操作。因此，该系统可以由任何人在没有相关专业技术的情况下操作，操作起来容易得多。

3.2 需求分析

基于 SpringBoot 的在线投稿系统有浏览人员、作者、专家、编辑和管理员这五种用户。下面分别对这五种用户的主要功能进行需求分析。

3.2.1 浏览人员主要功能需求分析

- (1) 注册：浏览人员，即未注册的用户，需要进行注册后才能登陆，使用更多功能。
- (2) 稿件信息：浏览人员可以浏览最新发布的稿件。
- (3) 公告信息：浏览人员可以浏览主页的公告信息。

3.2.2 作者主要功能需求分析

- (1) 登录：输入账号和密码进行登录，登陆成功进入后台页面使用更多功能。
- (2) 评论信息：可以查看其他人员对于自己稿件的留言及评论。
- (3) 个人中心：对自己的基本信息进行修改。
- (4) 稿件信息管理：对自己已投的稿件进行管理，查看稿件评论。
- (5) 分配信息：可以查看自己的稿件被谁所审核。
- (6) 收藏管理：可以查看自己收藏的所有稿件。
- (7) 留言板管理：对管理员进行留言以及查看管理员的回复。

3.2.3 专家主要功能需求分析

- (1) 登录：输入账号和密码进行登录，登陆成功进入后台页面使用更多功能。
- (2) 个人中心：修改自己的基本信息。
- (3) 分配信息管理：查看分配给自己的稿件信息。
- (4) 稿件信息管理：审核、修改作者的稿件。

3.2.4 编辑主要功能需求分析

- (1) 登录：管理员输入账号和密码进行登录，登录成功进入后台页面。
- (2) 个人中心：修改自己的基本信息。
- (3) 个人稿件管理：分配稿件给指定的专家审核及修改。
- (4) 分配信息管理：查看所有稿件的分配情况。
- (5) 稿件信息管理：编辑复审修改后的稿件。

3.2.5 管理员主要功能需求分析

- (1) 登录：管理员输入账号和密码进行登录，登录成功进入后台管理页面。
- (2) 个人中心：修改自己的基本信息。
- (3) 作者管理：对系统中所有的作者进行管理，可以添加或删除作者。
- (4) 专家管理：对系统中所有的专家进行管理，可以添加或删除专家。
- (5) 编辑管理：对系统中所有的编辑进行管理，可以添加或删除编辑。
- (6) 稿件管理：查看系统中所有的稿件以及审核情况。
- (7) 分配管理：查看系统中所有稿件的分配情况。
- (8) 类别管理：添加、修改或删除稿件的类别信息。
- (9) 留言板管理：查看系统中的所有人员的留言以及对他们进行回复。

(10) 系统管理：可以编辑系统公告以及轮播图等等。

4 概要设计

本章通过结构图对系统的整体功能结构进行展示，使用用例图对系统中的不同用例进行解释，通过 E-R 图对数据关系进行分析并简单解释数据库的设计^[8]。

4.1 系统的功能模块结构图

基于 SpringBoot 的在线投稿系统采用模块化设计将系统分为浏览人员、作者、专家、编辑和管理员五个模块，如图 4-1 所示。

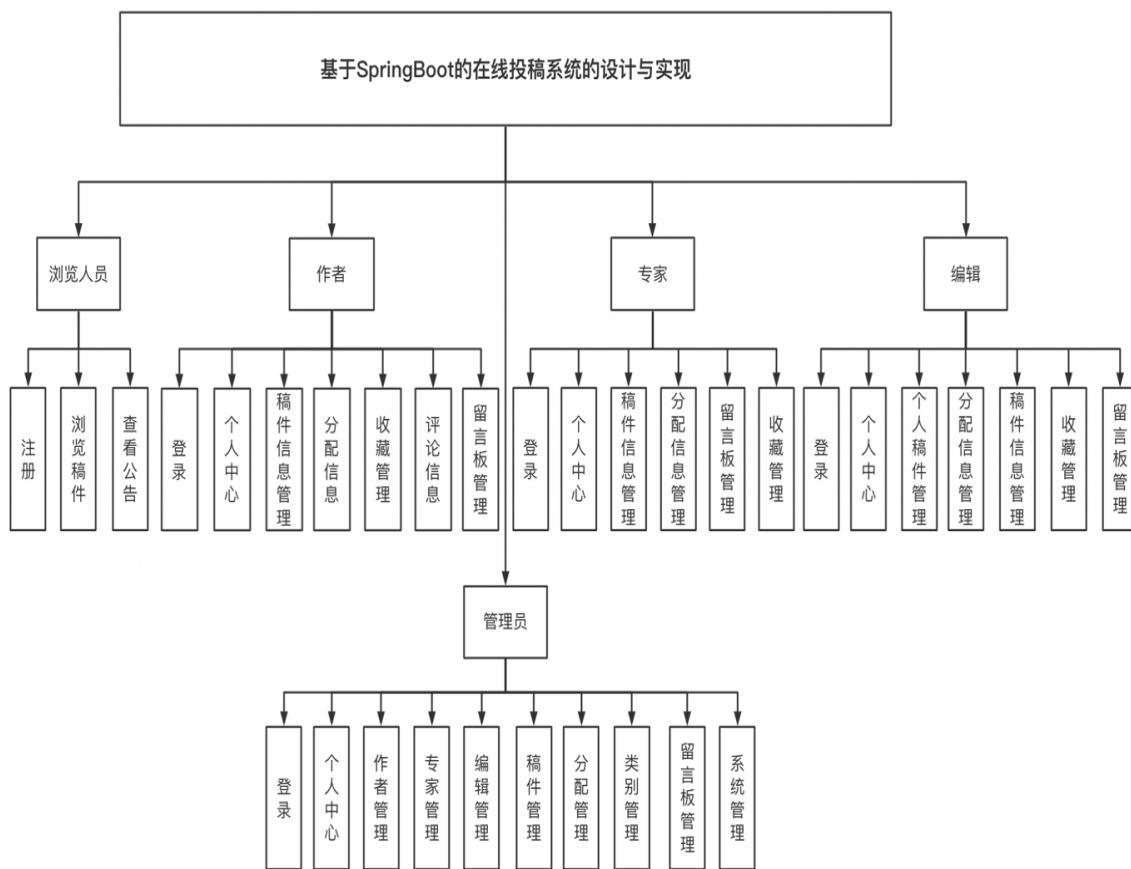


图 4-1 系统功能模块结构图

4.2 用例说明

4.2.1 浏览权员用例

浏览权员功能包括：注册，查看公告，浏览稿件信息等功能，如图 4-2 所示。

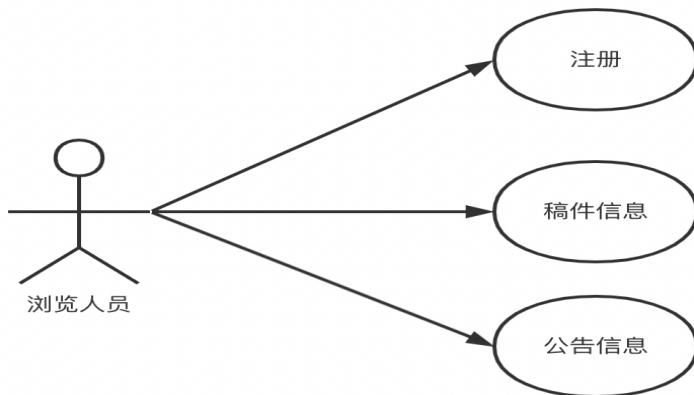


图 4-2 浏览人员用例图

4.2.2 作者用例

作者功能包括：登陆，评论信息，稿件信息管理，分配信息，个人中心，收藏管理，留言板管理等功能，如图 4-3 所示。

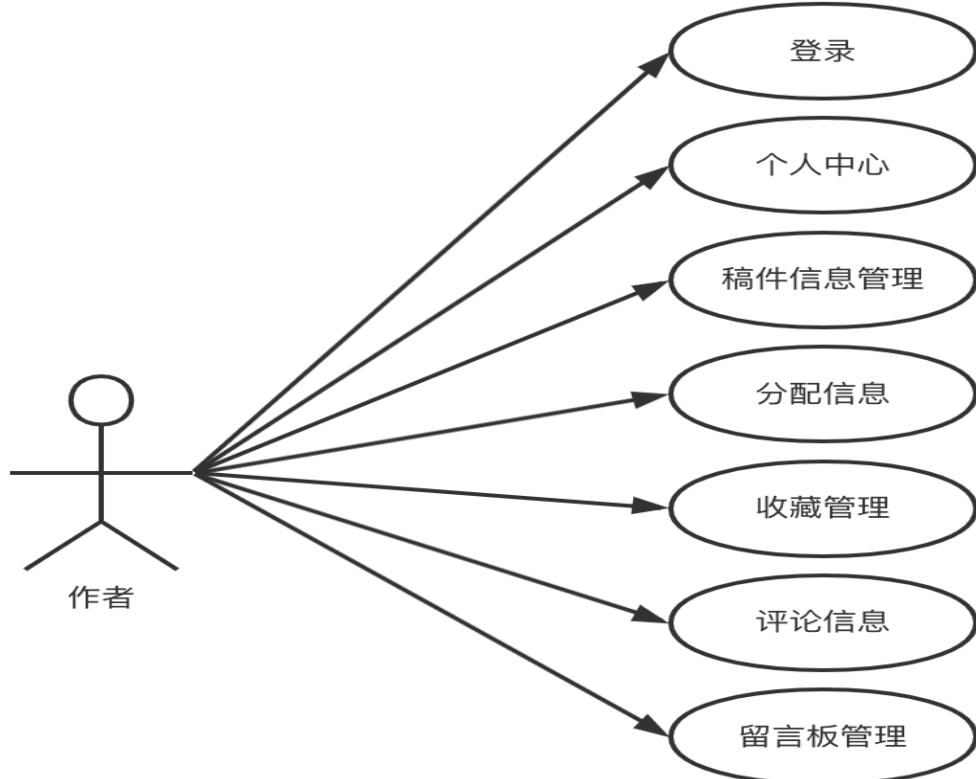


图 4-3 作者用例图

4.2.3 专家用例

专家功能包括：登录，个人中心，稿件信息管理，分配信息管理，留言板管理，收藏管理等功能，如图 4-4 所示。

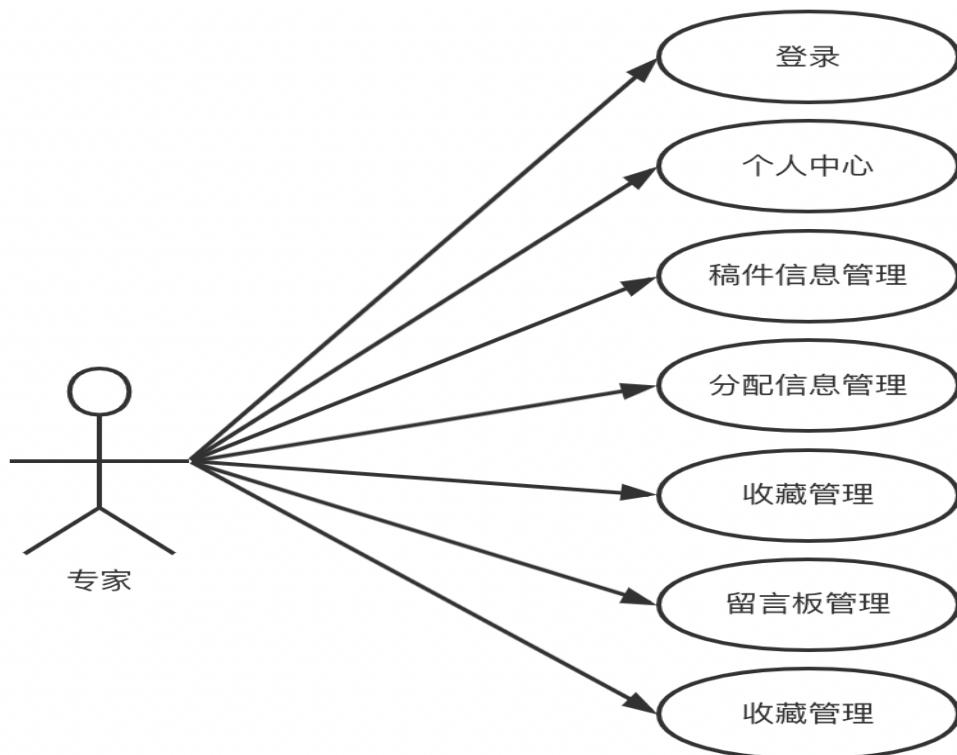


图 4-4 专家用例图

4.2.4 编辑用例

编辑功能包括：登录，个人中心，个人稿件管理，分配信息管理，稿件信息管理，收藏管理，留言板管理等功能，如图 4-5 所示。

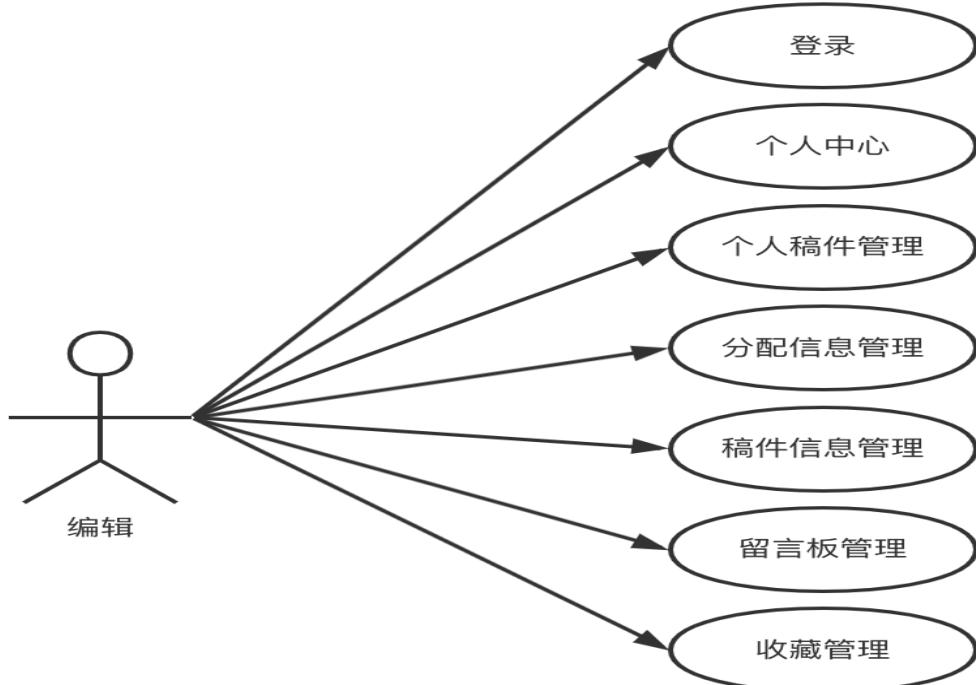


图 4-5 编辑用例图

4.2.5 管理员用例

管理员功能包括：登录，个人中心，作者管理，专家管理，编辑管理，稿件管理，分配管理，类别管理，留言板管理，系统管理等功能，如图 4-6 所示。

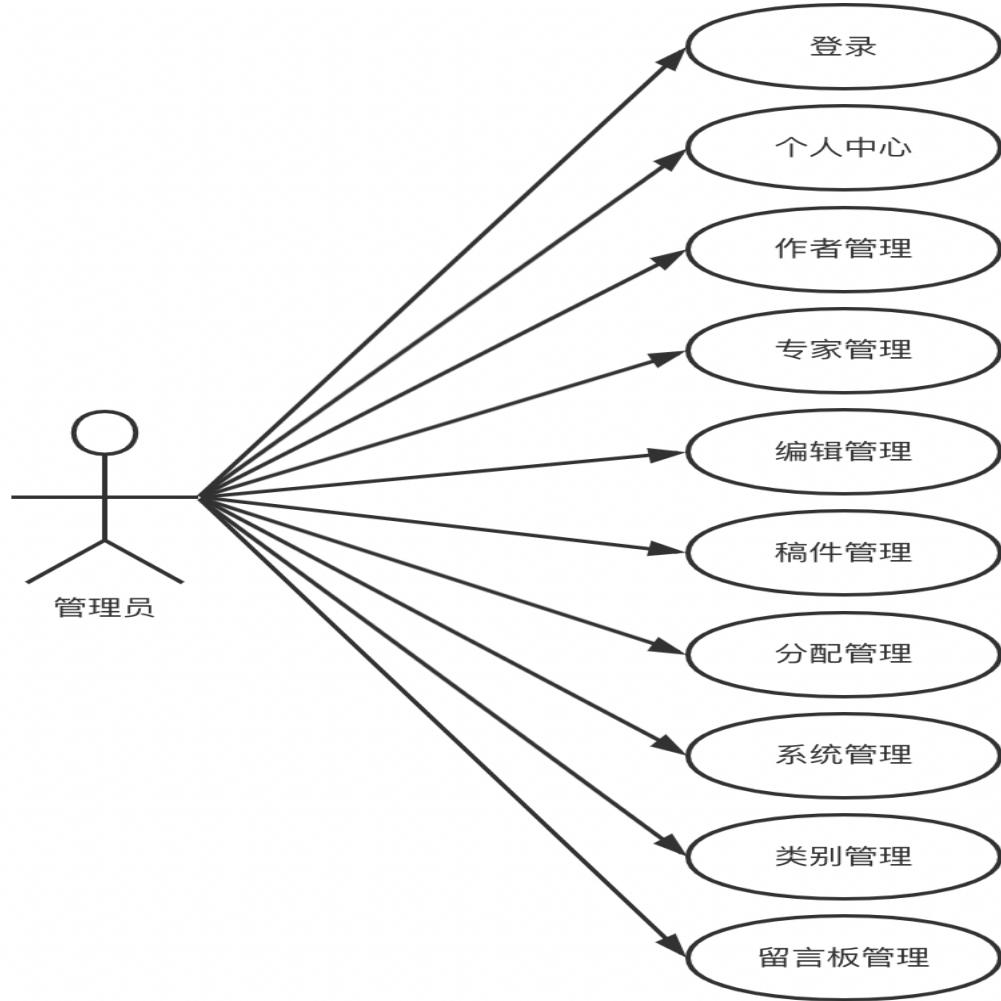


图 4-6 管理员用例图

4.3 数据库设计

一个系统的数据库设计是非常重要的，在程序开发中占据非常关键的地位。系统数据库设计是否优秀直接影响整个系统的运行效率^[9]。

4.3.1 数据库概念结构

(1) 作者信息 E-R 图

作者信息的 E-R 图中，矩形是实体，九个属性分别为主键、创建时间、账号、密码、姓名、性别、邮箱、手机号码、相片，具体 E-R 图如图 4-7 所示。

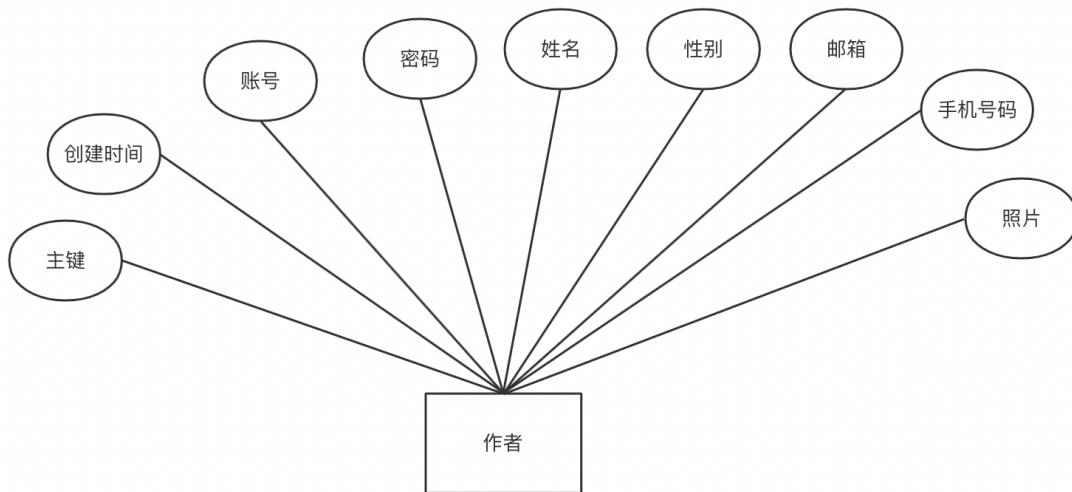


图 4-7 作者信息 E-R 图

(2) 专家信息 E-R 图

专家信息的 E-R 图中，矩形是实体，九个属性分别为主键、创建时间、专家账号、密码、专家姓名、性别、邮箱、手机号码、相片，具体 E-R 图如图 4-8 所示。

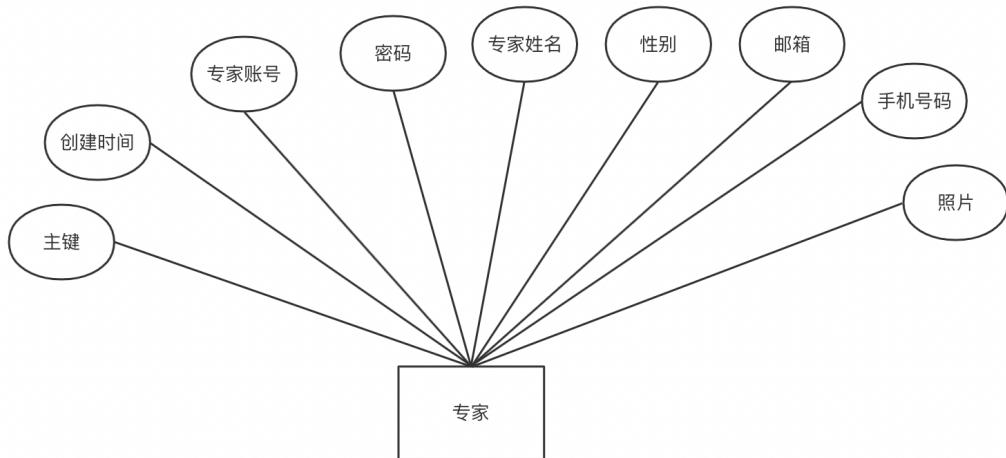


图 4-8 专家信息 E-R 图

(3) 编辑信息 E-R 图

编辑信息的 E-R 图中，矩形是实体，九个属性分别为主键、创建时间、编辑账号、密码、编辑姓名、性别、邮箱、手机号码、相片，具体 E-R 图如图 4-9 所示。

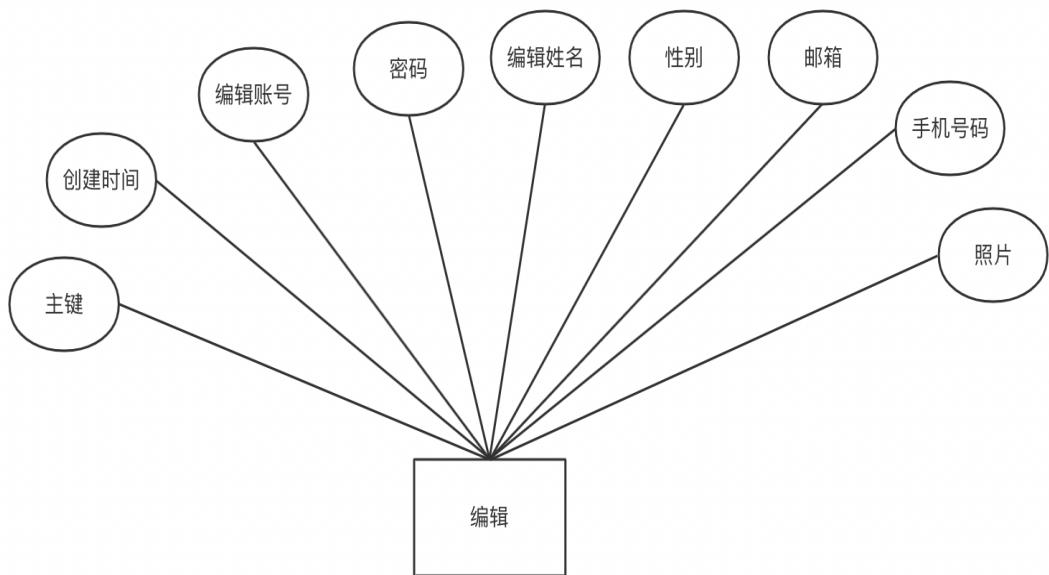


图 4-9 编辑信息 E-R 图

(4) 管理员信息 E-R 图

管理员信息的 E-R 图中，矩形是实体，五个属性分别为主键、用户名、密码、角色、添加时间，具体 E-R 图如图 4-10 所示。

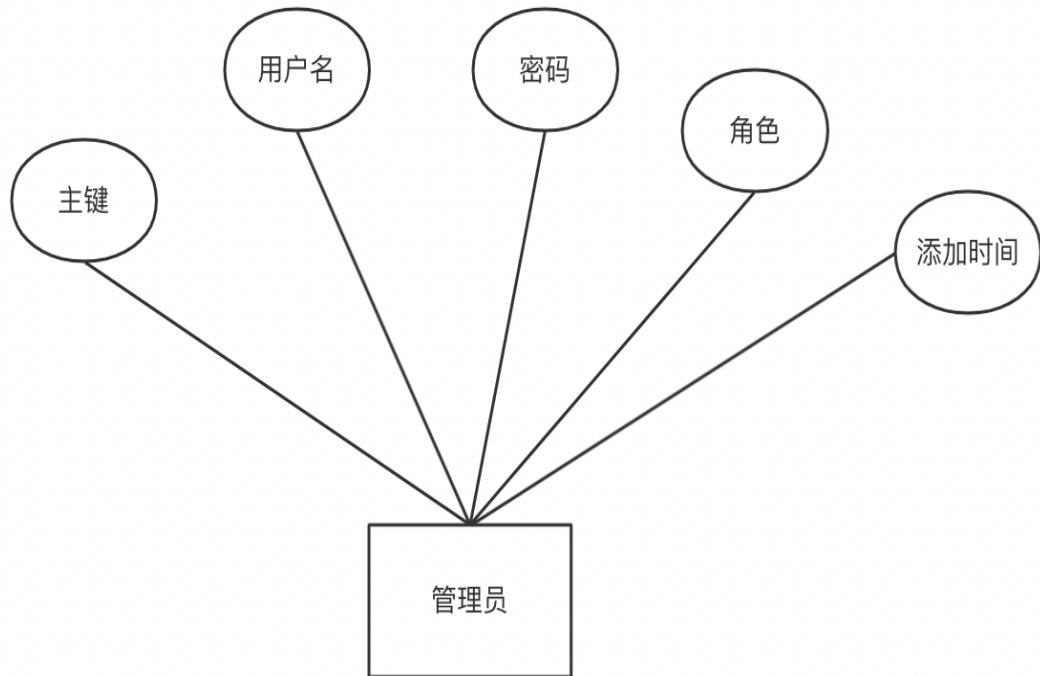


图 4-10 管理员信息 E-R 图

4.3.2 数据库逻辑结构

(1) 作者表

主要存储作者的信息，含有九个字段，主键 ID 设置为自增、bigint 类型，使用 MyBatis Plus 默认的雪花算法生成，创建时间设置为 timestamp 时间戳类型，默认为当前时间，账号、密码、姓名、性别、邮箱、手机号码和相片都设置为 varchar 可变长度字符类型。如表 4-1 所示。

表 4-1 作者表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	是	--
创建时间	addtime	timestamp	否	否	--
账号	zhanghao	varchar(200)	否	否	--
密码	mima	varchar(200)	否	否	--
姓名	xingming	varchar(200)	否	否	--
性别	xingbie	varchar(200)	是	否	--
邮箱	youxiang	varchar(200)	是	否	--
手机号码	shoujihaoma	varchar(200)	是	否	--
相片	xiangpian	varchar(200)	是	否	--

(2) 专家表

主要存储专家的信息，含有九个字段，主键 ID 设置为自增、bigint 类型，使用 MyBatis Plus 默认的雪花算法生成，创建时间设置为 timestamp 时间戳类型，默认为当前时间，专家账号、密码、专家姓名、性别、邮箱、手机号码和相片都设置为 varchar 可变长度字符类型。如表 4-2 所示。

表 4-2 专家表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	是	--
创建时间	addtime	timestamp	否	否	--
专家账号	zhanghao	varchar(200)	否	否	--
密码	mima	varchar(200)	否	否	--
专家姓名	xingming	varchar(200)	否	否	--
性别	xingbie	varchar(200)	是	否	--
邮箱	youxiang	varchar(200)	是	否	--

续表 4-2

字段名称	字段	类型长度	是否为空	主键	备注
手机号码	lianxishouji	varchar(200)	是	否	--
相片	xiangpian	varchar(200)	是	否	--

(3) 编辑表

主要存储编辑的信息，含有九个字段，主键 ID 设置为自增、bigint 类型，使用 MyBatis Plus 默认的雪花算法生成，创建时间设置为 timestamp 时间戳类型，默认为当前时间，编辑账号、密码、编辑姓名、性别、邮箱、手机号码和相片都设置为 varchar 可变长度字符类型。如表 4-3 所示。

表 4-3 编辑表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	是	--
创建时间	addtime	timestamp	否	否	--
编辑账号	zhanghao	varchar(200)	否	否	--
密码	mima	varchar(200)	否	否	--
编辑姓名	xingming	varchar(200)	否	否	--
性别	xingbie	varchar(200)	是	否	--
邮箱	youxiang	varchar(200)	是	否	--
手机号码	shoujihaoma	varchar(200)	是	否	--
相片	xiangpian	varchar(200)	是	否	--

(4) 管理员表

主要存储管理员的信息，含有五个字段，主键设置为自增、int 类型，用户名、密码和角色设置为 varchar 可变长度字符类型，创建时间设置为 timestamp 时间戳类型，默认为当前时间。如表 4-4 所示。

表 4-4 管理员表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	是	--
用户名	username	varchar(100)	否	否	--
密码	password	varchar(100)	否	否	--

续表 4-4

字段名称	字段	类型长度	是否为空	主键	备注
角色	role	varchar(100)	是	否	--
创建时间	addtime	timestamp	否	否	--

(5) 稿件信息评论表

主要存储对稿件评论的信息，含有七个字段，主键 ID 设置为自增、bigint 类型，使用 MyBatis Plus 默认的雪花算法生成，创建时间设置为 timestamp 时间戳类型，默认为当前时间，关联表 ID 和用户 ID 设置为 bigint 类型，用户名设置为 varchar 可变长度字符类型，评论内容和回复内容设置为 longtext 类型。如表 4-5 所示。

表 4-5 稿件信息评论表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	是	--
创建时间	addtime	timestamp	否	否	--
关联表 id	refid	bigint	否	否	--
用户 id	userid	bigint	否	否	--
用户名	nickname	varchar(200)	是	否	--
评论内容	content	longtext	否	否	--
回复内容	reply	longtext	是	否	--

(6) 分配信息表

主要存储对稿件分配的信息，含有十六个字段，主键 ID 设置为自增、bigint 类型，使用 MyBatis Plus 默认的雪花算法生成，创建时间设置为 timestamp 时间戳类型，默认为当前时间，名称、章节、封面、类别、附件、账号、姓名、发布日期、专家账号、专家姓名和分配备注都设置为 varchar 可变长度字符类型，简介和内容设置为 longtext 类型，分配时间设置为 date 类型。如表 4-6 所示。

表 4-6 分配信息表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	是	--
创建时间	addtime	timestamp	否	否	--
名称	mingcheng	varchar(200)	否	否	--
章节	zhangjie	varchar(200)	否	否	--

续表 4-6

字段名称	字段	类型长度	是否为空	主键	备注
封面	fengmian	varchar(200)	是	否	--
类别	leibie	varchar(200)	是	否	--
简介	jianjie	longtext	是	否	--
内容	neirong	longtext	是	否	--
附件	fujian	varchar(200)	是	否	--
账号	zhanghao	varchar(200)	是	否	--
姓名	xingming	varchar(200)	是	否	--
发布日期	faburiqi	varchar(200)	是	否	--
专家账号	zjzhanghao	varchar(200)	是	否	--
专家姓名	zjxingming	varchar(200)	是	否	--
分配备注	fenpeibeizhu	varchar(200)	是	否	--
分配时间	fenpeishijian	date	是	否	--

(7) 稿件信息表

主要存储稿件的基本信息，含有十六个字段，主键 ID 设置为自增、bigint 类型，使用 MyBatis Plus 默认的雪花算法生成，创建时间设置为 timestamp 时间戳类型，默认为当前时间，名称、章节、封面、类别、附件、账号、姓名、发布日期、专家账号、专家姓名和是否审核都设置为 varchar 可变长度字符类型，内容和简介设置为 longtext 类型，审核回复设置为 longtext 类型。如表 4-7 所示。

表 4-7 稿件信息表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	是	--
创建时间	addtime	timestamp	否	否	--
名称	mingcheng	varchar(200)	否	否	--
章节	zhangjie	varchar(200)	否	否	--
封面	fengmian	varchar(200)	是	否	--
类别	leibie	varchar(200)	是	否	--
简介	jianjie	longtext	是	否	--
内容	neirong	longtext	是	否	--

续表 4-7

字段名称	字段	类型长度	是否为空	主键	备注
附件	fujian	varchar(200)	是	否	--
账号	zhanghao	varchar(200)	是	否	--
姓名	xingming	varchar(200)	是	否	--
发布日期	faburiqi	varchar(200)	是	否	--
专家账号	zjzhanghao	varchar(200)	是	否	--
专家姓名	zjxingming	varchar(200)	是	否	--
是否审核	shifoushenhe	varchar(200)	是	否	--
审核回复	shenhehuifu	longtext	是	否	--

(8) 个人稿件表

主要存储稿件的审核情况信息，含有十四个字段，主键 ID 设置为自增、bigint 类型，使用 MyBatis Plus 默认的雪花算法生成，创建时间设置为 timestamp 时间戳类型，默认为当前时间，名称、章节、封面、类别、附件、账号、姓名、发布日期和是否审核都设置为 varchar 可变长度字符类型，简介、内容和审核回复设置为 longtext 类型。如表 4-8 所示。

表 4-8 个人稿件表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	否	--
创建时间	addtime	timestamp	否	否	--
名称	mingcheng	varchar(200)	否	否	--
章节	zhangjie	varchar(200)	否	否	--
封面	fengmian	varchar(200)	是	否	--
类别	leibie	varchar(200)	是	否	--
简介	jianjie	longtext	是	否	--
内容	neirong	longtext	是	否	--
附件	fujian	varchar(200)	是	否	--
账号	zhanghao	varchar(200)	是	否	--
姓名	xingming	varchar(200)	是	否	--
发布日期	faburiqi	varchar(200)	是	否	--

续表 4-8

字段名称	字段	类型长度	是否为空	主键	备注
是否审核	sfsh	varchar(200)	是	否	--
审核回复	sfhh	longtext	是	否	--

(9) 稿件类别表

主要存储稿件的类别信息，含有三个字段，主键 ID 设置为自增、bigint 类型，使用 MyBatis Plus 默认的雪花算法生成，创建时间设置为 timestamp 时间戳类型，默认使用当前时间，类别设置为 varchar 可变长度字符类型。如表 4-9 所示。

表 4-9 稿件类别表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	是	--
创建时间	addtime	timestamp	否	否	--
类别	leibie	varchar(200)	否	否	--

(10) 留言板表

主要存储系统人员的留言信息与管理员的回复信息，含有八个字段，主键 ID 设置为自增、bigint 类型，使用 MyBatis Plus 默认的雪花算法生成，创建时间设置为 timestamp 时间戳类型，默认使用当前时间，留言人 ID 设置为 bigint 类型，用户名、留言图片和回复图片设置为 varchar 可变长度字符串，留言内容和回复内容设置为 longtext 类型。如表 4-10 所示。

表 4-10 留言板表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	是	--
创建时间	addtime	timestamp	否	否	--
留言人 id	userid	bigint	否	否	--
用户名	username	varchar(200)	是	否	--
留言内容	content	longtext	否	否	--
留言图片	cpicture	varchar(200)	是	否	--
回复内容	reply	longtext	是	否	--
回复图片	rpicture	varchar(200)	是	否	--

(11) 公告信息表

主要存储系统的公告信息，含有六个字段，主键 ID 设置为自增、bigint 类型，使用 MyBatis Plus 默认的雪花算法生成，创建时间设置为 timestamp 时间戳类型，默认使用当前时间，标题和图片设置为 varchar 可变长度字符类型，简介和内容设置为 longtext 类型。如表 4-11 所示。

表 4-11 公告信息表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	是	--
创建时间	addtime	timestamp	否	否	--
标题	title	varchar(200)	否	否	--
简介	introduction	longtext	是	否	--
图片	picture	varchar(200)	否	否	--
内容	content	longtext	否	否	--

(12) 收藏表

主要存储对稿件收藏的信息，有九个字段，主键 ID 设置为自增、bigint 类型，使用 MyBatis Plus 默认的雪花算法生成，创建时间设置为 timestamp 时间戳类型，默认使用当前时间，用户 ID 和收藏 ID 设置为 bigint 类型，表名、收藏名称、收藏图片、类型和推荐类型设置为 varchar 可变长度字符类型。如表 4-12 所示。

表 4-12 收藏表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	是	--
创建时间	addtime	timestamp	否	否	--
用户 id	userid	bigint	否	否	--
收藏 id	refid	bigint	是	否	--
表名	tablename	varchar(200)	是	否	--
收藏名称	name	varchar(200)	否	否	--
收藏图片	picture	varchar(200)	否	否	--
类型	type	varchar(200)	是	否	--
推荐类型	inteltype	varchar(200)	是	否	--

(13) token 表

主要存储用户登录产生的 token 信息。token 是用户登录之后服务端生成的一串字符串，用于客户端请求的令牌。用户登录后，服务器生成一个 token 返回给客户端，客户端存储 token。每次客户端与服务端交互时验证 token 并返回数据。含有八个字段，主键 ID 和用户名 ID 设置为 bigint 类型，用户名、表名、角色和密码设置为 varchar 可变长度字符类型，新增时间和过期时间设置为 timestamp 时间戳类型，默认使用当前时间。如表 4-13 所示。

表 4-13 token 表

字段名称	字段	类型长度	是否为空	主键	备注
主键	id	bigint	否	是	--
用户名 id	userid	bigint	否	否	--
用户名	username	varchar(200)	否	否	--
表名	tablename	varchar(200)	是	否	--
角色	role	varchar(200)	是	否	--
密码	token	varchar(200)	否	否	--
新增时间	addtiem	timestamp	否	否	--
过期时间	expiratedtime	timestamp	否	否	--

5 详细设计

概要设计是对系统整体功能简单的解释，但是系统的设计和开发还需要更加详细的工作，详细设计里面介绍了系统内模块的详细流程的逻辑，是系统开发和代码编写的逻辑依据。该系统使用 Maven 作为构建工具，通过 IntelliJ IDEA 中自带的 Spring Initialize 模板创建 SpringBoot 项目。项目创建完成后，在 pom.xml 中引入 Maven 相关依赖，使用 spring-boot-start-parent 作为工程的父依赖，所有以 spring-boot-start 开头的依赖都会使用父依赖版本^[10]。Thymeleaf 作为模板引擎来解析数据，渲染页面。依赖导入完成后，在 SpringBoot 的启动类上加上@SpringBootApplication 注解，来标注此工程为 SpringBoot 应用。日志输出使用 MyBatis Plus 自带的标准日志输出。下面介绍部分模块的设计流程图。

5.1 注册模块设计流程图

用户注册时需填写相应的表单数据，切格式必须正确，再与数据库对比，看用户是否已存在，若已存在则弹出提示信息。注册模块设计流程图如图 5-1 所示。

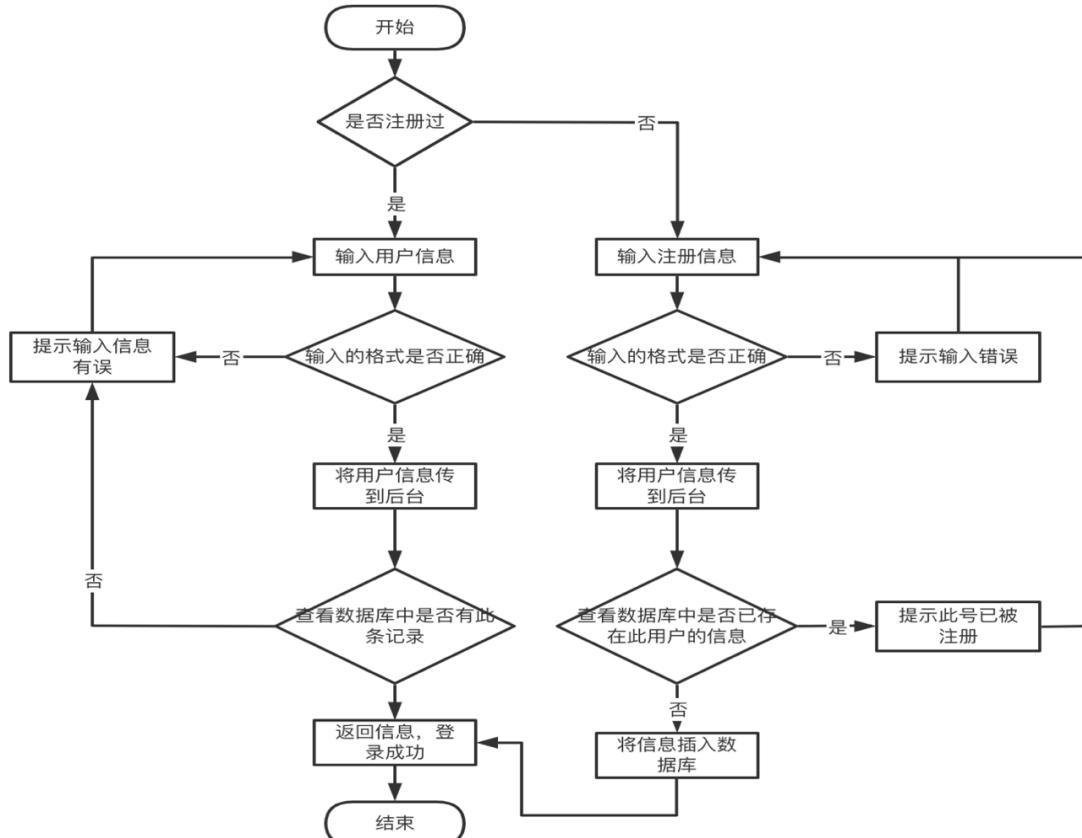


图 5-1 注册模块设计流程图

5.2 作者上传稿件模块设计流程图

作者上传稿件需是登录状态，若没登录则提示登录，登录后填写相应的表单数据即可上传稿件。作者上传稿件模块设计流程图如图 5-2 所示。

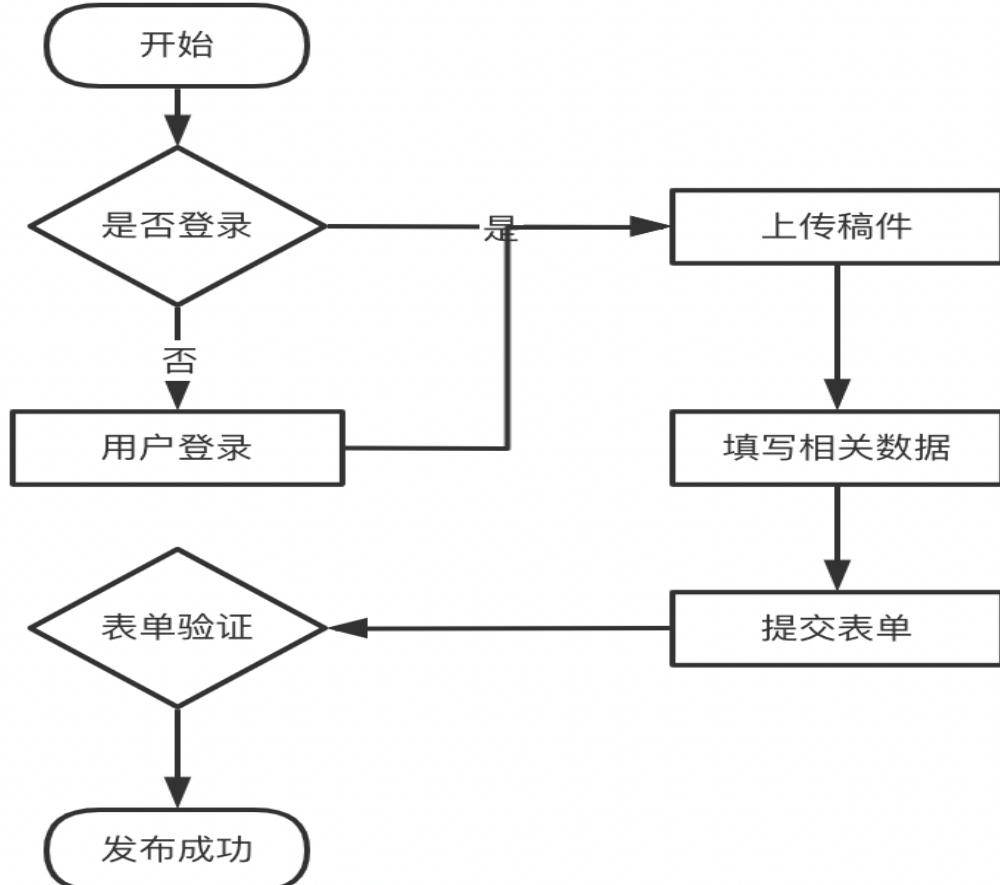
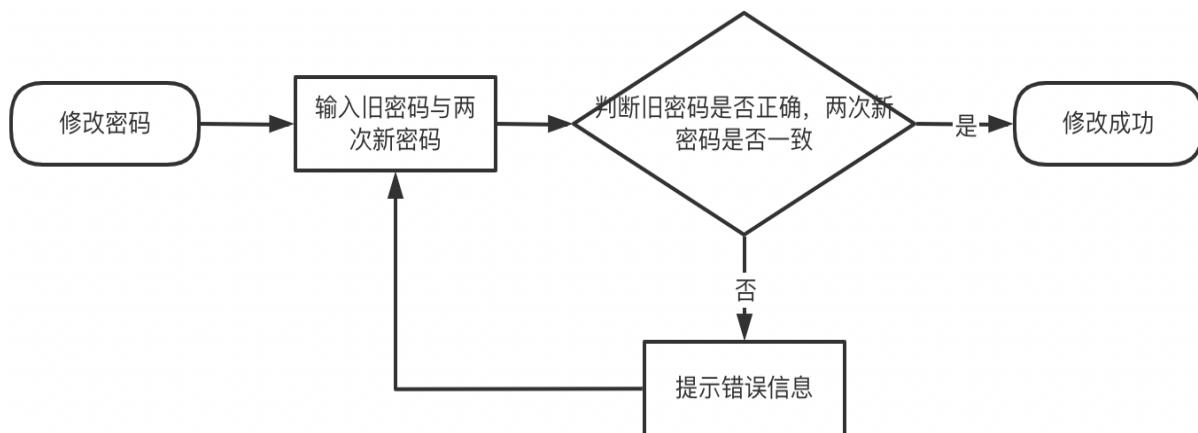


图 5-2 上传稿件模块设计流程图

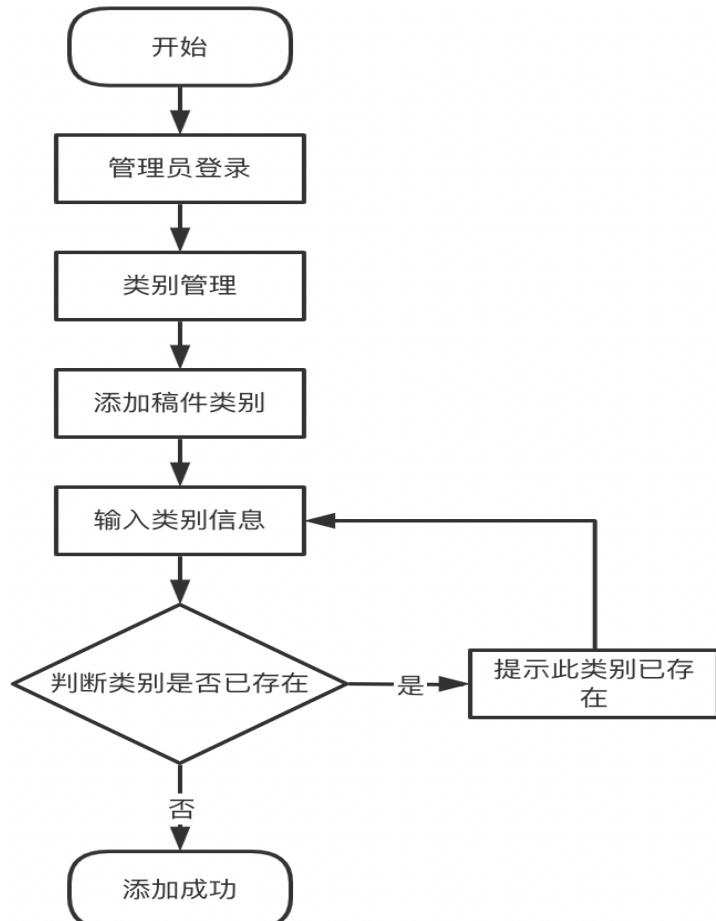
5.3 修改密码模块设计流程图

用户修改密码时需填写旧密码和两次新密码。判断旧密码输入是否正确，如果输入正确再判断两次新密码输入是否一致，不一致则弹出提示信息并重新输入，一致则完成修改。修改密码模块设计流程图如图 5-3 所示。



5.4 添加类别信息模块设计流程图

添加类别信息是管理员独有的功能之一，管理员在登录之后可以添加稿件的类别信息。输入稿件类别之后判断此类别是否存在，如果存在则提示已存在信息，否则添加成功。添加类别信息模块设计流程图如图 5-4 所示。



5.5 添加公告模块设计流程图

添加公告是管理员独有的功能之一，管理员在登录之后可以添加系统首页公告信息。输入标题、图片、简介以及内容之后验证表单信息，如果有未填写的输入框则提示未输入信息。添加公告信息模块设计流程图如图 5-5 所示。

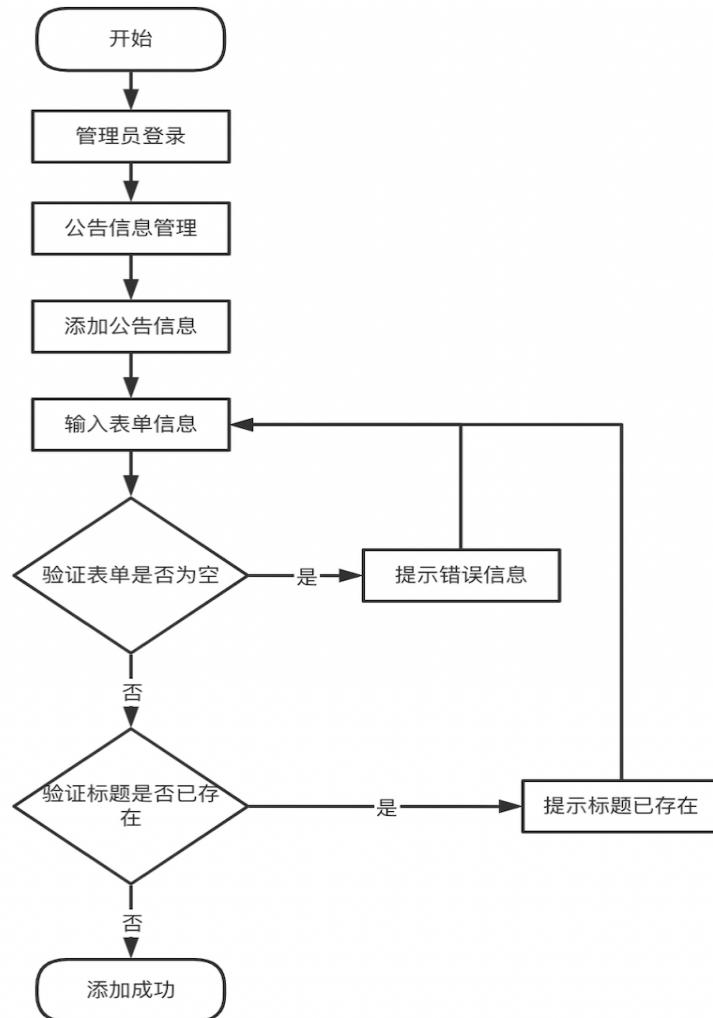


图 5-5 添加公告信息模块设计流程图

5.6 添加编辑模块设计流程图

管理员可以通过后台添加编辑。输入编辑账号、密码、姓名等信息，验证表单是否为空，为空则提示未输入信息，不为空再验证编辑账号是否已存在，如果存在则提示已存在信息，不存在则添加成功。添加编辑模块设计流程图如图 5-6 所示。

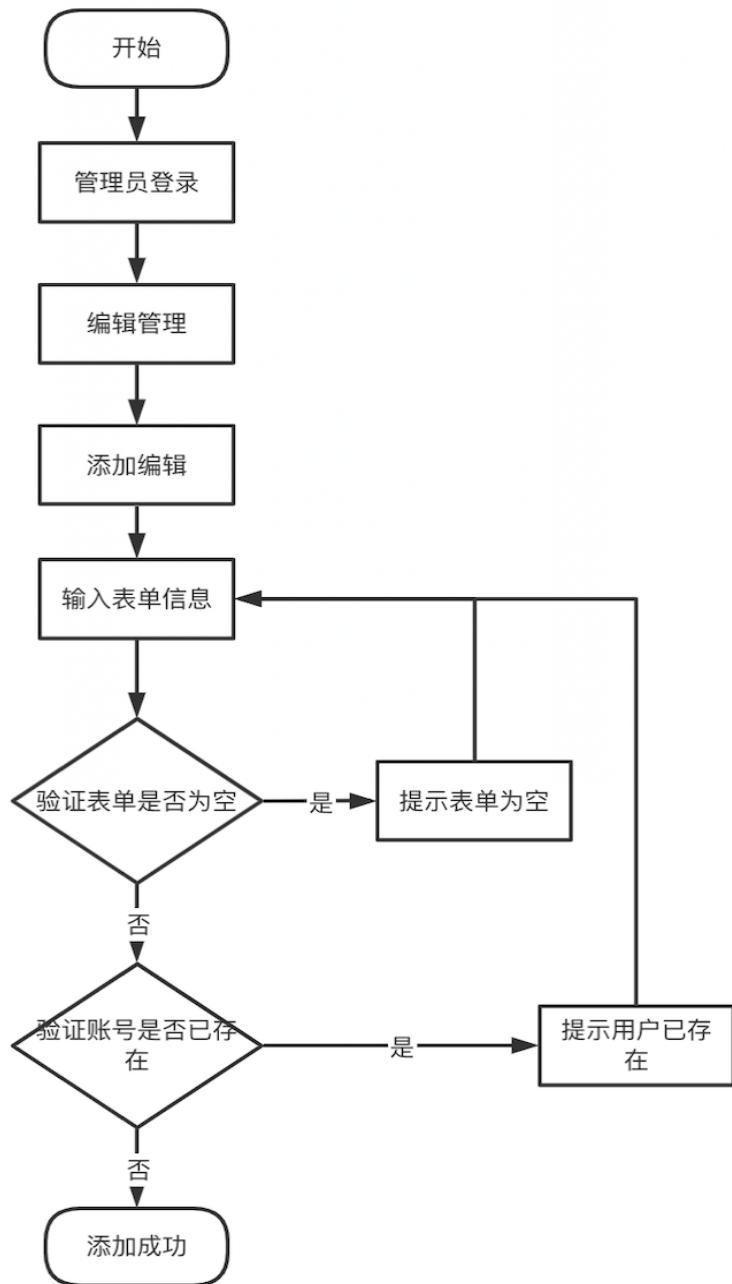


图 5-6 添加编辑模块设计流程图

6 编码实现

本章介绍系统功能在实现过程中具体的代码细节，包括开发环境介绍、软件代码架构、详细代码实现过程等。

6.1 开发环境与组织结构

本系统的开发环境如表 6-1 所示。

表 6-1 系统开发环境介绍表

操作系统	JDK 版本	数据库	IDE	开发框架
MacOS 11.6.5	OpenJDK 1.8.0_322	MySQL 8.0.28	IntelliJ IDEA Ultimate 2021.3.3	SpringBoot+MyBatis Plus

项目开发的代码结构如图 6-1 所示。src 目录下存放源代码，annotation 包下存放注解类，config 包下存放配置类，controller 包下存放控制层代码，dao 包下存放 Dao 接口，entity 包下存放实体类，interceptor 包下存放拦截器，service 包下存放业务层代码，utils 包下存放工具类。

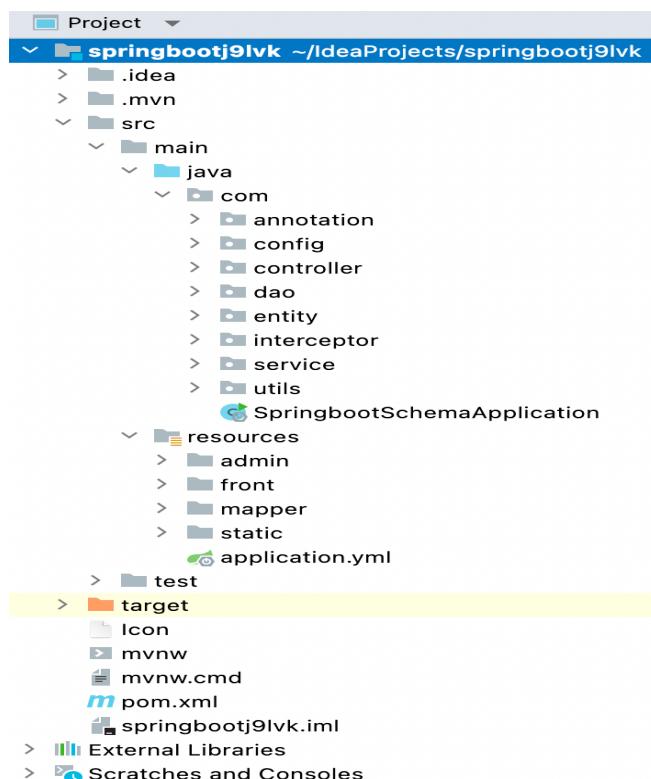


图 6-1 代码组织结构图

6.2 登录功能实现

用户登录时会生成一个 token 信息存在数据库中，每次登录都会把 token 带到浏览器中存储起来，当浏览器与服务器交互时会验证 token 信息是否一致，如果一致则返回相应的数据与信息。登录页面如图 6-2 所示。



图 6-2 登录页面

作者登录部分代码实现如图 6-3 所示，前端输入用户名和密码，后端通过 MyBatis Plus 提供的条件构造器 Wrapper 判断用户是否存在，若用户存在再判断密码是否正确，密码正确则调用 token 的业务层方法，结合用户输入的信息生成一个 token 用作令牌，最后返回信息。

```

@IgnoreAuth
@RequestMapping(value = "/Login")
public R login(String username, String password, String captcha, HttpServletRequest request) {
    ZuozheEntity user = zuozheService.selectOne(new EntityWrapper<ZuozheEntity>().eq(column: "zhanghao", username));
    if(user==null || !user.getMima().equals(password)) {
        return R.error("账号或密码不正确");
    }

    String token = tokenService.generateToken(user.getId(), username, tableName: "zuozhe", role: "作者");
    return R.ok().put("token", token);
}

```

图 6-3 作者登录功能代码实现

6.3 文件上传功能

文件上传是作者上传稿件时的功能。文件以附件的形式上传。数据库中存储文件路径，文件上传到 SpringBoot 自带的 Tomcat 服务器当中。作者上传稿件页面如图 6-4 所示。

The screenshot shows a web-based form for uploading a thesis. At the top, there are two required fields: '名称' (Name) and '章节' (Chapter), both marked with red asterisks. Below these are two upload sections: '封面' (Cover) and '附件' (Attachment), each with a large input field containing a plus sign and a placeholder '点击上传封面' (Click to upload cover) or '点击上传附件' (Click to upload attachment). Further down, there are fields for '账号' (Account) with value '11', '姓名' (Name) with value '张三', and '发布日期' (Release Date) with value '2022 年 04 月 28 日'. There is also a '简介' (Introduction) section with a text area and a rich text editor toolbar above it. The toolbar includes icons for bold (B), italic (I), underline (U), strikethrough (S), quote (‘’), code (‘‘), H1 and H2 headings, lists (list, bullet), standard font (A), and font size (14px). At the bottom, there are two buttons: a dark grey '提交' (Submit) button and a light grey '取消' (Cancel) button.

图 6-4 上传稿件页面

利用 Spring 提供的 `MultipartFile` 接口来接收用户上传的文件，再调用接口中的 `transferTo()` 方法将文件保存到指定位置上。文件上传代码实现如图 6-5 所示。

```

@RequestMapping("/upload")
public R upload(@RequestParam("file") MultipartFile file, String type) throws Exception {
    if (file.isEmpty()) {
        throw new EIException("上传文件不能为空");
    }
    String fileExt = file.getOriginalFilename().substring(file.getOriginalFilename().lastIndexOf(".") + 1);
    File path = new File(ResourceUtils.getURL(resourceLocation: "classpath:static").getPath());
    if (!path.exists()) {
        path = new File(pathname: "");
    }
    File upload = new File(path.getAbsolutePath(), child: "/upload/");
    if (!upload.exists()) {
        upload.mkdirs();
    }
    String fileName = new Date().getTime() + fileExt;
    File dest = new File(pathname: upload.getAbsolutePath() + "/" + fileName);
    file.transferTo(dest);
    FileUtils.copyFile(dest, new File(pathname: "src/main/resources/static/upload/" + "/" + fileName));
    if (StringUtils.isNotBlank(type) && type.equals("1")) {
        ConfigEntity configEntity = configService.selectOne(
            new EntityWrapper<ConfigEntity>().eq(column: "name", params: "faceFile"));
        if (configEntity == null) {
            configEntity = new ConfigEntity();
            configEntity.setName("faceFile");
            configEntity.setValue(fileName);
        } else {
            configEntity.setValue(fileName);
        }
        configService.insertOrUpdate(configEntity);
    }
    return R.ok().put("file", fileName);
}

```

图 6-5 上传稿件代码实现

6.4 文件下载功能

专家和编辑在进行审核稿件时需要下载附件进行查看。文件下载返回的是一个下载链接，下载 `byte[]` 流。浏览器通过 `http` 协议获取文件，只要设置好请求(`request`)和响应(`response`)的一些信息就可以了。通过 Spring 提供的 `HttpHeaders` 封装请求头信息，通知浏览器以 `attachment`(附件)的形式下载文件。文件下载代码如图 6-6 所示。

```
@IgnoreAuth
@RequestMapping(value="/download")
public ResponseEntity<byte[]> download(@RequestParam String fileName) {
    try {
        File path = new File(ResourceUtils.getURL(resourceLocation: "classpath:static").getPath());
        if(!path.exists()) {
            path = new File(pathname: "");
        }
        File upload = new File(path.getAbsolutePath(), child: "/upload/");
        if(!upload.exists()) {
            upload.mkdirs();
        }
        File file = new File(pathname: upload.getAbsolutePath() + "/" + fileName);
        if(file.exists()){
            HttpHeaders headers = new HttpHeaders();
            headers.setContentType(MediaType.APPLICATION_OCTET_STREAM);
            headers.setContentDispositionFormData(name: "attachment", fileName);
            return new ResponseEntity<byte[]>(FileUtils.readFileToByteArray(file),headers, HttpStatus.CREATED);
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
    return new ResponseEntity<byte[]>(HttpStatus.INTERNAL_SERVER_ERROR);
}
```

图 6-6 下载稿件代码实现

7 系统测试

7.1 测试目的

随着互联网的不断发展，目前各大领域都在使用互联网进行信息管理。质量问题是一个非常重要的标准，它也决定了是否有更多的人使用它。因此，必须检查软件的质量。必须把软件做好，到位，避免不必要的问题，这样更多的用户可以使用它，得到更多的推广。开发系统后，需要进行大量测试，以确保系统的稳定性和可用性，并确定系统的质量是否能够满足不同人群的需求^[11]。这是系统开发设计中非常重要的一个环节。测试结果直接关系到系统的质量。

集成测试：在系统测试中会有很多问题，应该及时标记，并在测试时进行自动化测试，以便准确快速，不会出现疲劳。手动测试容易出现疲劳期，测试结果有时会出错，所以测试时最好进行自动测试。

在测试过程中，及时发现问题并加以解决，使设计的系统能够正常稳定运行，无重大问题。所进行的软件测试参照以下三个步骤进行测试。

- (1) 模块测试：测试系统中的每个项目以发现问题。
- (2) 系统测试：让系统在各种情况下长时间运行，反馈运行中的稳定性问题并解决。
- (3) 验收测试：在完成其他测试后，最后的检查阶段，确保软件准备就绪。

7.2 测试方法

在对系统进行测试的时候主要应用到黑盒测试和白盒测试两种的方法，通过测试就能找出可能存在的问题保证系统成功运行。

黑盒测试：测试系统功能。用户进行相应操作时，系统是否能及时、准确地反馈数据并执行相应功能^[12]。必须对功能和使用方法进行详细测试，确保所有操作信息都能完整输出和输入。

白盒测试：测试系统的结构，了解系统在运行中的工作状态。

7.3 功能模块测试

基于 SpringBoot 的在线投稿系统的设计基本达到了理想的开发状态，各项功能运行良好，基本满足了用户的使用需求，及时更正了更多的错误信息。总的来说，软件已经通过了相应的测试。主要测试内容包括功能模块测试和压力测试。

7.3.1 登录模块测试

用户输入正确的登录内容，会进入前台页面；如果输入的信息不正确，则不能正常进入系统并提示用户错误信息。登录模块的测试表如表 7-1 所示。

表 7-1 登录模块测试表

功能模块	输入内容	预期效果	实际效果	是否通过
登录	直接点击登录	提示请输入用户名 和密码	提示请输入用户名 和密码	是
登录	输入错误密码	提示密码错误	提示密码错误	是
登录	输入错误用户名	提示用户名不存在	提示用户名不存在	是
登录	登录内容均输入 正确	登录成功	登录成功	是

7.3.2 作者管理模块测试

对作者管理模块的功能进行测试，包括对作者的增删改查。作者管理模块的部分测试用例及结果如表 7-2 所示。

表 7-2 作者管理模块测试表

功能模块	输入内容	预期效果	实际效果	是否通过
作者管理	添加信息缺失	提示添加对应缺失 信息	提示添加对应缺失 信息	是
作者管理	修改信息错误	提示修改信息有误	提示修改信息有误	是
作者管理	按姓名查询作者	显示该姓名的作者 信息	显示该姓名的作者 信息	是
作者管理	删除作者信息	删除成功	删除成功	是

7.3.3 专家管理模块测试

对专家管理模块的功能进行测试，包括对专家的增删改查。专家管理模块的部分测试用例及结果如表 7-3 所示。

表 7-2 专家管理模块测试表

功能模块	输入内容	预期效果	实际效果	是否通过
专家管理	添加信息缺失	提示添加对应缺失 信息	提示添加对应缺失 信息	是
专家管理	修改信息错误	提示修改信息有误	提示修改信息有误	是
专家管理	按姓名查询专家	显示该姓名的专家 信息	显示该姓名的专家 信息	是
专家管理	删除专家信息	删除成功	删除成功	是

7.3.4 编辑管理模块测试

对编辑管理模块的功能进行测试，包括对编辑的增删改查。编辑管理模块的部分测试用例及结果如表 7-4 所示。

表 7-4 编辑管理模块测试表

功能模块	输入内容	预期效果	实际效果	是否通过
编辑管理	添加信息缺失	提示添加对应缺失 信息	提示添加对应缺失 信息	是
编辑管理	修改信息错误	提示修改信息有误	提示修改信息有误	是
编辑管理	按姓名查询编辑	显示该姓名的编辑 信息	显示该姓名的编辑 信息	是
编辑管理	删除编辑信息	删除成功	删除成功	是

7.3.5 系统压力测试

测试所使用的软件是免费的 Apache JMeter，模拟在多个用户进行访问的情况下系统运行的能力，设置总线程数为 1000 个，线程全部启动的时间为 20 秒，也就是每秒模拟 50 个用户对网站进行访问，通过下图可以看到系统最大响应时间为 16 秒，最小响应时间为 0 秒，错误率为 0。系统性能总体还算平稳，可以满足基本的性能需求。如图 7-1, 7-2 所示。

Label	# 样本	平均值	中位数	90% 百...	95% 百...	99% 百...	最小值	最大值	异常 %	吞吐量	接收 KB/...	发送 KB/...
HTTP 请求	1000	3	3	4	5	6	0	16	0.00%	100.1/s...	1138.56	14.08
总体	1000	3	3	4	5	6	0	16	0.00%	100.1/s...	1138.56	14.08

图 7-1 压力测试聚合报告



图 7-2 压力测试性能检测报告

7.4 测试结论

测试过程应按照规定的计划逐步进行。测试过程中不要着急，详细记录测试结果。测试时做好选择自动化测试，这样更准确、更快捷^[13]。如果采用手动测试方法，将不会如此方便，并且可能会出现一些问题，极端测试将不会出现疲劳或问题。测试时，必须十分专注，并始终关注测试结果。一旦发现异常，应及时修改。最后，应保存测试后的文档，以备将来测试。

通过测试，可以直观地感受到，当开始系统设计时，应该在编写好代码之前首先理清想法与思路。好的逻辑性的代码可以避免后续测试中的问题，也可以节省大量时间和不必要的操作。

8 结束语

基于 SpringBoot 的在线投稿系统为用户提供了一个公平、包容、易操作的平台，基本满足了用户的需求和最初的开发目标和方向。Java 语言、MySQL 数据库等技术是开发的基础。这些技术有自己的优势。学好这些技术非常重要。利用这些优点设计出来的系统运行正常、稳定，能够满足人们的需求。在详细分析了系统和各个模块的需求后，对其进行设计。最后，通过测试，系统能够正常运行，完成了在线投稿系统的设计。

在这个开发过程中使用了 Java 技术。该技术具有代码编写简单方便、对平台无要求、面向对象性好等优点，在技术上比较成熟。使用 SpringBoot 框架作为系统的主要技术支持，可以使系统正常运行并实现相应的功能。该系统在设计过程中遇到了一些困难。幸运的是，在导师和同学的帮助下，顺利完成了系统的设计。

但系统还存在一些不足之处，例如，为保护系统的安全性，用户在进行注册时应只能注册作者，专家和编辑只能通过管理员进行添加，但该系统也可以对专家和编辑进行注册。

实践证明，在线投稿系统具有良好的发展前景。经过测试和运行，系统的各项功能都非常完善，界面美观，使用方便，操作简便。它在技术理论上已经成熟。

谢 辞

时光飞逝。一转眼大学就要过去了。我一直以认真的态度投入到学习当中。虽然有竞争和眼泪，但我不仅完成了学业，而且通过不断的学习和奋斗，在各个方面都得到了发展和进步。

大学生活即将结束。虽然有很多的不舍，但我们终究还是要说再见。回忆大学的学习和生活，有泪水和汗水。在这段时间里，我对自己很严格。除了学习专业知识，我还不断扩展其它的知识面，以不同的方式从不同领域获取新知识。努力成为一名全面合格的大学生。

这次毕业设计是我独自完成的时间最长、劳动强度最高的项目。幸运的是，在设计过程中，许多同学和老师帮助了我。虽然遇到了很多大大小小的问题，但我的室友和导师经常帮助和鼓励我。

我的导师王磊老师在自身工作十分繁忙的情况下，仍然可以及时回复我发送的电子邮件，并抽时间对我进行辅导。指出我的设计错误、逻辑错误和学习计划问题。可以说，如果没有导师的帮助，我的设计将陷入死胡同。导师为我指点了迷津，就像雾中的路灯，并为我指明了方向！

参 考 文 献

- [1] 朱长纯.“互联网+”时代下出版社编辑工作的转型与升级[J].新闻传播,2022(5):96-97.
- [2] 高玮,崔晓岑.“互联网+”背景下传统出版社的融合发展道路[J].产业与科技论坛,2021,20(14):12-13.
- [3] 霍福华,韩慧.基于 SpringBoot 微服务架构下前后端分离的 MVVM 模型[J].电子技术与软件工程,2022(1):73-76.
- [4] 石宝金,张正金.基于过程文档规范化训练的《JAVAEE 实践》课程教学研究[J].河北民族师范学院学报,2020,40(2):102-107.
- [5] 魏瑾.Java 多线程技术在网络通信系统中的应用[J].山西电子技术,2022(1):66-68.
- [6] 王薇.基于 B/S 架构的电力物联网应用软件的开发设计[J].电子测试,2022(2):79-81.
- [7] Yash Arora , Raghav Arya. Java A Language which is Vast in Itself[J]. Journal of Trend in Scientific Research and Development,2020,4(4):1325-1332.
- [8] 李艳杰.基于 JAVA 与 MySQL 数据库的移动端题库练习系统的设计与实现[J].黑龙江科学,2022,13(2):56-57.
- [9] 贺适.软件开发中数据库设计理论实践研究[J].电子测试,2020(8):65-66.
- [10] 方阿丽.Web 开发主流框架技术研究[J].无线互联科技,2021,18(8):64-65+96.
- [11] 王智超,肖玉,周燕.基于需求分析的可视化技术在软件开发中的应用[J].电子技术与软件工程,2021(20):41-43.
- [12] 崔华龙.基于软件测试技术的功能测试方法和应用策略分析[J].计算机应用文摘,2022,38(5):48-50.
- [13] 杨晨 . 软件自动化测试方法的分析及应用 [J]. 现代工业经济和信息化,2022,12(1):167-168+171.