

# 华北水利水电大学实验报告

2022 — 2023 学年  
第 2 学期

课程名称:	人工智能应用程序设计 与开发
学生专业:	2020 级人工智能
学生班级:	2020185
学生姓名:	高树林
学生学号:	202018526
实验编号:	实验二

信息工程学院

# 《基于 web 的前端应用程序开发与设计》实验报告

开课实验室：实验中心 S3-101

2023 年 5 月 30 日

实验名称	图书管理系统设计与实现	成绩	
教师评语	<a href="#">SpringBoot 图书管理系统运行教程 (qq.com)</a>  <div>教师签名： 年 月 日</div>		

## 一、实验目的（根据所选择的实验填写）

- 1.了解 SpringBoot 在 web 开发中的使用。
- 2.使用数据库进行数据库的存储

## 二、实验原理（根据所选择的实验填写）

XXX

## 三、实验内容（根据所选择的实验填写，描述做了什么）

### 1. 系统概述

图书管理系统有着手工管理无法比拟的优点，如检索迅速、查找方便、可靠性高、存储量大、保密性好、成本低等，这些优点能够极大地提高图书管理的效率。因此，开发一套能够为用户提供充足的信息和快捷的查询手段的图书管理系统是非常必要的。

本系统实现了图书管理、借阅管理、用户管理、公告管理、个人中心等功能，界面友好、操作容易、维护简单，同时具备数据的完整性和安全性，符合高校图书管理系统的要求。

### 2. 开发工具及技术

- 数据表现层：Html+JavaScript+CSS+JavaEx+JQuery
- 业务逻辑层：Java+SpringBoot
- 数据持久层：MySQL+MyBatis
- 开发工具：IDEA / Eclipse

### 3. 运行环境

JDK1.8 + Maven3 + MySQL5.7

### 4. 用户分析

本系统主要用于高校图书管理，使用人群为系统管理员、普通读者。

- 系统管理员：管理整个系统的各项功能，如：公告信息、图书信息、用户信息更新维护。
- 普通读者：借阅图书、归还图书、阅览公告信息、查询编辑个人信息等。

### 5. 功能分析

#### 系统管理员：

#### 1. 首页

- 名片方式展示系统管理员拥有的权限

#### 2. 图书管理

- 图书列表：显示已上架的图书信息，可对已上架图书进行搜索、修改、删除操作
- 图书上架：录入图书信息，输入图书名称、作者、图书分类，页数，定价等数据进行图书录入

#### 3. 借阅管理

- 借阅图书：根据图书名称、作者名称，图书分类等搜索、查看、借阅图书
- 归还图书：对已经借阅的图书进行归还操作

#### 4. 用户管理

- 用户列表：显示已注册的用户，可以对已注册的用户信息进行搜索、修

改、删除操作

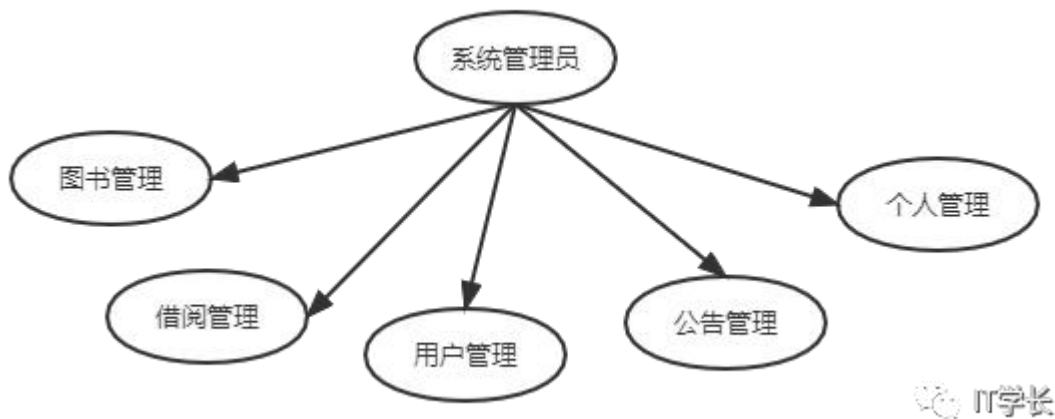
- 用户添加：录入用户的昵称、用户名、密码、生日、电话、邮箱等信息添加新用户

#### 5. 公告管理

- 公告列表：显示已发布的公告信息，可以对已发布的公告进行搜索、修改、删除等操作
- 公告发布：输入公告标题，发布时间，公告内容等要素发布公告

#### 6. 个人管理

- 个人信息详情：查看个人信息
- 个人信息修改：修改个人信息



IT学长

## 6. 数据库设计

**users:** 存储用户信息

字段名称	类型	是否为 null	是否主键	说明
id	int	否	是	ID 主键
address	varchar	是	否	地址
avatar	varchar	是	否	头像
birthday	datetime	是	否	生日
email	varchar	是	否	邮箱
is_admin	int	是	否	是否为管理员（0 管理员 1 普通用户）
nickname	varchar	是	否	昵称
password	varchar	是	否	密码
size	int	是	否	可借数量
tel	varchar	是	否	电话
username	varchar	是	否	用户名

**book:** 存储图书信息

字段名称	类型	是否为 null	是否主键	说明
id	int	否	是	ID 主键
author	varchar	是	否	图书作者
isbn	varchar	是	否	图书 ISBN 编码
name	varchar	是	否	图书名称

pages	int	是	否	图书页数
price	double	是	否	单价
publish	varchar	是	否	出版社
publish_time	datetime	是	否	出版时间
size	int	是	否	库存
translate	varchar	是	否	翻译
type	varchar	是	否	分类

**borrow:** 存储借阅信息

字段名称	类型	是否为 null	是否主键	说明
id	int	否	是	ID 主键
book_id	int	是	否	图书 ID
create_time	datetime	是	否	借阅时间
update_time	datetime	是	否	实际归还时间
user_id	int	是	否	用户 ID
end_time	datetime	是	否	归还时间
ret	int	是	否	是否归还（0 已归还 1 未归还）

**notice:** 存储公告信息

字段名称	类型	是否为 null	是否主键	说明
id	int	否	是	ID 主键
title	varchar	是	否	标题
date	datetime	是	否	发布时间
content	text	是	否	内容

7.接口示例

7.1、图书管理--添加图书

接口描述:

接口地址	/booksManageBoot/book/add
请求方式	POST
consumes	["application/json"]
produces	["*/*"]

请求示例:

```
1. {
2.   "author": "",
3.   "id": 0,
4.   "isbn": "",
5.   "name": "",
6.   "pages": 0,
7.   "price": 0,
8.   "publish": "",
9.   "publishTime": "",
```

```
10. "size": 0,
11. "translate": "",
12. "type": ""
13. }
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
book	book	body	true	Book	Book

**schema** 属性说明

**Book**

参数名称	参数说明	in	是否必须	数据类型	schema
author	图书作者	body	false	string	
id	ID 主键	body	false	integer(int32)	
isbn	图书 ISBN 编码	body	false	string	
name	图书名称	body	false	string	
pages	图书页数	body	false	integer(int32)	
price	单价	body	false	number(double)	
publish	出版社	body	false	string	
publishTime	出版时间	body	false	string(date-time)	
size	库存	body	false	integer(int32)	
translate	翻译	body	false	string	
type	分类	body	false	string	

响应示例:

```
1. {
2.   "code": 0,
3.   "data": {},
4.   "msg": ""
5. }
```

响应参数:

参数名称	参数说明	类型	schema
code	响应码	integer(int32)	integer(int32)
data	响应数据	object	
msg	响应信息	string	

响应状态:

状态码	说明	schema
200	OK	R
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	

## 7.2、借阅管理--借阅图书

接口描述:

接口地址	/booksManageBoot/book/add
请求方式	POST
consumes	["application/json"]
produces	["*/*"]

请求示例:

```
1. {
2.   "bookId": 0,
3.   "createTime": "",
4.   "endTime": "",
5.   "id": 0,
6.   "ret": 0,
7.   "updateTime": "",
8.   "userId": 0
9. }
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
borrow	borrow	body	true	Borrow	Borrow

**schema** 属性说明

### Borrow

参数名称	参数说明	in	是否必须	数据类型	schema
bookId	图书 ID	body	false	integer(int32)	
createTime	借阅时间	body	false	string(date-time)	
endTime	归还时间	body	false	string(date-time)	
id	ID 主键	body	false	integer(int32)	
ret	是否归还 (0 已归还 1 未归还)	body	false	integer(int32)	
updateTime	实际归还时间	body	false	string(date-time)	
userId	用户 ID	body	false	integer(int32)	

响应示例:

```
1. {
2.   "code": 0,
3.   "data": {},
4.   "msg": ""
5. }
```

响应参数:

参数名称	参数说明	类型	schema
code	响应码	integer(int32)	integer(int32)
data	响应数据	object	
msg	响应信息	string	

响应状态:

状态码	说明	schema
200	OK	R
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	

### 7.3、用户管理--用户列表

接口描述:

接口地址	/booksManageBoot/book/add
请求方式	POST
consumes	["application/json"]
produces	["*/*"]

请求示例:

```
1. {
2.   "currPage": 0,
3.   "keyword": "",
4.   "pageSize": 0
5. }
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
pageIn	pageIn	body	true	PageIn	PageIn

schema 属性说明

#### PageIn

参数名称	参数说明	in	是否必须	数据类型	schema
currPage	当前页	body	false	integer(int32)	
keyword	搜索关键字	body	false	string	
pageSize	每页数量	body	false	integer(int32)	

响应示例:

```
1. {
2.   "code": 0,
3.   "data": {},
4.   "msg": ""
5. }
```

响应参数:

参数名称	参数说明	类型	schema
------	------	----	--------



code	响应码	integer(int32)	integer(int32)
data	响应数据	object	
msg	响应信息	string	

响应状态:

状态码	说明	schema
200	OK	R
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	

7.4、公告管理--编辑公告

接口地址	/booksManageBoot/book/add
请求方式	POST
consumes	["application/json"]
produces	["*/*"]

请求示例:

```
1. {
2.   "content": "",
3.   "date": "",
4.   "id": 0,
5.   "title": ""
6. }
```

请求参数:

参数名称	参数说明	in	是否必须	数据类型	schema
notice	notice	body	true	Notice	Notice

schema 属性说明

Notice:

参数名称	参数说明	in	是否必须	数据类型	schema
content	公告内容	body	false	string	
date	发布时间	body	false	string(date-time)	
id	ID 主键	body	false	integer(int32)	
title	公告标题	body	false	string	

响应示例:

```
1. {
2.   "code": 0,
3.   "data": {},
4.   "msg": ""
5. }
```

响应参数:

参数名称	参数说明	类型	schema
code	响应码	integer(int32)	integer(int32)
data	响应数据	object	
msg	响应信息	string	

响应状态:

状态码	说明	schema
200	OK	R
201	Created	
401	Unauthorized	
403	Forbidden	
404	Not Found	

8. 功能展示及源码

8.1 登录页面



8.2 管理员—首页



部分代码:

```
1.<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w
```

```
3.org/TR/html4/loose.dtd">
2.<html>
3.<head>
4.<meta charset="utf-8">
5.<!--字体图标-->
6.<link href="javaex/pc/css/icomoon.css" rel="stylesheet" />
7.<!--动画-->
8.<link href="javaex/pc/css/animate.css" rel="stylesheet" />
9.<!--骨架样式-->
10.<link href="javaex/pc/css/common.css" rel="stylesheet" />
11.<!--皮肤（缇娜）-->
12.<link href="javaex/pc/css/skin/tina.css" rel="stylesheet" />
13.<!--jquery，不可修改版本-->
14.<script src="javaex/pc/lib/jquery-1.7.2.min.js"></script>
15.<!--全局动态修改-->
16.<script src="javaex/pc/js/common.js"></script>
17.<!--核心组件-->
18.<script src="javaex/pc/js/javaex.min.js"></script>
19.<!--表单验证-->
20.<script src="javaex/pc/js/javaex-formVerify.js"></script>
21.<title>图书管理系统-首页</title>
22.<style>
23.</style>
24.</head>
25.<body>
26.<!--顶部导航-->
27.<div class="admin-navbar">
28.<div class="admin-container-fluid clear">
29.<!--logo 名称-->
30.<div class="admin-logo">图书管理系统</div>
31.
32.<!--右侧-->
33.<ul class="admin-navbar-nav fr">
34.<li><input id="index_role" hidden /> <a href="javascript:;">欢迎您，
    <span
35.<id="login_user"></span></a>
36.<ul class="dropdown-menu" style="right: 10px;">
37.<li><a href="logout">退出当前账号</a></li>
38.</ul></li>
39.</ul>
40.</div>
41.</div>
42.
43.<!--主题内容-->
44.<div class="admin-mian" style="margin-bottom: 50px;">
45.<!--左侧菜单-->
```

```
46. <div class="admin-aside admin-aside-fixed">
47.   <!-- 应用标题 -->
48.   <div id="admin-toc" class="admin-toc">
49.     <div class="menu-box">
50.       <div id="menu" class="menu">
51.         <ul id="adminMenu" style="display: none;">
52.           <li class="menu-item hover"><a
53.             href="javascript:page('welcome');"><i class="icon-home2"></i>首页
54.           </a>
55.         </li>
56.         <li class="menu-item"><a href="javascript:;">图书管理<i
57.           class="icon-keyboard_arrow_left"></i></a>
58.         <ul>
59.           <li><a href="javascript:page('book/book-list');">图书列表
60.           </a></li>
61.           <li><a href="javascript:page('book/book-add');">图书上架
62.           </a></li>
63.         </ul></li>
64.         <li class="menu-item"><a href="javascript:;">借阅管理<i
65.           class="icon-keyboard_arrow_left"></i></a>
66.         <ul>
67.           <li><a href="javascript:page('borrow/book-list');">借阅图书
68.           </a></li>
69.           <li><a href="javascript:page('borrow/back');">归还图书</a></li>
70.         </ul></li>
71.         <li class="menu-item"><a href="javascript:;">用户管理<i
72.           class="icon-keyboard_arrow_left"></i></a>
73.         <ul>
74.           <li><a href="javascript:page('user/user-list');">用户列表
75.           </a></li>
76.           <li><a href="javascript:page('user/user-add');">用户添加
77.           </a></li>
78.         </ul></li>
79.         <li class="menu-item"><a href="javascript:;">公告管理<i
80.           class="icon-keyboard_arrow_left"></i></a>
81.         <ul>
82.           <li><a href="javascript:page('notice/notice-list');">公告列表
83.           </a></li>
84.           <li><a href="javascript:page('notice/notice-add');">公告发布
85.           </a></li>
86.         </ul></li>
87.       </div>
88.     </div>
89.   </div>
```

```
84.         <li class="menu-item"><a href="javascript:;">个人中心<i
85.             class="icon-keyboard_arrow_left"></i></a>
86.         <ul>
87.             <li><a href="javascript:page('my/my-info');">个人信息详情
            </a></li>
88.             <li><a href="javascript:page('my/my-update');">个人信息修改
            </a></li>
89.         </ul></li>
90.     </ul>
91.     <ul id="userMenu" style="display: none;">
92.         <li class="menu-item hover"><a
93.             href="javascript:page('welcome');"><i class="icon-home2"></i> 首页
            </a>
94.         </li>
95.
96.         <li class="menu-item"><a href="javascript:;">借阅管理<i
97.             class="icon-keyboard_arrow_left"></i></a>
98.         <ul>
99.             <li><a href="javascript:page('borrow/book-list');">借阅图书
            </a></li>
100.            <li><a href="javascript:page('borrow/back');">归还图书
            </a></li>
101.        </ul></li>
102.
103.        <li class="menu-item"><a href="javascript:;">公告管理<i
104.            class="icon-keyboard_arrow_left"></i></a>
105.        <ul>
106.            <li><a href="javascript:page('notice/notice-list');">公告列表
            </a></li>
107.        </ul></li>
108.
109.        <li class="menu-item"><a href="javascript:;">个人中心<i
110.            class="icon-keyboard_arrow_left"></i></a>
111.        <ul>
112.            <li><a href="javascript:page('my/my-info');">个人信息详情
            </a></li>
113.            <li><a href="javascript:page('my/my-update');">个人信息修改
            </a></li>
114.        </ul></li>
115.    </ul>
116. </div>
117. </div>
118. </div>
119. </div>
120.
121. <!--iframe 载入内容-->
```

```
122.     <div class="admin-markdown">
123.         <iframe id="page" src="welcome"></iframe>
124.     </div>
125. </div>
126.
127. <!-- foot -->
128. <div class="admin-footer" style="position: fixed; bottom: 0;">
129.     <div class="admin-container-fluid clear"
130.         style="line-height: 50px; text-align: center;">公众号【IT 学长】</div>
131.     </div>
132. <!-- foot -->
133. </body>
134. <script>
135.
136.     var hightUrl = "xxxx";
137.     javaex.menu({
138.         id : "menu",
139.         isAutoSelected : true,
140.         key : "",
141.         url : hightUrl
142.     });
143.
144.     $(function() {
145.         // 设置左侧菜单高度
146.         setMenuHeight();
147.     });
148.
149.     /**
150.      * 设置左侧菜单高度
151.      */
152.     function setMenuHeight() {
153.         var height = document.documentElement.clientHeight - $("#admin-toc").offset().top;
154.         heightheight = height - 10;
155.         $("#admin-toc").css("height", height+"px");
156.     }
157.
158.     // 控制页面载入
159.     function page(url) {
160.         $("#page").attr("src", url);
161.     }
162.
163.     $(document).ready(function(){
164.         // 页面一加载，读取登录用户信息
165.         $.get("user/currUser", function(data){
166.             var user = data.data;
```

```
167.         var useruserId = user.id;
168.         console.log("user",user);
169.         if (userId >0) {
170.             $("#login_user").text(user.username+"["+user.id+"]");
171.
172.             $("#index_role").val(user.isAdmin);
173.             if(user.isAdmin==0){
174.                 $("#adminMenu").css("display","block");
175.                 $("#userMenu").css("display","none");
176.             }
177.             else{
178.                 $("#adminMenu").css("display","none");
179.                 $("#userMenu").css("display","block");
180.             }
181.
182.         }else {
183.             // 找不到用户，不可进行借阅操作
184.             javaex.message({
185.                 content : "登录信息已失效，请登录后再借阅",
186.                 type : "error"
187.             });
188.         }
189.     });
190.
191. });
192. </script>
193. </html>
```

### 8.3 管理员端—图书管理

#### 图书列表

图书管理系统

欢迎您, admin[管理员]

首页

图书管理

图书列表

图书上架

借阅管理

用户管理

公告管理

个人中心

图书列表

输入图书名称、作者、ISBN等关键字

搜索

返回

序号	书名	作者	ISBN	页数	定价	出版社	出版时间	库存数量	翻译	分类	操作
1	俄国社会民主党人的任务	列宁	CYA0013	200	35	中国人民大学出版社	2022-09-26	498		马克思列宁主义、毛泽东思想	<a href="#">编辑</a> <a href="#">删除</a>
2	共产党宣言	马克思、恩格斯	CYA0012	8001	80	中国人民大学出版社	2022-09-26	23		马克思列宁主义、毛泽东思想	<a href="#">编辑</a> <a href="#">删除</a>
3	身体功能训练手册	小名	CYA0011	200	35	人民体育出版社	2022-09-26	198		文化、科学、教育、体育	<a href="#">编辑</a> <a href="#">删除</a>
4	核心体能训练	陈明	CYA0010	200	35	光线传媒出版社	2022-09-26	200	无	文化、科学、教育、体育	<a href="#">编辑</a> <a href="#">删除</a>
5	纯粹理性批判	伊曼努尔·康德	CYA009	200	35	光线传媒出版社	2022-09-26	599		哲学、宗教	<a href="#">编辑</a> <a href="#">删除</a>
6	存在与时间	马丁·海德格尔	CYA008	100	10	中国人民大学出版社	2022-09-26	600	无	哲学、宗教	<a href="#">编辑</a> <a href="#">删除</a>
7	德国民法总论	汉斯·布洛克斯	CYA007	8001	80	中国人民大学出版社	2014-09-01	124		政治、法律	<a href="#">编辑</a> <a href="#">删除</a>
8	民法思维	王泽鉴	CYA006	600	20	北京大学出版社	2019-12-06	600		政治、法律	<a href="#">编辑</a> <a href="#">删除</a>

#### 图书上架

图书管理系统

欢迎您, admin[管理员]

首页

图书管理

图书列表

图书上架

借阅管理

用户管理

公告管理

个人中心

添加图书

\* 书名

请输入图书名称

\* ISBN编号

ISBN编号

\* 作者

作者

图书分类

请选择

页数

图书页数

定价

定价

出版时间

2022-09-29

出版社

出版社

库存数量

库存数量

翻译

翻译

## 图书修改

图书管理系统

欢迎您, admin[管理员]

首页

图书管理

图书列表

图书上架

借阅管理

用户管理

公告管理

个人中心

编辑图书

\* 书名

俄国社会主义人的任务

\* ISBN编号

CYA0013

\* 作者

列宁

图书分类

马克思列宁主义、毛泽东思想、邓小平理论

页数

200

定价

35

出版时间

2022-09-26

出版社

中国人民大学出版社

库存数量

499

翻译

翻译

## 部分源码:

```

1. package com.book.manager.controller;
2.
3. import com.book.manager.entity.Book;
4. import com.book.manager.service.BookService;
5. import com.book.manager.service.BorrowService;
6. import com.book.manager.util.R;
7. import com.book.manager.util.http.CodeEnum;
8. import com.book.manager.util.ro.PageIn;
9. import io.swagger.annotations.Api;
10. import io.swagger.annotations.ApiOperation;
11. import org.springframework.beans.factory.annotation.Autowired;
12. import org.springframework.stereotype.Controller;
13. import org.springframework.web.bind.annotation.*;
14.
15. /**
16.  * @Description 用户管理

```



```
17. * @Date 2022/9/4 16:35
18. * @Author by 公众号【IT 学长】
19. */
20. @Api(tags = "图书管理")
21. @RestController
22. @RequestMapping("/book")
23. public class BookController {
24.
25.     @Autowired
26.     private BookService bookService;
27.
28.     @Autowired
29.     private BorrowService borrowService;
30.
31.     @ApiOperation("图书搜索列表")
32.     @PostMapping("/list")
33.     public R getBookList(@RequestBody PageIn pageIn) {
34.         if (pageIn == null) {
35.             return R.fail(CodeEnum.PARAM_ERROR);
36.         }
37.
38.         return R.success(CodeEnum.SUCCESS, bookService.getBookList(pageIn));
39.     }
40.
41.     @ApiOperation("添加图书")
42.     @PostMapping("/add")
43.     public R addBook(@RequestBody Book book) {
44.         return R.success(CodeEnum.SUCCESS, bookService.addBook(book));
45.     }
46.
47.     @ApiOperation("编辑图书")
48.     @PostMapping("/update")
49.     public R modifyBook(@RequestBody Book book) {
50.         return R.success(CodeEnum.SUCCESS, bookService.updateBook(book));
51.     }
52.
53.
54.     @ApiOperation("图书详情")
55.     @GetMapping("/detail")
56.     public R bookDetail(Integer id) {
57.         return R.success(CodeEnum.SUCCESS, bookService.findBookById(id));
58.     }
59.
60.     @ApiOperation("图书详情 根据 ISBN 获取")
61.     @GetMapping("/detailByIsbn")
```

```
62.     public R bookDetailByIsbn(String isbn) {
63.         return R.success(CodeEnum.SUCCESS,bookService.findBookByIsbn(isbn))
64.     };
65.
66.     @ApiOperation("删除图书")
67.     @GetMapping("/delete")
68.     public R delBook(Integer id) {
69.         bookService.deleteBook(id);
70.         borrowService.deleteBorrowByBookId(id);
71.         return R.success(CodeEnum.SUCCESS);
72.     }
73.
74. }
```

### 8.4 管理员端—借阅管理

#### 借阅图书

图书管理系统

欢迎您, admin[管理员]

首页

图书管理

借阅管理

借阅图书

归还图书

用户管理

公告管理

个人中心

序号	书名	作者	ISBN	页数	定价	出版社	出版时间	库存数量	翻译	分类	操作
1	俄国社会民主党人的任务	列宁	CYA0013	200	35	中国人民大学出版社	2022-09-26	498		马克思主义、毛泽东思想、邓小平理论	借阅
2	共产党宣言	马克思、恩格斯	CYA0012	8001	80	中国人民大学出版社	2022-09-26	23		马克思主义、毛泽东思想、邓小平理论	借阅
3	身体功能训练手册	小名	CYA0011	200	35	人民体育出版社	2022-09-26	198		文化、科学、教育、体育	借阅
4	核心体能训练	陈明	CYA0010	200	35	光线传媒出版社	2022-09-26	200	无	文化、科学、教育、体育	借阅
5	纯粹理性批判	伊曼努尔·康德	CYA009	200	35	光线传媒出版社	2022-09-26	599		哲学、宗教	借阅
6	存在与时间	马丁·海德格尔	CYA008	100	10	中国人民大学出版社	2022-09-26	600	无	哲学、宗教	借阅
7	德国民法总论	汉斯·布洛克斯	CYA007	8001	80	中国人民大学出版社	2014-09-01	124		政治、法律	借阅
8	民法思维	王泽鉴	CYA006	600	20	北京大学出版社	2019-12-06	600		政治、法律	借阅
9	民法总则	王泽鉴	CYA005	600	60	北京大学出版社	2019-12-01	300	无	政治、法律	借阅
10	政治哲学名著译丛	吴彦	CYA004	500	60	光线传媒出版社	2022-09-26	199		政治、法律	借阅

10 条/页 上一页 1 2 下一页

图书管理系统

欢迎您, admin[管理员]

首页

图书管理

借阅管理

借阅图书

归还图书

用户管理

公告管理

个人中心

图书借阅

\* ISBN编号

CYA0013

\* 书名

俄国社会民主党人的任务

作者

列宁

出版社

中国人民大学出版社

借阅时间

2022-09-27 - 2022-09-28

图书分类

1

页数

200

定价

35

库存数量

499

返回

借阅

#### 归还图书

图书管理系统欢迎您, admin(管理员)

首页

图书管理

借阅管理

借阅图书

归还图书

用户管理

公告管理

个人中心

图书归还

\* 用户名

admin

\* 昵称

系统管理员

\* 身份

管理员

\* 可借数量

0

\* 已借阅列表

序号	书名	作者	ISBN	页数	定价	出版社	分类	借阅时间	应还时间	是否逾期	操作
1	俄国社会民主党的任务	列宁	CYA0013	200	35	中国人民大学出版社	1	2022-09-27	2022-09-28	否	归还图书
2	纯粹理性批判	伊曼努尔·康德	CYA009	200	35	光线传媒出版社	2	2022-09-27	2022-09-28	否	归还图书
3	政治哲学名著译丛	吴彦	CYA004	500	60	光线传媒出版社	4	2022-09-27	2022-09-28	否	归还图书
4	身体功能训练手册	小名	CYA0011	200	35	人民体育出版社	3	2022-09-27	2022-09-28	否	归还图书
5	德国民法总论	汉斯·布洛克斯	CYA007	8001	80	中国人民大学出版社	4	2022-09-27	2022-09-28	否	归还图书

部分源码:

```

1. package com.book.manager.service;
2.
3. import cn.hutool.core.bean.BeanUtil;
4. import com.book.manager.dao.BookMapper;
5. import com.book.manager.dao.BorrowMapper;
6. import com.book.manager.dao.UsersMapper;
7. import com.book.manager.entity.Book;
8. import com.book.manager.entity.Borrow;
9. import com.book.manager.entity.Users;
10. import com.book.manager.repos.BookRepository;
11. import com.book.manager.repos.BorrowRepository;
12. import com.book.manager.util.consts.Constants;
13. import com.book.manager.util.vo.BookOut;
14. import org.springframework.beans.BeanUtils;
15. import org.springframework.beans.factory.annotation.Autowired;
16. import org.springframework.stereotype.Service;
17. import org.springframework.transaction.annotation.Transactional;
18.
19. import java.util.Date;
20. import java.util.List;
21. import java.util.Optional;
22.
23. /**
24.  * @Description 借阅管理
25.  * @Date 2022/9/4 16:35
26.  * @Author by 公众号【IT 学长】
27.  */
28. @Service
29. public class BorrowService {
30.
31.     @Autowired
32.     private BorrowRepository borrowRepository;

```

```
33.
34.     @Autowired
35.     private BorrowMapper borrowMapper;
36.
37.     @Autowired
38.     private BookService bookService;
39.
40.     @Autowired
41.     private UserService userService;
42.
43.     /**
44.      * 添加
45.      */
46.     @Transactional
47.     public Integer addBorrow(Borrow borrow) {
48.         Book book = bookService.findBook(borrow.getBookId());
49.         Users users = userService.findUserById(borrow.getUserId());
50.
51.         // 查询是否已经借阅过该图书
52.         Borrow bor = findBorrowByUserIdAndBookId(users.getId(),book.getId()
53.     );
54.         if (bor!=null) {
55.             Integer ret = bor.getRet();
56.             if (ret!=null) {
57.                 // 已借阅，未归还 不可再借
58.                 if (ret == Constants.NO) {
59.                     return Constants.BOOK_BORROWED;
60.                 }
61.             }
62.
63.             // 库存数量减一
64.             int size = book.getSize();
65.             if (size>0) {
66.                 size--;
67.                 book.setSize(size);
68.                 bookService.updateBook(book);
69.             }else {
70.                 return Constants.BOOK_SIZE_NOT_ENOUGH;
71.             }
72.
73.             // 用户可借数量减一
74.             int userSize = users.getSize();
75.             if (userSize>0) {
76.                 userSize --;
77.                 users.setSize(userSize);
```

```
78.         userService.updateUser(users);
79.     }else {
80.         return Constants.USER_SIZE_NOT_ENOUGH;
81.     }
82.
83.
84.     // 添加借阅信息，借阅默认为未归还状态
85.     borrow.setRet(Constants.NO);
86.     borrowRepository.saveAndFlush(borrow);
87.
88.     // 一切正常
89.     return Constants.OK;
90. }
91.
92. /**
93.  * user id 查询所有借阅信息
94.  */
95. public List<Borrow> findAllBorrowById(Integer userId) {
96.     return borrowRepository.findAllBorrowById(userId);
97. }
98.
99. /**
100.  * user id 查询所有 已借阅信息
101.  */
102. public List<Borrow> findBorrowByIdAndRet(Integer userId, Integer ret) {
103.     return borrowRepository.findBorrowByIdAndRet(userId, ret);
104. }
105.
106.
107. /**
108.  * 详情
109.  */
110. public Borrow findById(Integer id) {
111.     Optional<Borrow> optional = borrowRepository.findById(id);
112.     if (optional.isPresent()) {
113.         return optional.get();
114.     }
115.     return null;
116. }
117.
118. /**
119.  * 编辑
120.  */
121. public boolean updateBorrow(Borrow borrow) {
122.     return borrowMapper.updateBorrow(borrow)>0;
```

```
123.     }
124.
125.
126.     /**
127.      * 编辑
128.      */
129.     public Borrow updateBorrowByRepo(Borrow borrow) {
130.         return borrowRepository.saveAndFlush(borrow);
131.     }
132.
133.     /**
134.      * 根据 ID 删除
135.      */
136.     public void deleteBorrow(Integer id) {
137.         borrowRepository.deleteById(id);
138.     }
139.
140.     /**
141.      * 根据 book_id 删除
142.      */
143.     public void deleteBorrowByBookId(Integer bookId) {
144.         borrowMapper.deleteBorrowByBookId(bookId);
145.     }
146.
147.
148.     /**
149.      * 查询用户某一条借阅信息
150.      * @param userId 用户 id
151.      * @param bookId 图书 id
152.      */
153.     public Borrow findBorrowByUserIdAndBookId(int userId,int bookId) {
154.         return borrowMapper.findBorrowByUserIdAndBookId(userId,bookId);
155.     }
156.
157.     /**
158.      * 归还书籍,
159.      * @param userId 用户 Id
160.      * @param bookId 书籍 id
161.      */
162.     @Transactional(rollbackFor = Exception.class)
163.     public void retBook(int userId,int bookId) {
164.         // 用户可借数量加 1
165.         Users user = userService.findUserById(userId);
166.         Integer size = user.getSize();
167.         size++;
```

```

168.         user.setSize(size);
169.         userService.updateUser(user);
170.
171.
172.         // 书籍库存加 1
173.         Book book = bookService.findBook(bookId);
174.         Integer bookSize = book.getSize();
175.         bookSize++;
176.         book.setSize(bookSize);
177.         bookService.updateBook(book);
178.         // 借阅记录改为已归还,删除记录
179.         Borrow borrow = this.findBorrowByUserIdAndBookId(userId, bookId)
180.         ;
181.         this.deleteBorrow(borrow.getId());
182.     }
183. }

```

## 8.5 管理员端--用户管理

### 用户列表

图书管理系统

欢迎, admin[管理员]

首页

图书管理

借阅管理

用户管理

用户添加

公告管理

个人中心

用户列表

返回

搜索用户名或者昵称

搜索

序号	昵称	用户名	生日	电话	邮箱	地址	可借数量	身份	操作
1	系统管理员	admin	2022-09-30	18888888888	cya@163.com	四川成都	0	管理员	编辑 删除
2	普通用户1	user1	2022-09-26	18888888888	user1@163.com	四川成都	5	普通用户	编辑 删除
3	普通用户2	user2	2022-09-26	1773832315	user2@163.com	四川德阳	6	普通用户	编辑 删除
4	普通用户3	user3	2022-09-26	18888888888	user3@163.com	四川德阳	5	普通用户	编辑 删除
5	普通用户4	user4	2022-09-26	18888888888	user4@163.com	四川成都	5	普通用户	编辑 删除

用户添加:

图书管理系统

欢迎您, admin[管理员]

首页

图书管理

借阅管理

用户管理

用户列表

用户添加

公告管理

个人中心

添加用户

\* 昵称

\* 用户名

\* 密码

生日

2022-09-27

电话

\* 身份

请选择

邮箱

地址

\* 可借数量

返回

保存

用户修改:

图书管理系统

欢迎您, admin[管理员]

首页

图书管理

借阅管理

用户管理

用户列表

用户添加

公告管理

个人中心

编辑用户信息

\* 昵称

系统管理员

\* 用户名

admin

\* 密码

\*\*\*\*\*

生日

2022-09-30

电话

18888888888

\* 身份

管理员

邮箱

cya@163.com

地址

四川成都

\* 可借数量

0

返回

保存

部分源码:

```
1. package com.book.manager.service;
2.
3. import cn.hutool.core.bean.BeanUtil;
4. import com.book.manager.dao.BookMapper;
5. import com.book.manager.dao.BorrowMapper;
6. import com.book.manager.dao.UsersMapper;
7. import com.book.manager.entity.Book;
8. import com.book.manager.entity.Borrow;
9. import com.book.manager.entity.Users;
10. import com.book.manager.repos.BookRepository;
11. import com.book.manager.repos.BorrowRepository;
12. import com.book.manager.util.consts.Constants;
13. import com.book.manager.util.vo.BookOut;
14. import org.springframework.beans.BeanUtils;
15. import org.springframework.beans.factory.annotation.Autowired;
```



```
16. import org.springframework.stereotype.Service;
17. import org.springframework.transaction.annotation.Transactional;
18.
19. import java.util.Date;
20. import java.util.List;
21. import java.util.Optional;
22.
23. /**
24.  * @Description 借阅管理
25.  * @Date 2022/9/4 16:35
26.  * @Author by 公众号【IT 学长】
27.  */
28. @Service
29. public class BorrowService {
30.
31.     @Autowired
32.     private BorrowRepository borrowRepository;
33.
34.     @Autowired
35.     private BorrowMapper borrowMapper;
36.
37.     @Autowired
38.     private BookService bookService;
39.
40.     @Autowired
41.     private UserService userService;
42.
43.     /**
44.      * 添加
45.      */
46.     @Transactional
47.     public Integer addBorrow(Borrow borrow) {
48.         Book book = bookService.findBook(borrow.getBookId());
49.         Users users = userService.findUserById(borrow.getUserId());
50.
51.         // 查询是否已经借阅过该图书
52.         Borrow bor = findBorrowByUserIdAndBookId(users.getId(), book.getId()
53.         );
54.         if (bor!=null) {
55.             Integer ret = bor.getRet();
56.             if (ret!=null) {
57.                 // 已借阅, 未归还 不可再借
58.                 if (ret == Constants.NO) {
59.                     return Constants.BOOK_BORROWED;
60.                 }
61.             }
62.         }
63.     }
64. }
```

```
61.     }
62.
63.     // 库存数量减一
64.     int size = book.getSize();
65.     if (size>0) {
66.         size--;
67.         book.setSize(size);
68.         bookService.updateBook(book);
69.     }else {
70.         return Constants.BOOK_SIZE_NOT_ENOUGH;
71.     }
72.
73.     // 用户可借数量减一
74.     int userSize = users.getSize();
75.     if (userSize>0) {
76.         userSize --;
77.         users.setSize(userSize);
78.         userService.updateUser(users);
79.     }else {
80.         return Constants.USER_SIZE_NOT_ENOUGH;
81.     }
82.
83.
84.     // 添加借阅信息，借阅默认为未归还状态
85.     borrow.setRet(Constants.NO);
86.     borrowRepository.saveAndFlush(borrow);
87.
88.     // 一切正常
89.     return Constants.OK;
90. }
91.
92. /**
93.  * user id 查询所有借阅信息
94.  */
95. public List<Borrow> findAllBorrowById(Integer userId) {
96.     return borrowRepository.findAllBorrowById(userId);
97. }
98.
99. /**
100.  * user id 查询所有 已借阅信息
101.  */
102. public List<Borrow> findBorrowByIdAndRet(Integer userId, Integer ret) {
103.     return borrowRepository.findBorrowByIdAndRet(userId,ret);
104. }
105.
```

```
106.
107.     /**
108.      * 详情
109.      */
110.     public Borrow findById(Integer id) {
111.         Optional<Borrow> optional = borrowRepository.findById(id);
112.         if (optional.isPresent()) {
113.             return optional.get();
114.         }
115.         return null;
116.     }
117.
118.     /**
119.      * 编辑
120.      */
121.     public boolean updateBorrow(Borrow borrow) {
122.         return borrowMapper.updateBorrow(borrow)>0;
123.     }
124.
125.
126.     /**
127.      * 编辑
128.      */
129.     public Borrow updateBorrowByRepo(Borrow borrow) {
130.         return borrowRepository.saveAndFlush(borrow);
131.     }
132.
133.     /**
134.      * 根据 ID 删除
135.      */
136.     public void deleteBorrow(Integer id) {
137.         borrowRepository.deleteById(id);
138.     }
139.
140.     /**
141.      * 根据 book_id 删除
142.      */
143.     public void deleteBorrowByBookId(Integer bookId) {
144.         borrowMapper.deleteBorrowByBookId(bookId);
145.     }
146.
147.
148.     /**
149.      * 查询用户某一条借阅信息
150.      * @param userId 用户 id
151.      * @param bookId 图书 id
```

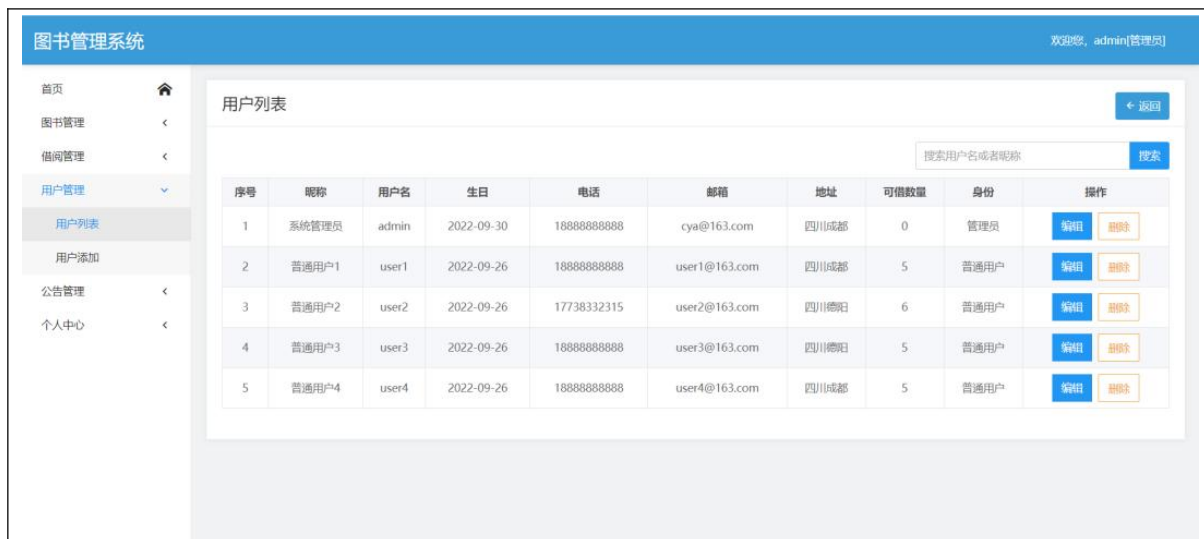
```

152.         */
153.         public Borrow findBorrowByUserIdAndBookId(int userId,int bookId) {
154.             return borrowMapper.findBorrowByUserIdAndBookId(userId,bookId);
155.         }
156.
157.         /**
158.          * 归还书籍,
159.          * @param userId 用户 Id
160.          * @param bookId 书籍 id
161.          */
162.         @Transactional(rollbackFor = Exception.class)
163.         public void retBook(int userId,int bookId) {
164.             // 用户可借数量加 1
165.             Users user = userService.findUserById(userId);
166.             Integer size = user.getSize();
167.             size++;
168.             user.setSize(size);
169.             userService.updateUser(user);
170.
171.
172.             // 书籍库存加 1
173.             Book book = bookService.findBook(bookId);
174.             Integer bookSize = book.getSize();
175.             bookSize++;
176.             book.setSize(bookSize);
177.             bookService.updateBook(book);
178.             // 借阅记录改为已归还,删除记录
179.             Borrow borrow = this.findBorrowByUserIdAndBookId(userId, bookId)
180.             ;
181.             this.deleteBorrow(borrow.getId());
182.         }
183.     }

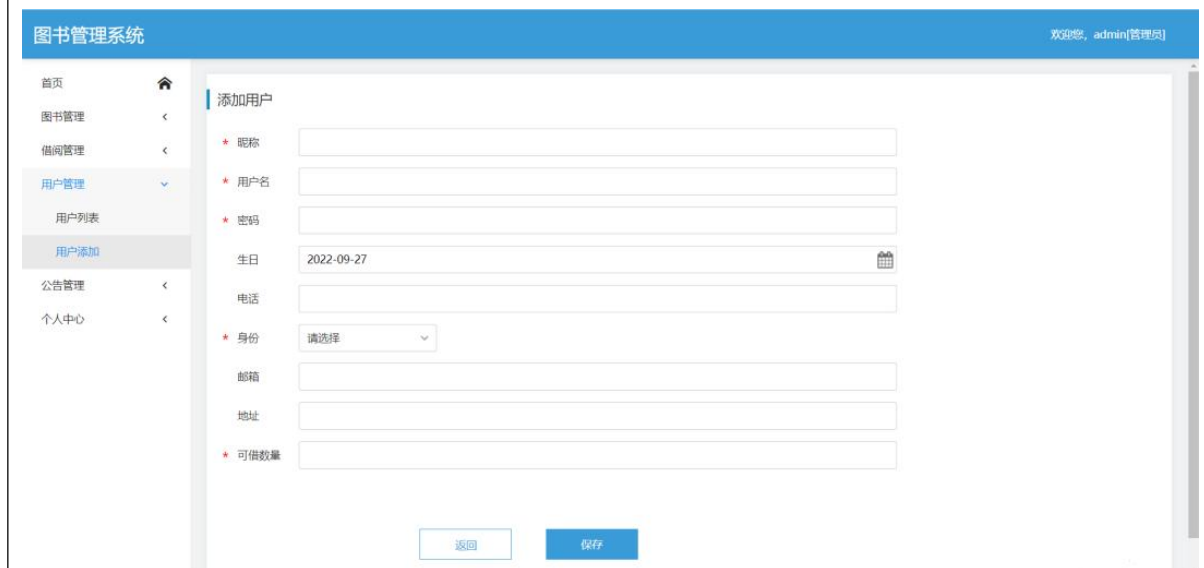
```

## 8.5 管理员端—用户管理

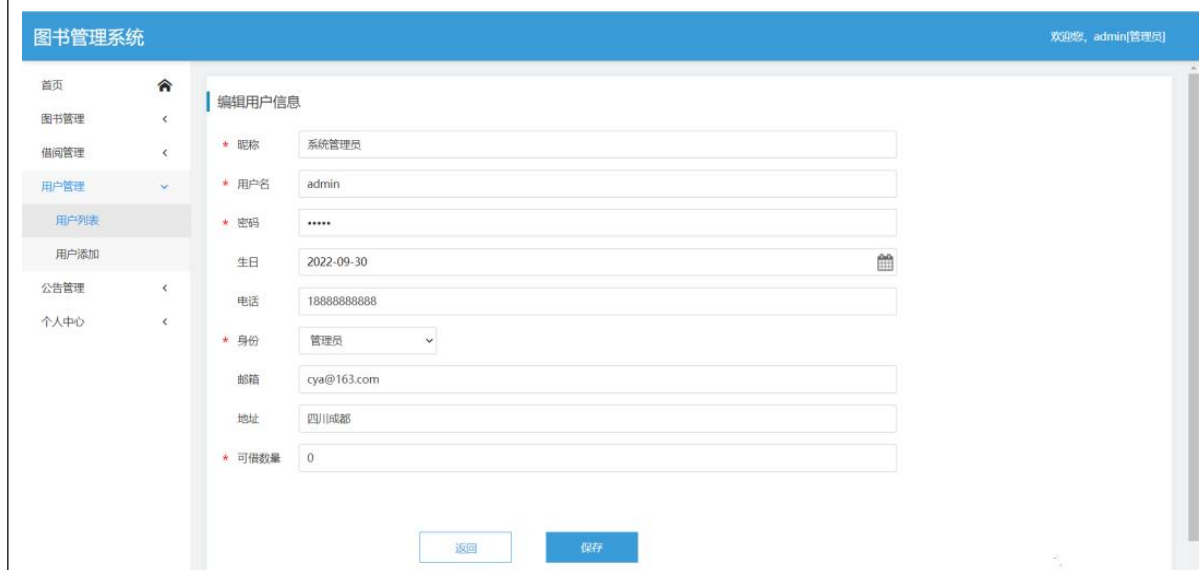
用户列表：



用户添加：



用户修改：



部分源码：

```
1. <?xml version="1.0" encoding="UTF-8"?>
```

```
2. <!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis
   .org/dtd/mybatis-3-mapper.dtd">
3. <mapper namespace="com.book.manager.dao.UsersMapper">
4.   <resultMap id="BaseResultMap"
5.     type="com.book.manager.entity.Users">
6.     <id column="id" property="id" jdbcType="INTEGER" />
7.     <result column="avatar" property="avatar" jdbcType="VARCHAR" />
8.     <result column="nickname" property="nickname"
9.       jdbcType="VARCHAR" />
10.    <result column="username" property="username"
11.      jdbcType="VARCHAR" />
12.    <result column="password" property="password"
13.      jdbcType="VARCHAR" />
14.    <result column="birthday" property="birthday"
15.      jdbcType="TIMESTAMP" />
16.    <result column="is_admin" property="isAdmin" jdbcType="INTEGER" />
17.    <result column="tel" property="tel" jdbcType="VARCHAR" />
18.    <result column="email" property="email" jdbcType="VARCHAR" />
19.    <result column="address" property="address" jdbcType="VARCHAR" />
20.    <result column="size" property="size" jdbcType="INTEGER" />
21.  </resultMap>
22.
23.  <!-- 模糊查询 -->
24.  <select id="findListByLike" resultMap="BaseResultMap">
25.    select * from users
26.    <where>
27.      <if test="_parameter != null and _parameter != ''">
28.        and `username` like CONCAT('%',{_parameter},'%') or `nickname` like
29.        CONCAT('%',{_parameter},'%')
30.      </if>
31.    </where>
32.  </select>
33.
34.  <update id="updateUsers" parameterType="map">
35.    update users
36.    <set>
37.      <if test="avatar != null and avatar != ''">
38.        avatar = #{avatar},
39.      </if>
40.      <if test="nickname != null and nickname != ''">
41.        nickname = #{nickname},
42.      </if>
43.      <if test="username != null and username != ''">
44.        username = #{username},
45.      </if>
46.      <if test="password != null and password != ''">
```

```

47.     password = #{password},
48. </if>
49. <if test="birthday != null">
50.     birthday = #{birthday},
51. </if>
52. <if test="isAdmin != null">
53.     is_admin = #{isAdmin},
54. </if>
55. <if test="tel != null and tel != ''">
56.     tel = #{tel},
57. </if>
58. <if test="email != null and email != ''">
59.     email = #{email},
60. </if>
61. <if test="address != null and address != ''">
62.     address = #{address},
63. </if>
64. <if test="size != null">
65.     `size` = #{size},
66. </if>
67. </set>
68. where id = #{id}
69. </update>
70. </mapper>

```

## 8.6 管理员端—公告管理

公告列表：

图书管理系统

欢迎您, admin(管理员)

首页

图书管理

借阅管理

用户管理

公告管理

公告列表

公告发布

个人中心

公告列表

返回

搜索公告标题

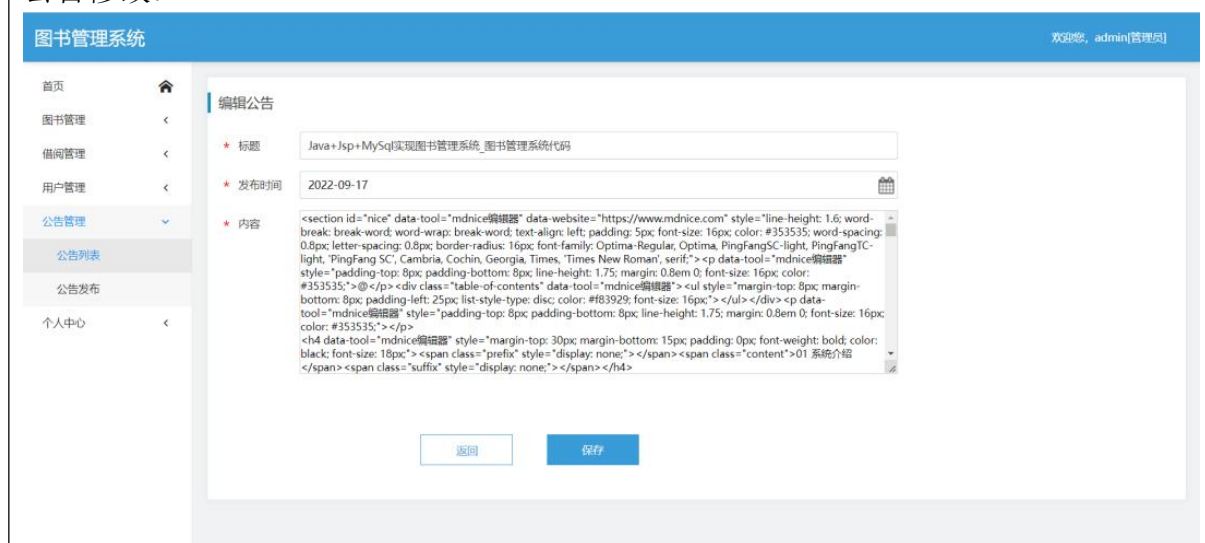
搜索

序号	标题	发布时间	操作
1	Java+Jsp+MySQL实现图书管理系统_图书管理系统代码	2022-09-17	<a href="#">详情</a> <a href="#">编辑</a> <a href="#">删除</a>
2	Java+Spring+SpringMVC+Mybatis+Vue宿舍管理系统	2022-09-27	<a href="#">详情</a> <a href="#">编辑</a> <a href="#">删除</a>
3	Java1.8 + SSM + AngularJS+MySQL机震管理系统	2022-09-27	<a href="#">详情</a> <a href="#">编辑</a> <a href="#">删除</a>
4	关于SpringBoot图书管理系统发布的通知	2022-09-27	<a href="#">详情</a> <a href="#">编辑</a> <a href="#">删除</a>

公告详情：



公告修改:



部分源码:

```
1. package com.book.manager.controller;
2.
3. import cn.hutool.core.bean.BeanUtil;
4. import cn.hutool.core.date.DateUtil;
5. import cn.hutool.core.util.StrUtil;
6.
7. import com.book.manager.entity.Notice;
8. import com.book.manager.entity.Users;
9. import com.book.manager.service.NoticeService;
10. import com.book.manager.util.R;
11. import com.book.manager.util.consts.Constants;
12. import com.book.manager.util.consts.ConvertUtil;
13. import com.book.manager.util.http.CodeEnum;
14. import com.book.manager.util.vo.NoticeOut;
15. import com.book.manager.util.vo.PageOut;
16. import com.book.manager.util.vo.UserOut;
17. import com.book.manager.util.ro.PageIn;
```



```
18. import com.github.pagehelper.PageInfo;
19. import io.swagger.annotations.Api;
20. import io.swagger.annotations.ApiOperation;
21. import org.springframework.beans.BeanUtils;
22. import org.springframework.beans.factory.annotation.Autowired;
23. import org.springframework.security.core.context.SecurityContextHolder;
24. import org.springframework.web.bind.annotation.*;
25.
26. import java.util.ArrayList;
27. import java.util.List;
28. import java.util.Map;
29.
30. /**
31.  * @Description 公告管理
32.  * @Date 2022/9/4 16:35
33.  * @Author by 公众号【IT 学长】
34.  */
35. @Api(tags = "公告管理")
36. @RestController
37. @RequestMapping("/notice")
38. public class NoticeController {
39.
40.     @Autowired
41.     private NoticeService noticeService;
42.
43.     @ApiOperation("公告列表")
44.     @PostMapping("/list")
45.     public R getNotices(@RequestBody PageIn pageIn) {
46.         if (pageIn == null) {
47.             return R.fail(CodeEnum.PARAM_ERROR);
48.         }
49.         // 封装分页出参对象
50.         PageInfo<Notice> noticeList = noticeService.getNoticeList(pageIn);
51.
52.         PageOut pageOut = new PageOut();
53.         pageOut.setCurrPage(noticeList.getPageNum());
54.         pageOut.setPageSize(noticeList.getPageSize());
55.         pageOut.setTotal((int) noticeList.getTotal());
56.         List<Notice> outs = new ArrayList<>();
57.         for (Notice notice : noticeList.getList()) {
58.             NoticeOut out = new NoticeOut();
59.             BeanUtils.copyProperties(notice, out);
60.             out.setDate1(DateUtil.format(notice.getDate(), Constants.DATE_FORMAT));
61.             outs.add(out);
62.         }
```

```
62.         pageOut.setList(outs);
63.
64.         return R.success(CodeEnum.SUCCESS,pageOut);
65.     }
66.
67.     @ApiOperation("添加公告")
68.     @PostMapping("/addNoticer")
69.     public R addNoticer(@RequestBody Notice notice) {
70.         if (notice == null) {
71.             return R.fail(CodeEnum.PARAM_ERROR);
72.         }
73.         return R.success(CodeEnum.SUCCESS,noticeService.addNotice(notice));
74.     }
75.
76.
77.     @ApiOperation("编辑公告")
78.     @PostMapping("/update")
79.     public R modifyNotice(@RequestBody Notice notice) {
80.         return R.success(CodeEnum.SUCCESS,noticeService.updateNotice(notice
81.     ));
82.     }
83.
84.     @ApiOperation("公告详情")
85.     @GetMapping("/detail")
86.     public R noticeDetail(Integer id) {
87.         Notice notice = noticeService.findNoticeById(id);
88.         if (notice!=null) {
89.             NoticeOut out = new NoticeOut();
90.             BeanUtils.copyProperties(notice,out);
91.             out.setDate1(DateUtil.format(notice.getDate(),Constants.DATE_FO
92. RMAT));
93.             return R.success(CodeEnum.SUCCESS,out);
94.         }
95.         return R.fail(CodeEnum.NOT_FOUND);
96.     }
97.
98.     @ApiOperation("删除公告")
99.     @GetMapping("/delete")
100.    public R delNotice(Integer id) {
101.        noticeService.deleteNotice(id);
102.        return R.success(CodeEnum.SUCCESS);
103.    }
104.
```

105. }

## 8.7 管理员端个人中心

个人信息详情：

图书管理系统

欢迎您, admin[管理员]

首页

图书管理

借阅管理

用户管理

公告管理

个人中心

个人信息详情

个人信息修改

个人信息

\* 昵称

系统管理员

\* 用户名

admin

\* 密码

\*\*\*\*\*

生日

2022-09-30

电话

18888888888

身份

管理员

邮箱

cya@163.com

地址

四川成都

可借数量

0

部分源码：

```
1.<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w
  3.org/TR/html4/loose.dtd">
2.<html>
3.<head>
4.<meta charset="utf-8">
5.<!--字体图标-->
6.<link href="../javaex/pc/css/icomoon.css" rel="stylesheet" />
7.<!--动画-->
8.<link href="../javaex/pc/css/animate.css" rel="stylesheet" />
9.<!--骨架样式-->
10.<link href="../javaex/pc/css/common.css" rel="stylesheet" />
11.<!--皮肤（缇娜）-->
12.<link href="../javaex/pc/css/skin/tina.css" rel="stylesheet" />
13.<!--jquery, 不可修改版本-->
14.<script src="../javaex/pc/lib/jquery-1.7.2.min.js"></script>
15.<!--全局动态修改-->
16.<script src="../javaex/pc/js/common.js"></script>
17.<!--核心组件-->
18.<script src="../javaex/pc/js/javaex.min.js"></script>
19.<!--表单验证-->
20.<script src="../javaex/pc/js/javaex-formVerify.js"></script>
21.<title>图书管理系统-查询个人信息</title>
22.</head>
23.<style>
24.</style>
25.
26.<body>
27.<!--全部主体内容-->
```

```
28. <div class="list-content">
29.   <!--块元素-->
30.   <div class="block">
31.     <!--修饰块元素名称-->
32.     <div class="banner">
33.       <p class="tab fixed">个人信息</p>
34.     </div>
35.
36.   <!--正文内容-->
37.   <div class="main">
38.     <form id="form">
39.       <!--文本框-->
40.       <div class="unit clear">
41.         <div class="left">
42.           <span class="required">*</span>
43.           <p class="subtitle">昵称</p>
44.         </div>
45.         <div class="right">
46.           <input type="text" class="text" id="nickname" name="nickname"
47.             readonly="readonly" />
48.         </div>
49.       </div>
50.
51.       <div class="unit clear">
52.         <div class="left">
53.           <span class="required">*</span>
54.           <p class="subtitle">用户名</p>
55.         </div>
56.         <div class="right">
57.           <input type="text" class="text" id="username" name="username"
58.             readonly="readonly" />
59.         </div>
60.       </div>
61.
62.       <div class="unit clear">
63.         <div class="left">
64.           <span class="required">*</span>
65.           <p class="subtitle">密码</p>
66.         </div>
67.         <div class="right">
68.           <input type="password" class="text" id="password" name="password"
69.             readonly="readonly" />
70.         </div>
71.       </div>
72.
73.     <div class="unit clear">
```

```
74.     <div class="left">
75.         <span class="required"></span>
76.         <p class="subtitle">生日</p>
77.     </div>
78.     <div class="right">
79.         <input type="text" class="text" id="birthday" name="birthday"
80.             readonly="readonly" />
81.     </div>
82. </div>
83.
84. <div class="unit clear">
85.     <div class="left">
86.         <span class="required"></span>
87.         <p class="subtitle">电话</p>
88.     </div>
89.     <div class="right">
90.         <input type="text" class="text" id="tel" name="tel"
91.             readonly="readonly" />
92.     </div>
93. </div>
94.
95. <!-- 下拉选择框-->
96. <div class="unit clear">
97.     <div class="left">
98.         <span class="required"></span>
99.         <p class="subtitle">身份</p>
100.    </div>
101.    <div class="right">
102.        <input type="text" class="text" id="ident" name="ident"
103.            readonly="readonly" />
104.    </div>
105. </div>
106.
107. <div class="unit clear">
108.     <div class="left">
109.         <span class="required"></span>
110.         <p class="subtitle">邮箱</p>
111.     </div>
112.     <div class="right">
113.         <input type="text" class="text" id="email" name="email"
114.             readonly="readonly" />
115.     </div>
116. </div>
117.
118. <div class="unit clear">
119.     <div class="left">
```

```
120.         <span class="required"></span>
121.         <p class="subtitle">地址</p>
122.     </div>
123.     <div class="right">
124.         <input type="text" class="text" id="address" name="address"
125.             readonly="readonly" />
126.     </div>
127. </div>
128.
129.     <div class="unit clear">
130.         <div class="left">
131.             <span class="required"></span>
132.             <p class="subtitle">可借数量</p>
133.         </div>
134.         <div class="right">
135.             <input type="text" class="text" id="size" name="size"
136.                 readonly="readonly" />
137.         </div>
138.     </div>
139.     <br> <br>
140.
141. </form>
142. </div>
143. </div>
144. </div>
145.
146. <script>
147.     javaex.select({
148.         id : "select",
149.         isSearch : false
150.     });
151.
152.     // 页面加载执行
153.     $(document).ready(function() {
154.         // get 读取参数
155.         $.get("../user/currUser", function(data) {
156.             var user = data.data;
157.             $("#nickname").val(user.nickname);
158.             $("#username").val(user.username);
159.             $("#password").val(user.password);
160.             $("#birthday").val(user.birthday);
161.             $("#tel").val(user.tel);
162.             $("#email").val(user.email);
163.             $("#address").val(user.address);
164.             $("#size").val(user.size);
165.             $("#ident").val(user.ident);
```

```
166.
167.     return false;
168. });
169. });
170. </script>
171. </body>
172. </html>
```

### 8.8 普通用户端—首页



### 8.9 普通用户端—借阅管理

借阅图书:



图书管理系统

欢迎您, user1[普通用户]

首页

借阅管理

借阅图书

归还图书

公告管理

个人中心

图书借阅

ISBN编号

CYA0013

书名

俄国社会民主党人的任务

作者

列宁

出版社

中国人民大学出版社

借阅时间

2023-02-16 - 2023-02-17

图书分类

1

页数

200

定价

35

库存数量

498

借阅成功

返回

借阅

归还图书:

图书管理系统

欢迎您, user1[普通用户]

首页

借阅管理

借阅图书

归还图书

公告管理

个人中心

图书归还

用户名

user1

昵称

普通用户1

身份

普通用户

可借数量

3

已借阅列表

序号	书名	作者	ISBN	页数	定价	出版社	分类	借阅时间	应还时间	是否逾期	操作
1	俄国社会民主党人的任务	列宁	CYA0013	200	35	中国人民大学出版社	1	2023-02-16	2023-02-17	否	归还图书
2	共产党宣言	马克思、恩格斯	CYA0012	8001	80	中国人民大学出版社	1	2023-02-16	2023-02-17	否	归还图书

## 8.10 普通用户端—公告管理

图书管理系统

欢迎您, user1[普通用户]

首页

借阅管理

公告管理

公告列表

个人中心

公告列表

返回

搜索公告标题

搜索

序号	标题	发布时间	操作
1	Java + Jsp + MySQL实现图书管理系统_图书管理系统代码	2022-09-17	详情
2	Java + Spring + SpringMVC + Mybatis + VUE宿舍管理系统	2022-09-27	详情
3	Java1.8 + SSM + AngularJS + MySQL机票管理系统	2022-09-27	详情
4	关于SpringBoot图书管理系统发布的通知	2022-09-27	详情



#### 四、实验工具（根据所选择的实验及采用的工具填写）

软件	版本	说明
IDEA	2020.1	开发工具版本无要求
JDK	1.8	推荐使用 JDK1.8
MySQL	5.7.26	使用 MySQL5.5+，强烈推荐使用 MySQL5.7.x
Navicat Premium	12.0.16	版本无要求
Maven	3.6.3	版本无要求

#### 五、实验步骤（根据所选择的实验填写，描述如何做的：思路，算法，是否自己编码，用到了哪些库和类；代码解释）

步骤 1:

六、实验结果及分析（附上实验结果截图，并分析实验结果）