

**本 科 毕 业 论 文**

基于VUE的智慧图书管理平台

**Intelligent book management platform based on VUE.**

学院名称： 计算机科学与信息工程学院

专业班级： 网络工程14-1

学生姓名： 宋 将 炜

学生学号： 14031110105

指导教师姓名： 张 炜

指导教师职称： 讲 师

2018年5 月

**毕业设计（论文）原创性声明和使用授权说明**

**原创性声明**

本人郑重承诺：所呈交的毕业设计（论文），是我个人在指导教师的指导下进行的研究工作及取得的成果。尽我所知，除文中特别加以标注和致谢的地方外，不包含其他人或组织已经发表或公布过的研究成果，也不包含我为获得安阳工学院及其它教育机构的学位或学历而使用过的材料。对本研究提供过帮助和做出过贡献的个人或集体，均已在文中作了明确的说明并表示了谢意。

作 者 签 名：　 　 　　 日　 期：

指导教师签名：　　 　 　 日　 期：

**使用授权说明**

本人完全了解安阳工学院关于收集、保存、使用毕业设计（论文）的规定，即：按照学校要求提交毕业设计（论文）的印刷本和电子版本；学校有权保存毕业设计（论文）的印刷本和电子版，并提供目录检索与阅览服务；学校可以采用影印、缩印、数字化或其它复制手段保存论文；在不以赢利为目的前提下，学校可以公布论文的部分或全部内容。

作者签名：　 　 　 日　 期：

目 录

[引 言 1](#_Toc23195)

[第1章 绪论 2](#_Toc22085)

[1.1 课题内容 2](#_Toc25947)

[1.2 课题背景 2](#_Toc6723)

[1.3 课题意义 3](#_Toc28671)

[1.4 开发环境 3](#_Toc22757)

[1.5 开发平台介绍 4](#_Toc864)

[1.5.1 Visual Studio Code 4](#_Toc7508)

[1.5.2 Element UI 4](#_Toc8956)

[1.5.3 VUE技术栈 4](#_Toc30825)

[1.5.4 Node.JS 5](#_Toc25123)

[1.5.5 Express 5](#_Toc20326)

[1.5.6 MongoDB 5](#_Toc3831)

[1.6 本章小结 6](#_Toc26150)

[第2章 系统设计 7](#_Toc23332)

[2.1 系统需求整体设计 7](#_Toc6136)

[2.2 系统需求实现的可行性分析 8](#_Toc11209)

[2.3 程序功能实现具体设计 9](#_Toc24695)

[2.4 数据库设计 9](#_Toc31315)

[2.5 用例图 11](#_Toc8692)

[2.6 E-R模型图 13](#_Toc3651)

[2.7 系统流程图 13](#_Toc24898)

[2.8 本章小结 14](#_Toc23090)

[第3章 系统功能具体实现 15](#_Toc14216)

[3.1登陆注册功能 15](#_Toc11594)

[3.2超级管理员功能模块 16](#_Toc12325)

[3.2.1 个人信息查看模块 16](#_Toc4830)

[3.2.2 用户管理模块 16](#_Toc8647)

[3.2.3 图书管理模块 18](#_Toc9472)

[3.2.4 公告管理模块 22](#_Toc19429)

[3.3模块管理员模块 23](#_Toc17464)

[3.4普通用户模块 24](#_Toc22574)

[第4章 系统测试 27](#_Toc6573)

[4.1什么是黑盒测试 27](#_Toc16506)

[4.2 黑盒测试环境 27](#_Toc8167)

[4.3 黑盒测试方式 27](#_Toc1302)

[4.4 测试环节 28](#_Toc3669)

[4.4.1超管用户的权限及系统功能 28](#_Toc16039)

[4.4.2 模块管理员的权限及系统功能 28](#_Toc9778)

[4.4.3 普通用户模块功能测试 28](#_Toc12285)

[4.4.3 登陆注册模块测试 29](#_Toc5004)

[4.4 测试结果 29](#_Toc24448)

[4.5 本章小结 30](#_Toc27504)

[致 谢 32](#_Toc15659)

[参考文献 33](#_Toc22666)

基于VUE的智慧图书管理平台

摘要**：**随着社会的发展，互联网的普及以及互联网技术的发展，生活中越来越多的地方开始使用互联网技术进行资源和人力的管理，互联网应用在带给人们带来巨大便利的同时，大大释放了劳动生产力。本文所阐述的课题是借助当今火热的WEB技术搭建的智慧型图书管理平台。如今，在互联网快速普及及高速发展的背景下，网络已经成为人们日常生活的重要组成部分，如今人们有很多的工作都需要借助网络平台来完成，在大学中，互联网的基础建设并不是太全面，学生获取知识的途径主要通过教师的课堂讲解以及图书馆的典籍，两者中最有效的是第一种，但是教师讲解是有时间空间，系统上的局限性，图书馆便作为学生获取课堂外知识等的一种主要途径。

对于高校的图书馆，藏书典籍数量数不胜数，但是却没有一个合理的定位，学生并不能通过一个途径知道图书馆目前存在什么书籍，图书信息一般无法共享到学生中去，藏书种类以及借阅情况都是不可知的，学生找到想要的书籍消耗时间较多，为了能更好的发挥图书馆的作用，节省学生的借书时间，建立一个小型智慧型图书管理平台是很有必要的，它可以帮助很多同学解决借阅图书过程中遇到的问题，节省大家的时间，更大效率的发挥图书馆存在的价值以及作用。

本文较为全面的向大家介绍了整个系统的实现过程以及用到的技术栈，基于我在学校以及工作中的所学所知，本系统主要选用界面风格统一，高度可扩展型UI框架Element UI，基于Node.JS的Express框架以及数据存储的[MongoDB](http://www.baidu.com/link?url=Ogj_AWz9CPoDmiqj9JdRYRg0WeyfqVEdStCxxofxREm2IUoVuHNWu1F937nbVmJC" \t "https://www.baidu.com/_blank)非关系型数据库，还有如今流行的MVVM框架VUE技术栈作为整体的技术支持，进行整体的系统开发。

本文主要从上述的技术栈以及软件开发流程进行文章编排，尽可能详细的向大家介绍整个系统的实现，主要包含用户注册登陆，密码修改，图书管理，人员管理，公告管理，借阅归还等主要功能。

**关键词：**VUE技术栈；智慧图书管理；MongoDB；Node.JS

Intelligent book management platform based on VUE.

**Abstract:** with the development of the society, the popularity of the Internet and the development of Internet technology, life more and more places begin using the management of the resources and manpower of the Internet technology, Internet applications while give people bring great convenience, the release of the labor productivity greatly. The topic of this paper is the platform of the intelligent library built by the hot WEB technology. Today, under the background of popularization and the quick development of Internet, the Internet has become an important component of People's Daily life, people a lot of work needs to be done with the help of a network platform, in the university, the infrastructure of the Internet is not too full, students acquire knowledge mainly by the way of a teacher's lectures and library books, is the first, which is the most effective but teachers explain there is a time space, the limitation of the system, the library as the students obtain the knowledge such as a main way outside the classroom.

Number of library in colleges and universities, the collection of books, but not a reasonable orientation, students cannot pass what there is a way to know the library books, books are usually placed on the shelf, titles and lending conditions are unknown, to find the books to relatively time consuming more, thus building a small book management platform for the problem is very be necessary, can help many students to solve the problems in the process of borrowing books.

This paper are to introduce you to the implementation of the whole system process and stack used technology, based on what I learned in school and work, this system mainly USES the following knowledge technology, as the front-end unified interface style Element type extensible framework of UI, based on the Node. JS language background and directing the relational database of data storage, front desk language to choose the popular MVVM framework VUE technology stack.

**Key words:** VUE technology stack; Management of smart books; MongoDB; Node. JS

# 

# **引 言**

随着互联网的快速发展以及网络化的普及，生活中越来越多的地方开始采用计算机应用，用来协助人们日程的生活工作，计算机的应用现在已经深入各个行业，不在只是军工方面的科技，而是深入大众的平民科技。

在中国的高校，对于图书的管理早已经采用计算机软件对图书做系统的管理，但是图书馆的图书系统还停留在以前的概念，没有随着需求的变化而做出自己的改变，用户参与度基本为零，在生活中，我们在遇到一个比较棘手的问题，需要到图书馆借阅相关书籍用来解决这个棘手问题，但是发现到了图书馆没有该方面的书籍，或者询问图书管理人员有该书籍但是已经被借阅完毕，到了最后，问题没有被解决反而还浪费了大量的宝贵时间，久而久之，愿意来图书馆借书的人越来越少了，图书馆存在的价值以及意义也就不复存在了，为何更好的发挥图书馆的作用，我们需要设计一个智慧型的图管理系统，让用户（学生或者其他爱好者）参与进来，进一步的焕发图书馆的活力，让图书馆存在的价值和意义得到充分的发挥，本文即是对该系统的需求与实现进行深一度的阐述，展示大体的功能与作用。

# 绪论

什么才能称之为智慧？了解用户的需求，适时作出改变称之为智慧，作为一个智慧型图书馆理平台，一定不能让用户在使用的过程中觉得繁琐，要在平台中充分的体现用户的重要性，以用户的切身体验出发，让用户快速的找到自己想要的书籍并借阅成功，经过二周左右的调查研究发现，要让用户能够快速的借阅到自己想要的书籍，不来会奔波，需要实现一个用户参与并且操作简易实用性强的图书管理平台。

## 1.1 课题内容

随着社会的发展，互联网的普及以及互联网技术的发展，生活中越来越多的地方开始使用互联网技术进行资源和人力的管理，互联网应用在带给人们带来巨大便利的同时，大大释放了劳动生产力。高校的图书管理系统每年都在升级，管理图书的效率也大大提升，但是，图书馆里系统只是针对管理员使用，对于学生而言，学校图书管新引进什么类型的书籍，书籍的借阅情况一无所知，随着图书数目和种类的增加，学生借阅到一本适合自己当前阶段的书的时间也就相应的得到提升。作为学生本身，并不希望自己大量的时间浪费去寻找一本适合目前提升自己的书籍上，如果图书馆不加以改变，将信息分享出去，学生去图书馆借阅书籍的热情也会褪去，图书馆的存在意义也就不那么明显，没有太多的价值意义。本系统旨在将图书馆的信息分享给学生，让他们能参与进来，方便学生的同时提升图书馆在大学中存在的价值及意义，让书籍为人类发挥最大的作用，提升学生及爱好学习人士的能力。从此方面来讲，研究此课题具有很大的意义。

## 1.2 课题背景

就目前来言，图书馆的系统在管理层面做的无可厚非，但对于信息分享方面就有所欠缺了。如今大学的校园越来越大，而高校的图书馆一般只有一个，学生距图书馆的距离一般而言都比较远，学生遇到或者解决一个比较难的问题时，需要到图书馆借阅一堆书籍，方便找到问题的解决思路，然而有的时候，等学生到了图书馆之后发现该类型的书已经被借阅完了，白跑一次，问题不仅没有丝毫的进展，反而还浪费了大量的时间和精力，下次再来借也不知道什么时候有书，总之十分不方便，假若，这个人在来的时候知道图书馆有这种类型的书，他就不会白跑一次，为了避免这种类似事情的发生，信息的共享便变得很有价值和意义，因此，做一个智慧型图书管理也变得无可厚非了。

## 1.3 课题意义

基于该课题实施前，阅览了网上大量的关于智慧型图书馆搭建的案例以及带来的便利和影响，对于身边的同学也采用问卷调查或者直接询问等方式对于提出的问题进行归纳整理，大多数问题暴露在借阅时间长，效率慢，图书库存少，不知道图书馆关于所需书籍的情况，本系统针对以上的问题在本系统中做出优化处理，尽量解决学生在借阅图书的过程中所遇到的问题，方便他们进行知识的学习和进步，为了更好的帮助学生，该系统还需要做到以下几点

1. 方便，快捷，简单易操作
2. 安全可靠，防数据丢失
3. 实时，可靠，参与度高

## 1.4 开发环境

本系统采用WIN10操作系统，要求电脑内存4G以上，代码编译器选择程序界比较流行的Visual Studio Code编译器，使用Node.JS作为后台程序开发语言，Node.JS是一个基于 [Chrome V8 引擎](https://developers.google.com/v8/)的 JavaScript ，Node.JS 采用高效、轻量级的事件驱动、非阻塞 I/O 模型,作为解释性后台编译语言，十分高效和便捷，对开发者十分友好，本次开发使用的是基于Node.JS官方推荐的WEB应用开发框架Express。数据库语言选择轻量级的非关系型数据库MongoDB，MongoDB是一个基于分布式文件存储的数据库。它是由[C++](http://baike.baidu.com/item/C++" \t "http://baike.baidu.com/_blank)语言编写。是一个采用BSON格式来存储数据的一个高性能数据库。前端界面选择目前比较流行的MVVM框架，VUE技术栈以及界面风格统一的可扩展型UI框架，Element UI来实现。

## 1.5 开发平台介绍

### 1.5.1 Visual Studio Code

Visual Studio代码是一个轻量级但强大的源代码编辑器，它运行在您的桌面上，可用于Windows、macOS和Linux。它内置对JavaScript、TypeScript和Node.JS的支持。拥有丰富的其他语言扩展生态系统(如c++、c#、Java、Python、PHP、Go)和运行时(如. net和Unity)。具有非常高的可扩展性，强大的包工具支持对开发人员编译软件带来巨大的便利，提升编译人员的开发效率。

### 1.5.2 Element UI

Element UI目标在于降低开发门槛、提高开发效率、减少代码量，在RIA崭露头角的时代此类特点显得尤为重要，同JQuery的推崇的Write less,Do more思想类似，在UI开发方面的提供UI基类、底层统一的API及风格统一的界面，让设计师也可构建成熟的用户界面系统、模块化的开发方式让开发者更专注于业务层的处理，提升开发效率及用户体验。

Element UI采用诸如、ExtJS、Dojo、JQuery现今流行框架的精髓（JQuery的简单易用、YUI3、Dojo等优秀的企业级应用架构管理、ExtJs的丰富组件及组件工作模式），尽量缩减体积并向标准靠拢，试图打造轻量级的Web前端及RIA应用框架，如其名，它拥有优雅易用的API、轻量级的身材、功能丰富的插件、简单易扩展的机制，旨在创建一个功能丰富但又轻量级同时又兼顾企业级应用开发的系统。

### 1.5.3 VUE技术栈

VUE是一套用于构建用户界面的渐进式框架。与其它大型框架不同的是，VUE被设计为可以自底向上逐层应用。VUE的核心库只关注视图层，不仅易于上手，还便于与第三方库或既有项目整合。另一方面，当与[现代化的工具链](https://cn.vuejs.org/v2/guide/single-file-components.html)以及各种[支持类库](https://github.com/vuejs/awesome-vue" \l "libraries--plugins" \t "https://cn.vuejs.org/v2/guide/_blank)结合使用时，VUE也完全能够为复杂的单页应用提供驱动。

### 1.5.4 Node.JS

Node.js是一个Javascript运行环境(runtime environment)，发布于2009年5月，由Ryan Dahl开发，实质是对Chrome V8引擎进行了封装。Node.js对一些特殊用例进行优化，提供替代的API，使得V8在非浏览器环境下运行得更好。

V8引擎执行Javascript的速度非常快，性能非常好。Node.js是一个基于Chrome JavaScript运行时建立的平台，用于方便地搭建响应速度快、易于扩展的网络应用。Node.js 使用[事件驱动](https://baike.baidu.com/item/%E4%BA%8B%E4%BB%B6%E9%A9%B1%E5%8A%A8" \t "https://baike.baidu.com/item/node.js/_blank)，非阻塞[I/O](https://baike.baidu.com/item/I/O/84718" \t "https://baike.baidu.com/item/node.js/_blank)模型而得以轻量和高效，非常适合在分布式设备上运行数据密集型的实时应用。

### 1.5.5 Express

Express 是一个基于 Node.JS 平台的极简、灵活的WEB应用开发框架，它提供一系列强大的特性，帮助你创建各种 Web 和移动设备应用。丰富的 HTTP 快捷方法和任意排列组合的 Connect 中间件，让你创建健壮、友好的 API 变得既快速又简单。Express 不对 Node.JS已有的特性进行二次抽象，我们只是在它之上扩展了WEB应用所需的基本功能。

### 1.5.6 MongoDB

MongoDB 是由C++语言编写的，是一个基于分布式文件存储的开源数据库系统。在高负载的情况下，添加更多的节点，可以保证服务器性能。MongoDB 旨在为WEB应用提供可扩展的高性能数据存储解决方案。MongoDB 将数据存储为一个文档，数据结构由键值(key=>value)对组成。MongoDB 文档类似于 JSON 对象。字段值可以包含其他文档，数组及文档数组。MongoDB 是一个面向文档存储的数据库，操作起来比较简单和容易。你可以在MongoDB记录中设置任何属性的索引 来实现更快的排序。你可以通过本地或者网络创建数据镜像，这使得MongoDB有更强的扩展性。

## 1.6 本章小结

本章主要详细的阐述了课题研究的背景及存在的意义，开发系统所使用到知识点以及所需知识点的简单介绍，为下一步的系统功能分析，详细设计，功能实施打下坚实的基础。

# 系统设计

在阐述了我研究该课题的背景以及含义后，接下来便是整个系统的大体设计，一个好的系统设计是非常有必要的，参考了大量同类型功能设计的文档和同事意见，在这个章节里我将我实现的系统功能及数据库设计呈现出来。

## 2.1 系统需求整体设计

详细的需求分析和开发设计，是开发过程中十分重要的一步，同时也是十分必要的一步，在需求分析的过程中你会发现实现该功能中可能会出现的问题，根据这个需求做出详细的设计，开发的代码，系统的架构整体可能也会同时发生更改，避免在后期实现的过程中出现一些不可期的事件和不必要时间消耗，譬如，需求实现起来困难度巨大，实现起来耗费时间特别长，需求十分鸡肋，很少有人去使用，所以详细的需求分析和整体设计还是十分必要的，经过我的调研以及参考大量相似系统后，第一期主要完成以下几个方面的需求。

1. 权限功能的划分，整个系统，分为三个等级，对应三中不同的用户集群，相应权限大小依次递减，最高权限为root权限，即超级管理员，超级管理员享有系统中的所有模块功能，其次是作为图书管理的工作人员，称之为admin权限，具有图书的录入功能，和查看所有的用户权限，最后则是普通的用户，为user权限，享有图书信息的查阅以及个人信息的管理。
2. 公告模块：对于图书馆新增加的书籍，或者一些其他重要的事情，需要一个途径第一时间广播到各级用户，公告模块也就由此产生，管理员可以对公告进行编辑以及修改，普通用户可以查看到图书馆发布的最新消息，使信息变得更加透明化。
3. 图书展示模块：这个模块主要展示图书馆内目前所有的图书信息，让用户能够查阅到图书馆内最新的图书信息，对喜欢的书籍进行借阅和电子书籍的下载，节省他们的借书的时间。
4. 图书录入模块：这个模块主要针对admin及root用户，该用户群体可以对图书馆内的图书信息进行录入，将图书馆内最新的图书信息传达给用户。

## 2.2 系统需求实现的可行性分析

学过软件开发流程的人都明白，系统需求实现的可行性分析这一步骤在需求确认后是十分必要的，通过对其分析得出系统大致开发生命周期以及部分功能是否可行，是否需要从新讨论，可行性分析对于软件开发十分重要，可以在前期避免一些因为错误估算给公司或者团队造成的不必要的损失和重大case,下面我将从技术可行性，时间可行性以及经济可行性需求做出整体客观的分析。

1. 技术可行性

智慧图书馆里平台是一个可以让用户参与其中的图书馆里系统，其中主要的难点就是对数据库的设计以及系统架构的整体设计，在技术选型上面，采用目前比较流行的一些渐进式，扩展程度高的框架进行开发，在一定程度上解决了整个系统架构设计经验不足这一问题，对于数据库，采用NOSQL的非关系型数据库MongoDB来作图书信息的存储，MongoDB相对MYSQL而言十分精简，在安全层面基本相同，对于表的设计可以通过JSON文件的形式来实现，相对来说创建表以及实施都比较简单，剩下的基本就是对数据库中信息的展示以及权限集合的拆分，从以上几点来，看在技术层面该系统实现起来完全可行。

1. 时间可行性

阻碍程序开发不仅有技术方面还有时间方面，对于每个人来言，时间都是极其珍贵重要的，不可过度浪费的，从三月五号到四月八号，是我们的正常实习时间，一个多月左右，在这段时间里，我有充分的时间可以利用到该程序的开发实现上面，所以从时间角度考虑系统实现的可能性是完全没有任何问题的。

1. 经济可行型

在整个软件开发的过程中，支持软件开发整个流程走下去的基础就是经济基础，还好，这个系统并不是特别复杂，开发又是我自己一人开发，不需要支付费用给第三方人员，对于数据库以及服务器都是使用的本地数据库以及服务器，相应的来说，效果也会大打折扣，但是对于目前的程序而言也没有任何的问题，整个开发环境只需要一台4G运行内存以上的电脑即可满足，所以，从经济可行性这个角度考虑系统实现的可行性是完全没有任何问题的。

## 2.3 程序功能实现具体设计

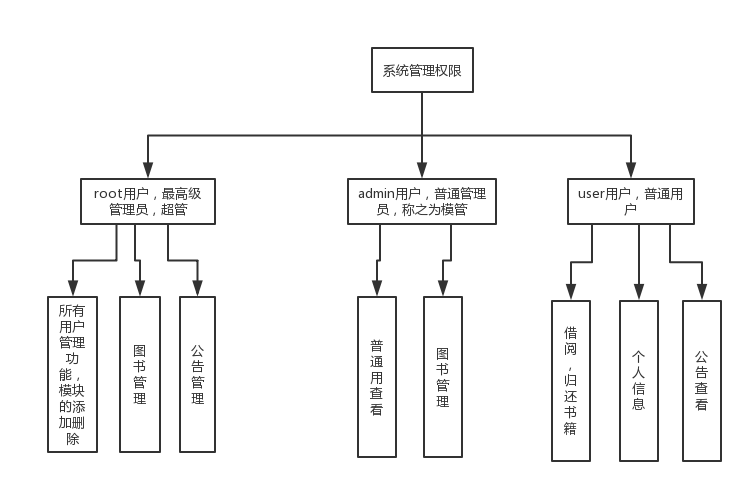


图2-1 系统管理基本权限

整个系统主要基本管理权限如图2-1所示

1. 超级用户：简称之为超管，拥有该系统所有的管理功能，能够指定模管，删除模管及普通用户，对于图书馆内的书籍进行添加，编辑以及删除，发布公告，修改及删除公告。
2. 模块管理员：简称之为模管，权限比超管权限低，能够查阅该系统中的所有注册用户，但是不能对其设置管理员以及删除，拥有与超管一样的图书馆里功能。
3. 普通用户：可以借阅归还系统中的书籍，查看书籍的详细信息，修改账户密码，查看系统中最新的公告信息。

## 2.4 数据库设计

作为一个系统，数据库尤为重要，是一个企业的命脉，数据库丢失或者被盗将会导致非常严重的灾难，很可能几十年的辛苦付之东流，毁之一旦，根据前人的教训以及忠告，在进行系统实现前对数据库设计显得尤为重要，根据第一章介绍，本系统选用MongoDB作为系统的数据库，通过JS方法类将JSON数据转为Schema形式的数据表结构，表名及参数如下

1. Users（用户基本信息）表

该表包含用户的个人基本信息，图书借阅信息，以及角色权限

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 数据结构 | 默认值 | 是否为主键 | 含义 |
| \_id | ObjectId | 无 | 是 | 用户ID |
| username | String | “” | 否 | 用户名 |
| password | String | “” | 否 | 密码 |
| isAdmin | Number | 0 | 否 | 是否为管理员 |
| bookIds | Array | [] | 否 | 借阅图书ID |
| booknames | Array | [] | 否 | 借阅图书名称 |
| encryted | String | “” | 否 | 密保 |

表2-1

在该表中，对于隐私性的密码信息采用MD5不可逆加密，即使用户的账号信息遭到泄露，只要对方不知道加密的密钥，就无法获得用户的其他信息，进一步保证用户在使用该系统个人信息的安全性。

1. book-list（图书基本信息）表

该表包含了图书馆里的基本信息，是整个系统中最重要的一张数据表，该表关乎整个系统的图书录入功能，其基本字段名称含义如下，如表2-2所示

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 数据结构 | 默认值 | 是否为主键 | 含义 |
| \_id | ObjectId | 无 | 是 | 图书ID |
| bookname | String | “” | 否 | 图书名称 |
| bookAutor | String | “” | 否 | 作者 |
| autorCountry | String | “” | 否 | 国籍 |
| bookPrice | String | “” | 否 | 价格 |
| bookPress | String | “” | 否 | 出版社 |
| bookUrl | String | “” | 否 | 图书封面地址 |
| isleave | Number | 0 | 否 | 借阅情况 |
| bookInstroduce | String | “” | 否 | 图书简介 |
| bookCount | Number | 0 | 否 | 图书数量 |
| bookCategory | String | “” | 否 | 图书类别 |
| EbooksUrl | String | “” | 否 | 电子版地址 |

表2-2

1. Notice(公告信息)表

该表以\_id为作主键，存储用户发布公告的所有基本信息，其结构内容如下表2-3所示

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 数据结构 | 默认值 | 是否为主键 | 含义 |
| \_id | ObjectId | 无 | 是 | 用户ID |
| username | String | “” | 否 | 用户名 |
| title | String | “” | 否 | 公告标题 |
| noticeValue | String | “” | 否 | 公告内容 |

表2-3

目前者三张表基本可以完成调研时所列出的所有功能，以后如果有新的需求加入，可以在JSON文件中加入对应的JSON数据，通过这种方式创建表结构，十分方便有效。

## 2.5 用例图

超级管理员用例图如图2-4所示，该用户具有以下操作能力

1. 图书搜索，筛选，公告查阅，图书借阅归还，个人密码修改
2. 设置模块管理员，取消模块管理员权限，删除用户
3. 图书的录入，编辑以及删除
4. 公告的添加，编辑，删除

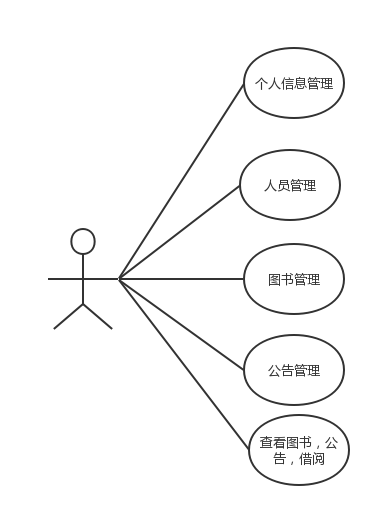


图2-4 超级管理员用例图

模块管理员用例图如图2-5所示，该用户具有以下操作能力

1. 图书搜索，筛选，公告查阅，图书借阅归还，个人密码修改
2. 系统内用户查看
3. 图书的录入，编辑以及删除

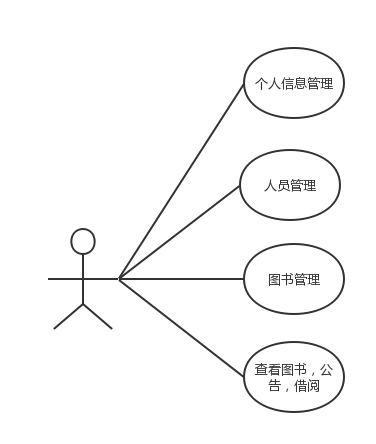


图2-5 模块管理员用例图

普通用户的用例图如图2-6所示，该用户具有以下操作能力

1. 图书搜索，筛选，公告查阅，图书借阅归还，个人密码修改

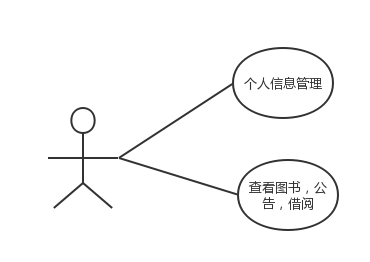


图2-6 普通用户用例图

## 2.6 E-R模型图

根据以上分析设计，系统整体E-R图2-7如下所示：

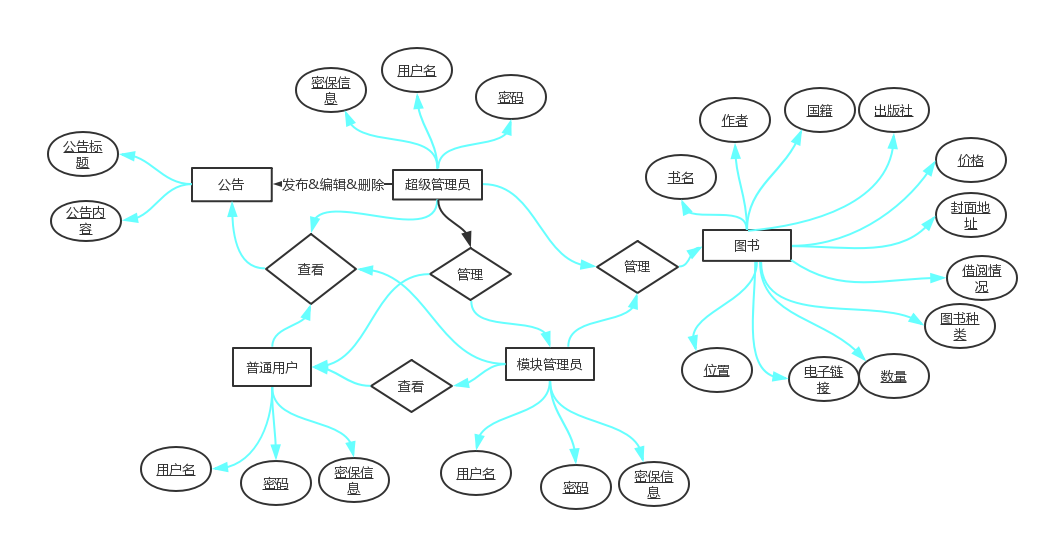


图2-7 系统整体E-R图

## 2.7 系统流程图

用户在登陆注册后，根据用户的权限不同，展示不同的操作能力，相应的模块根据权限进行动态增加或者隐藏，只有权限足够才可以看到对应的界面，权限不足只能看到当前权限下的功能和模块，该系统的流程图2-8如图所示

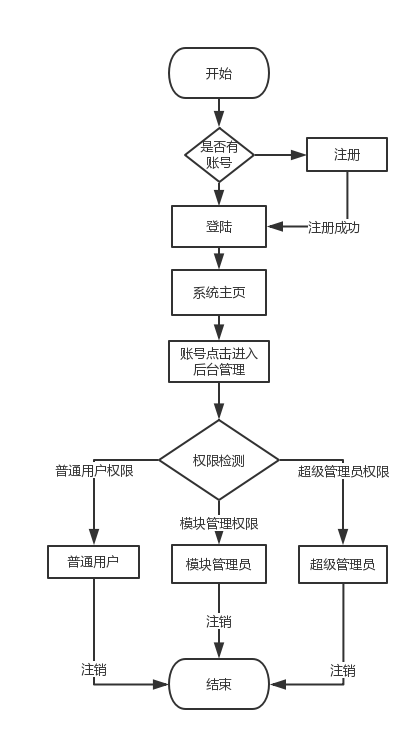


图2-8 系统流程图

## 2.8 本章小结

本章主要在第一章的基础上进行后续的延伸，主要是关于系统整体开发前的设计以及整个系统流程图的绘制，详细的向大家阐明了我在接到这个需求后的整体实现思路，为后续的功能开发奠定了一定的基础，下一章我将向大家整体介绍系统功能的代码结构实现。

# 系统功能具体实现

本系统在实现层面上采用前后端分离技术，前端主要负责系统的展示，后端负责数据的存储以及提供API接口，后台管理主要分为四个模块，分别为人员信息，用户管理，图书管理以及公告管理，本章将逐重向大家介绍各个模块的功能实现以及主要代码并结合效果图，实现之前分析的需求。

## 3.1登陆注册功能

登陆注册功是每一个网站必不可少的功能，是新用户注册，老用户登陆，进行系统操作的主要途径，用户在进行注册后可以通过登陆的方式进入主系统，进行进一步的系统操作，实现的效果图如图3-1，3-2所示

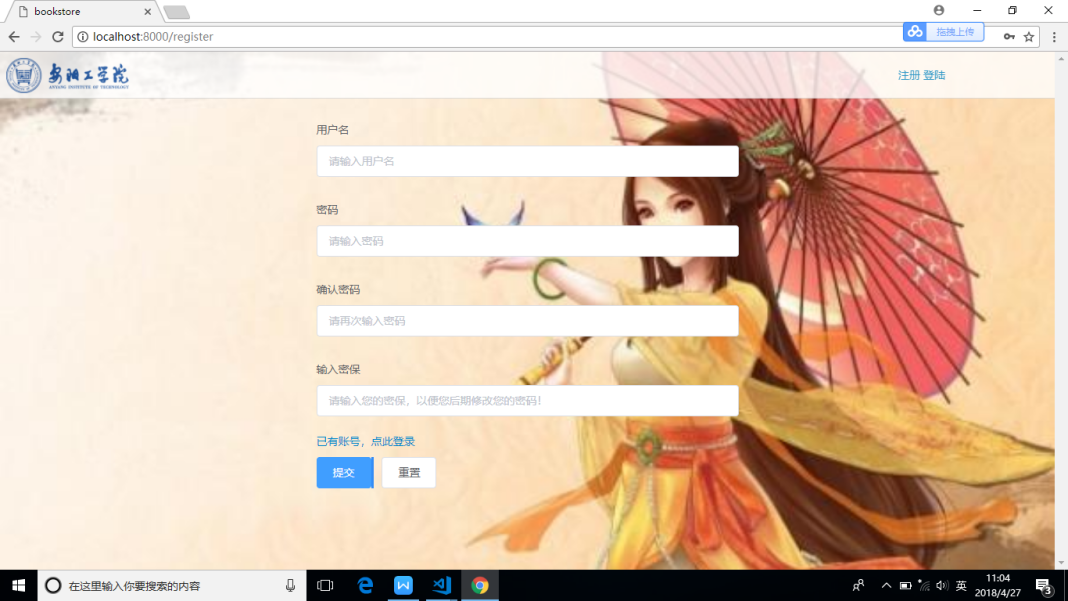


图3-1注册页面

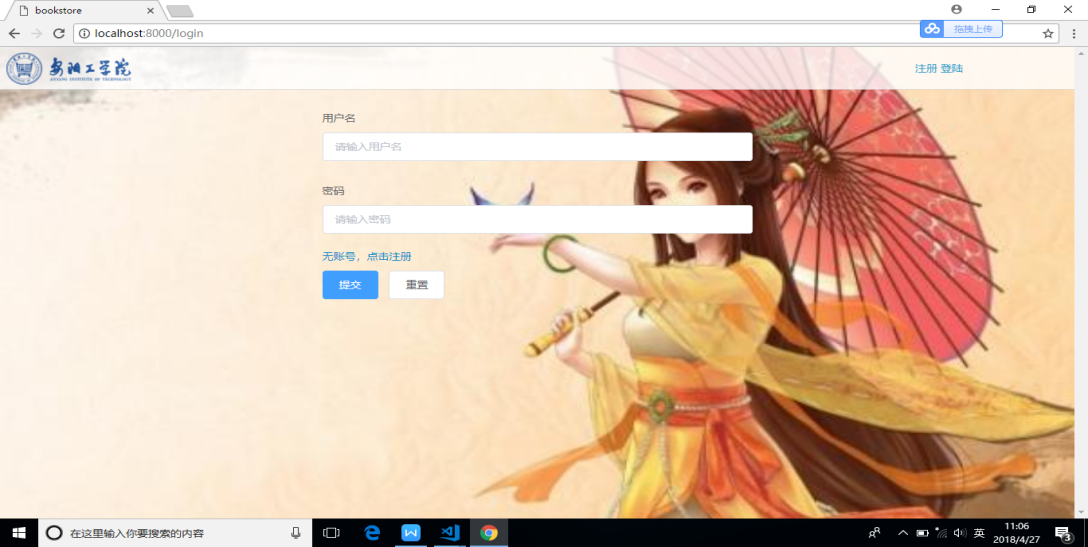


图3-2登陆页面

## 3.2超级管理员功能模块

对于超级用户即超管，通过主页的后台管理进入到管理页，可以查看个人的图书借阅情况，以及密码修改情况，设置管理员以及用户的删除，图书管理以及公告管理功能，如图3-3所示，具体细节功能如下：

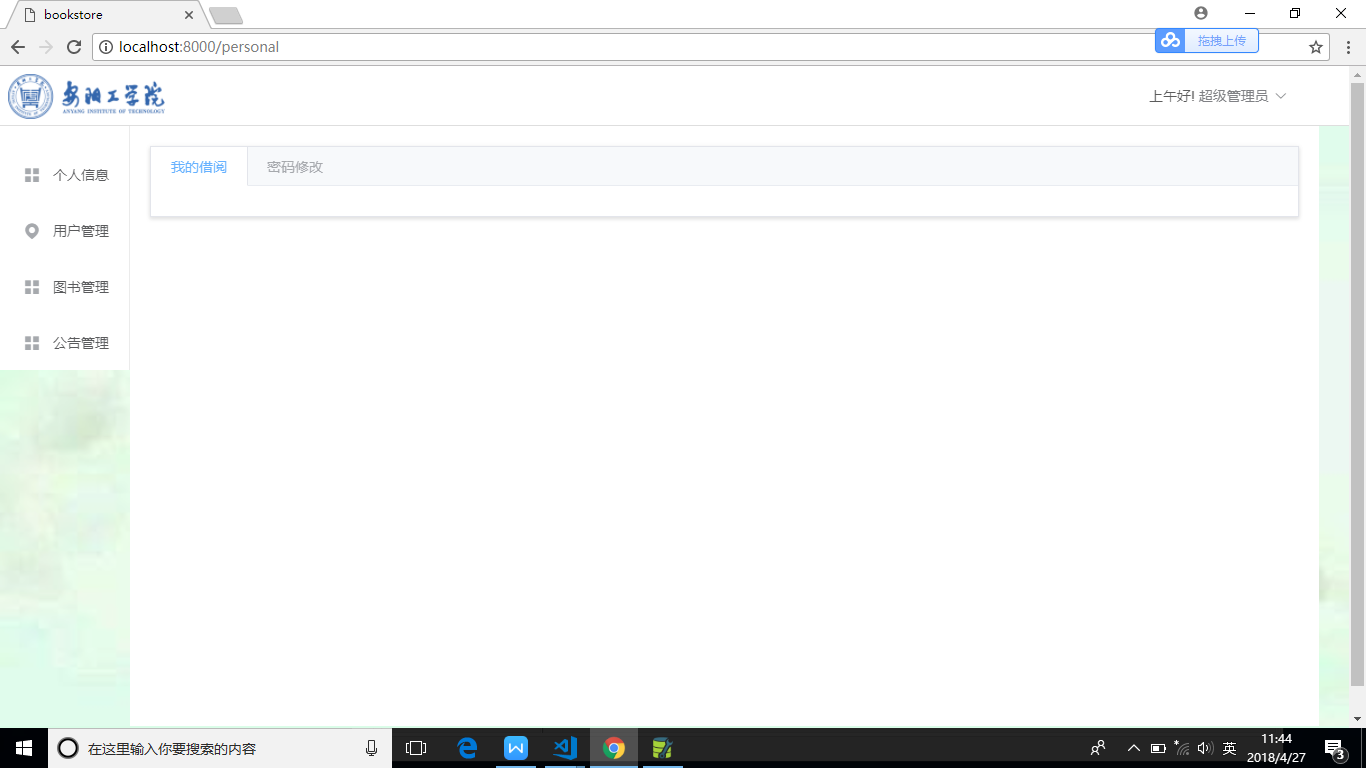


图3-2超级管理员界面

### 3.2.1 个人信息查看模块

超级管理员也是系统的组成成分之一，在整个系统功能面前，超级管理员也是一个用户，也享有查公告，借阅图书的功能。

### 3.2.2 用户管理模块

用户管理模块是超级管理员的重要模块之一，这个界面模块也可以访问，但是模管只有产看系统中所有的用户权限，没有用户的操作权限，所以在进入时一定注意登陆账号的权限，只有超级管理员才可以对用进行模块管理权限设计以及用户的删除，具体界面如下图3-3所示：

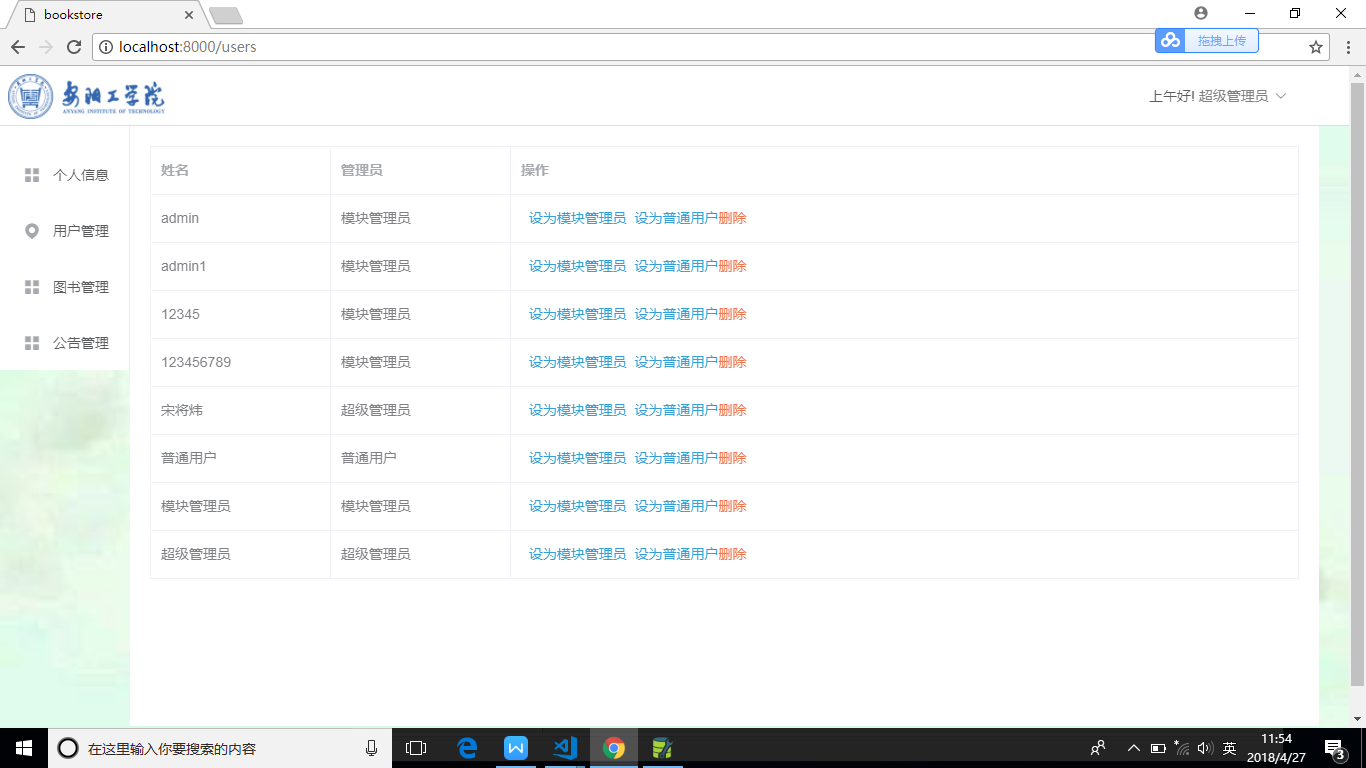


图3-3 超级管理员用户管理界面

如果想要将一个模块管理员设置为普通用户，只需点击设置普通用户即可，操作成功出现操作成功提示，如下图3-4所示，将用户名为admin的模块管理员设置为普通用户。

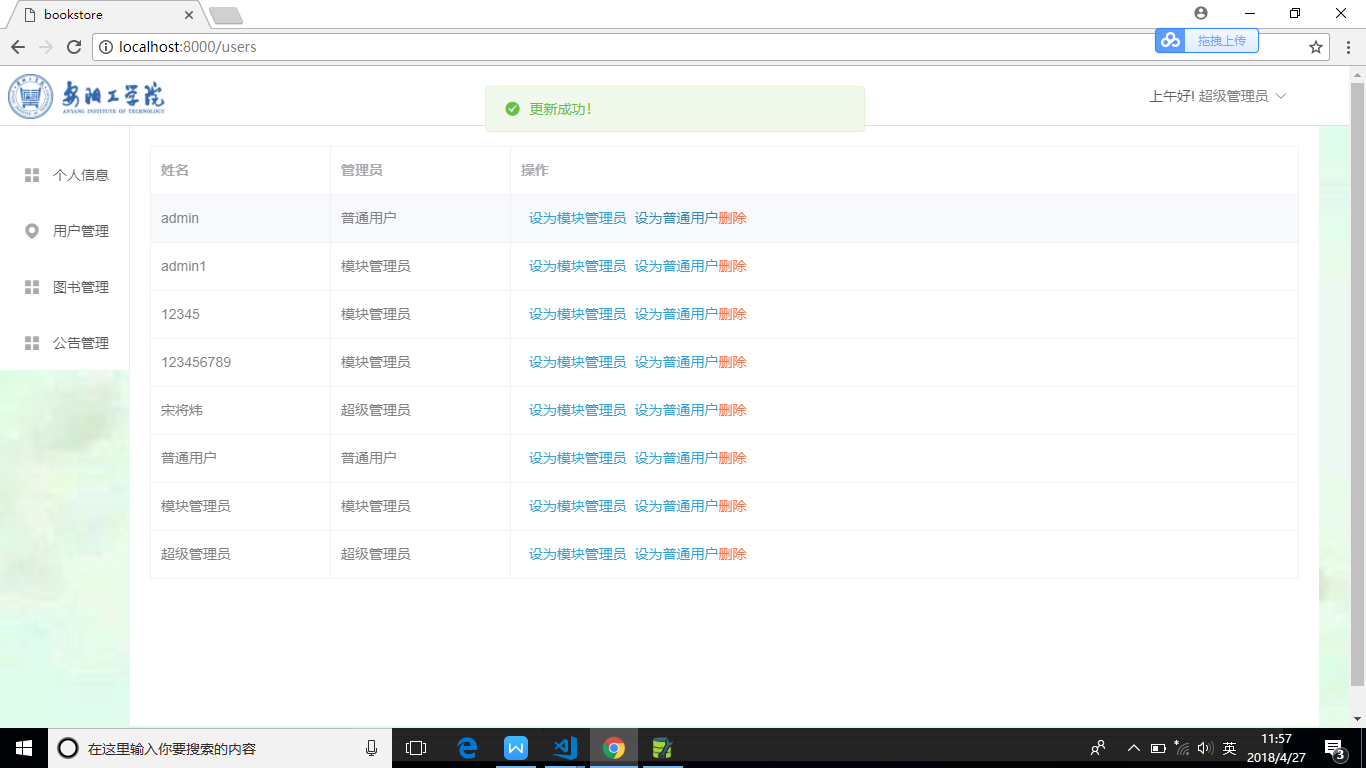


图3-4 超级管理员用户更新模块管理员

如果想将一个普通用户升级为模块管理员，只需点击设为模块管理员即可，如下图3-5所示，将之前设置为普通用户的admin用户从新设为模块管理员，其结果如下图所示。

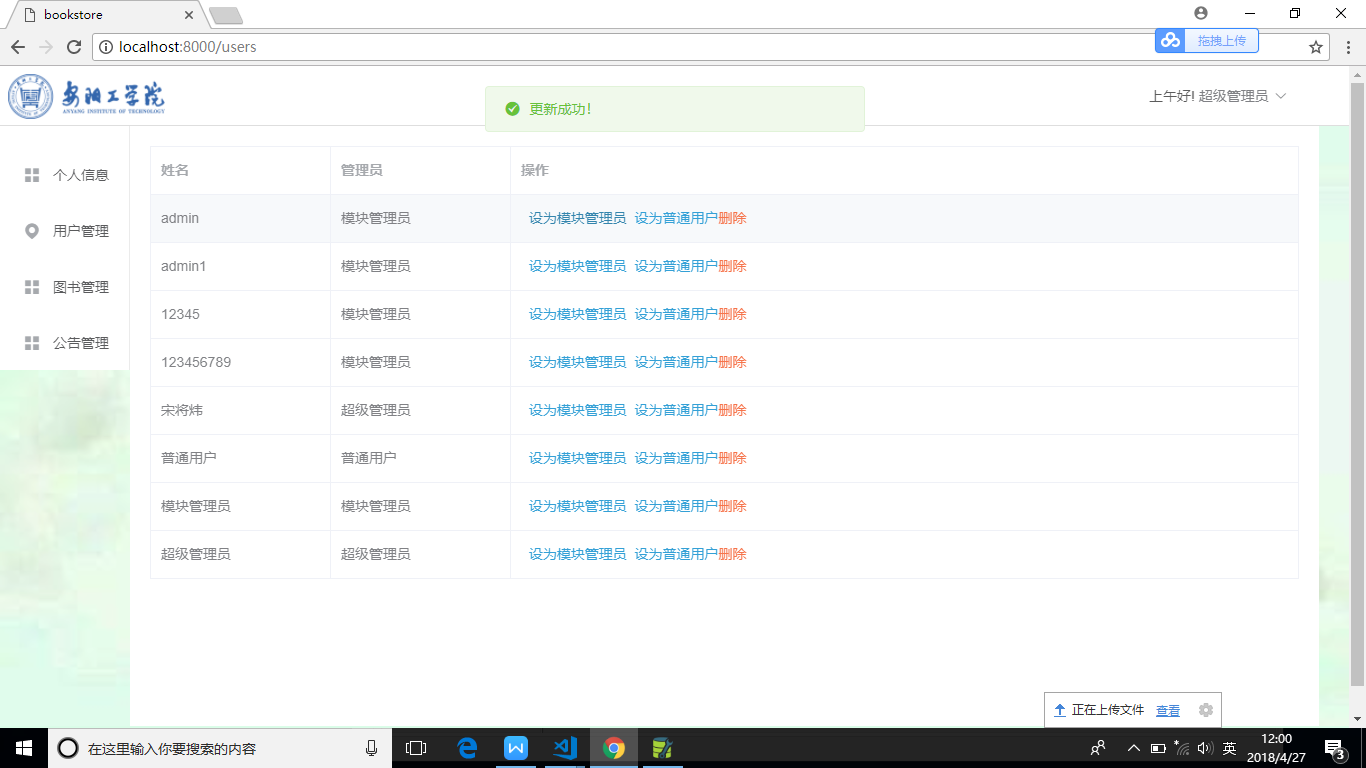


图3-5 超级管理员用户设置普通用户

如果想对一个用户进行删除操作的话，只需点击删除按钮即可，如下图3-6所示，将用户名admin的用户从系统中删除出去，用户列表中admin消失。

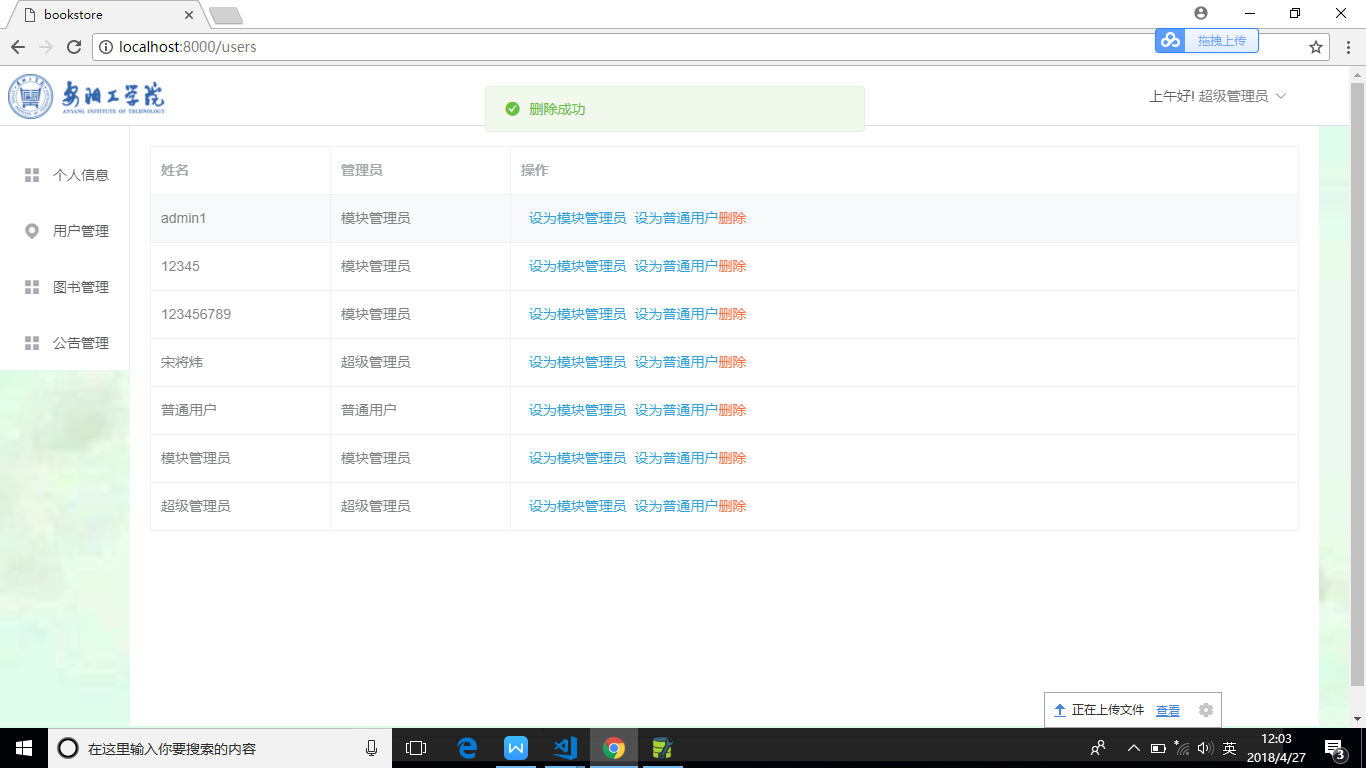


图3-6 超级管理员用户管理删除用户

### 3.2.3 图书管理模块

进入图书管理模块，展示当前图书馆下所有藏书，可以进行图书名称的模糊查询，以便快速的定位到图书，进行编辑与删除，可以选择图书的种类进行筛选，选择图书馆下某一类别的书籍，两者在同时执行时取结果的并集，基本结构如下图3-7所示

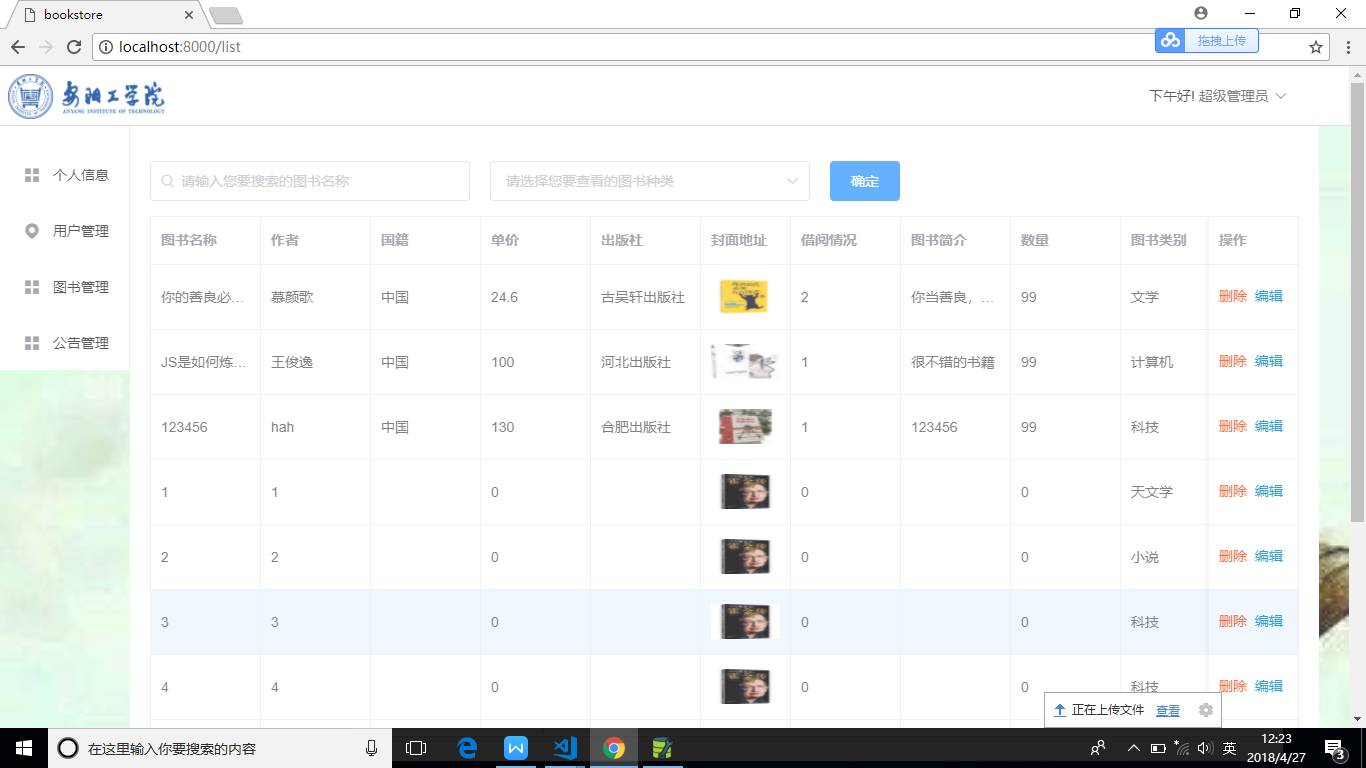


图3-7 超级管理员查看图书

当你想查看图书馆内的某一本图书时，在搜索框内输入你想要查询的图书名称，支持模糊查询，点击确定，显示你想要的图书信息，整体结果如下3-8，3-9所示

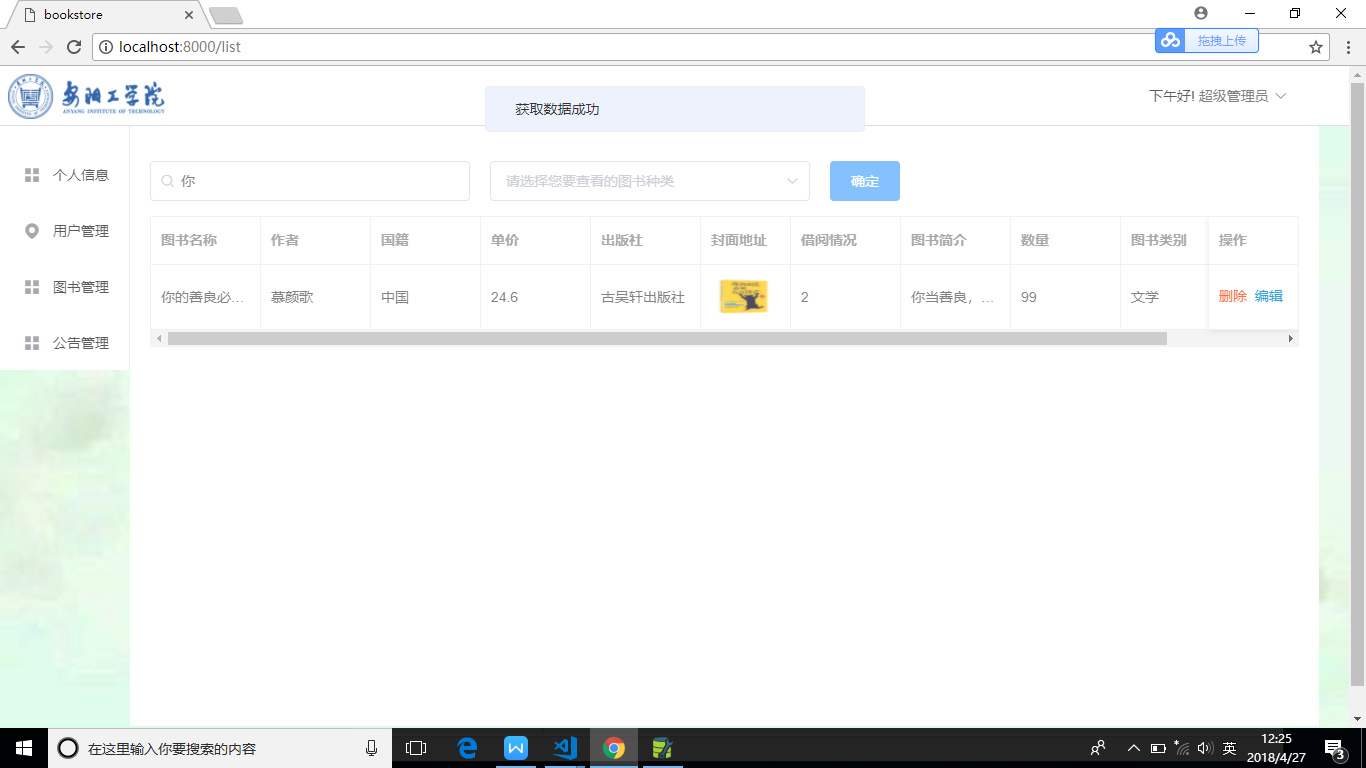


图3-8 图书搜索

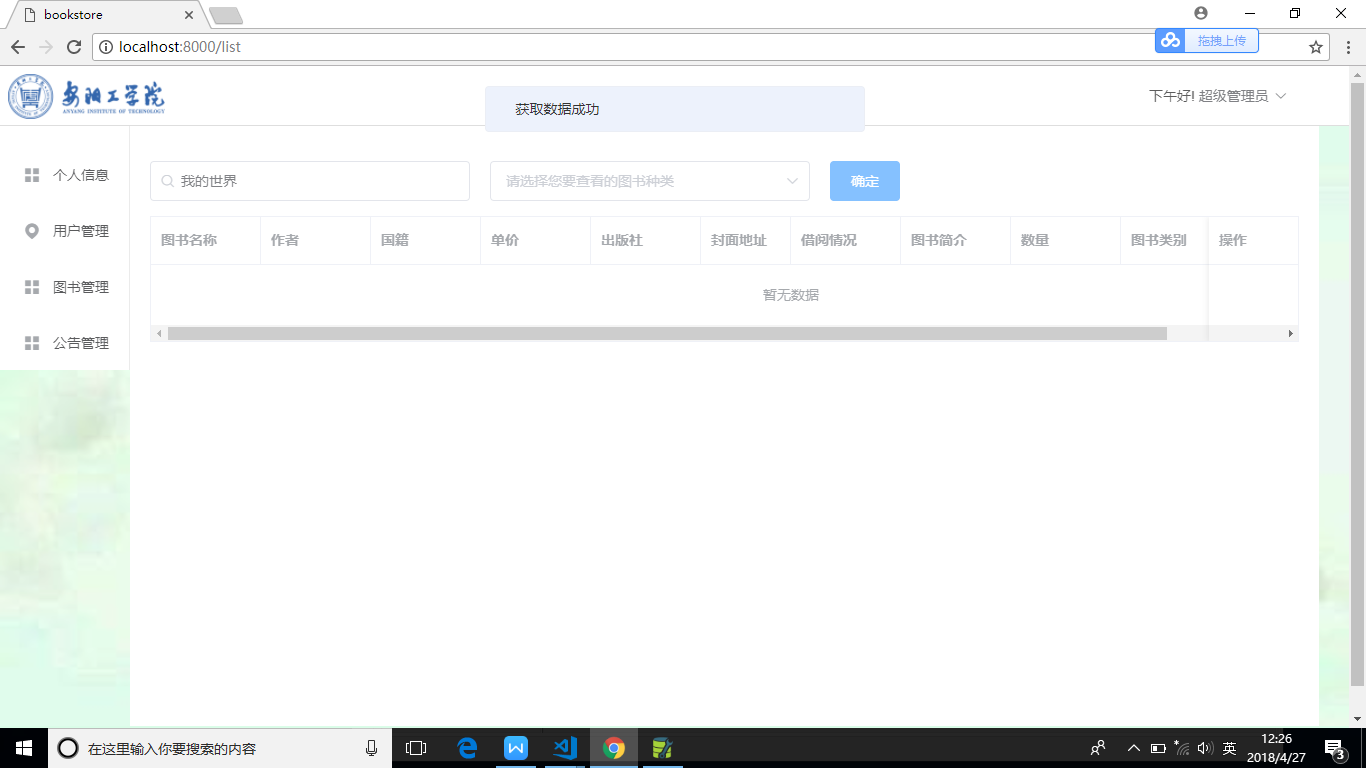


图3-9 图书搜索结果展示

当你想查阅图书馆内某一类别图书的时候，可以在下拉选择框里选择你想要查找的书籍种类，点击确定，显示你想要的图书信息，整体结果如图3-10所示

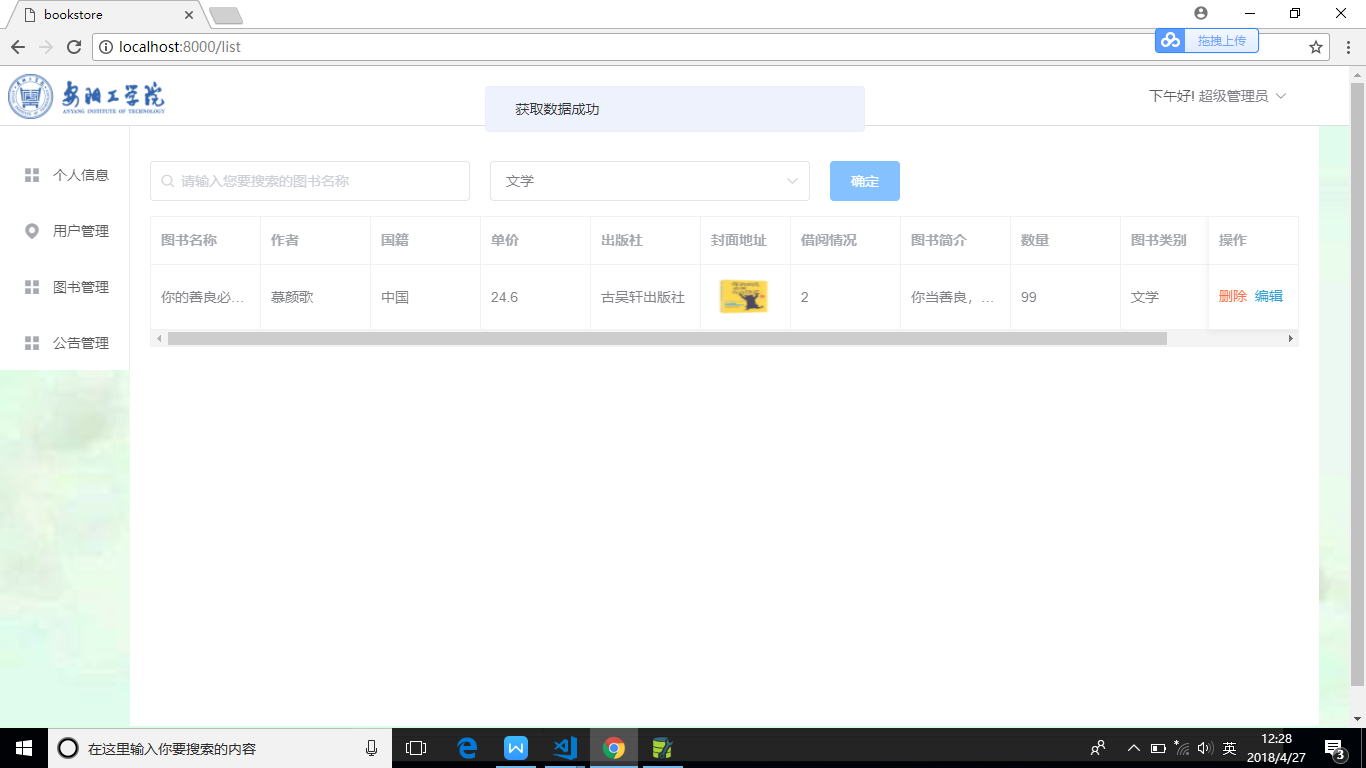


图3-10 图书类别筛选及结果

当你选中的图书类别没有相应的图书信息的时候，如下图3-11所示

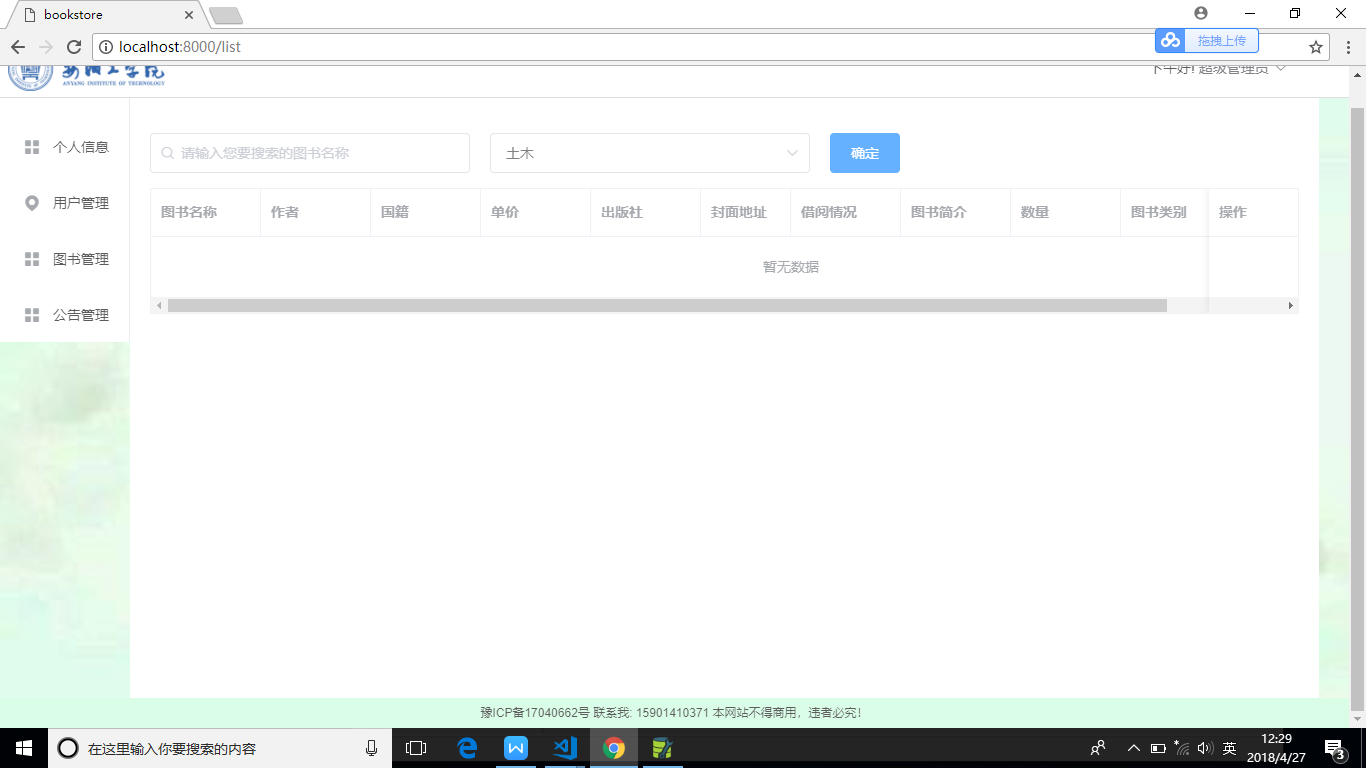


图3-11 图书类别筛选结果

当你想要在某一个类别下搜索固定的书籍，可以在搜索框内以及下拉框中输入你想要查询的数据，比如你想在文学下面查询“你的善良必须有点锋芒”这本图书，可以在搜索框内写入图书名称，下拉框选择文学类别，点击确定，搜索结果如图所示，未查到相关数据如图3-12所示

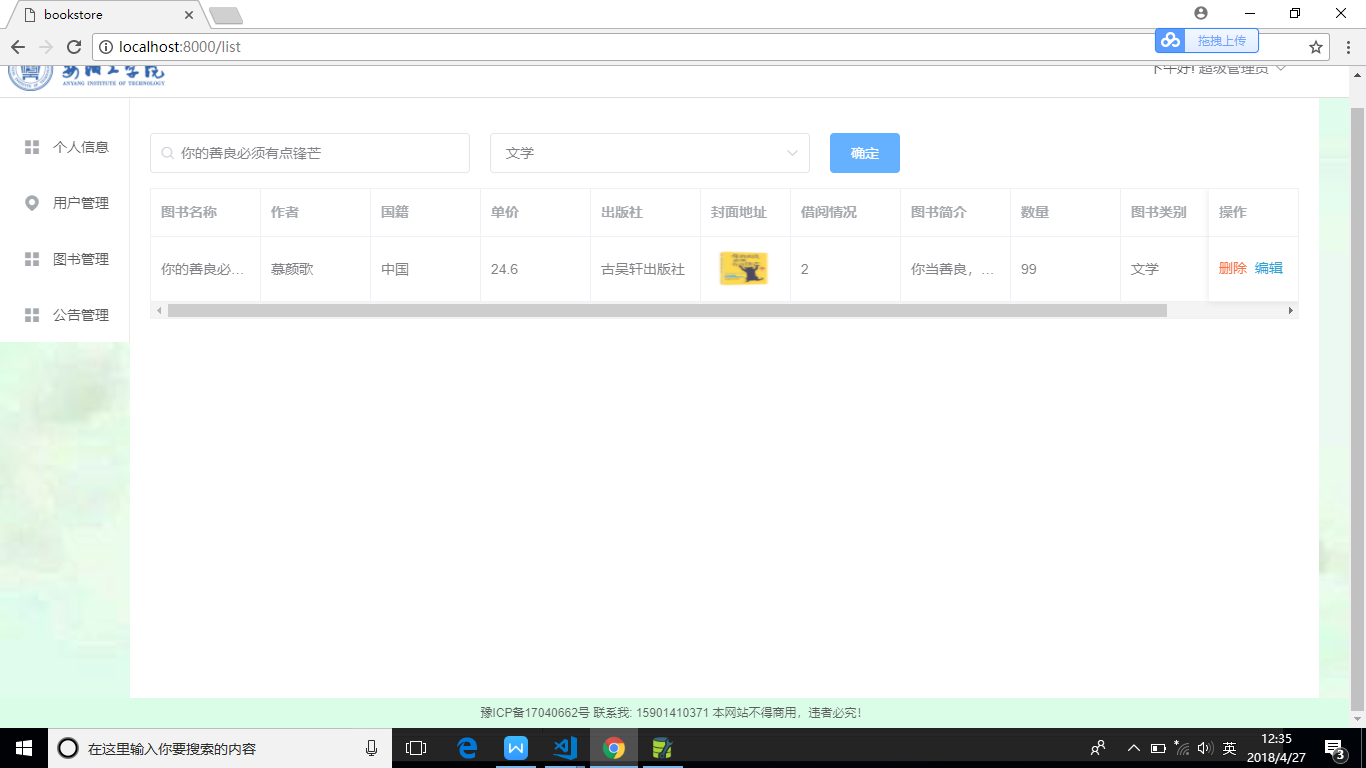


图3-12 图书类别筛选以及搜索结果

当图书馆引进一批新书的时候，可以将这些图书信息录入到数据库中，点击图书管理下的录入，进入图书录入界面，如下图3-13所示：

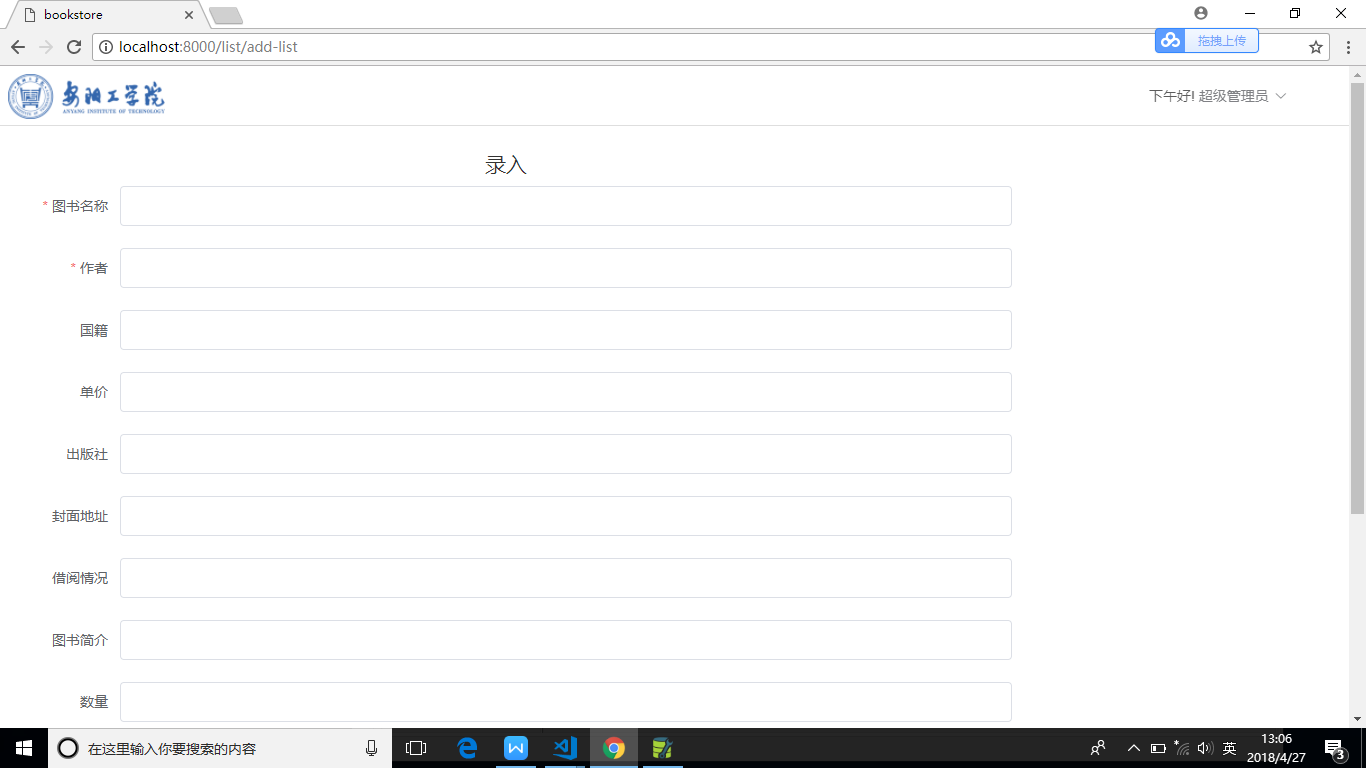


图3-13 图书信息的录入

按照提示要求录入图书信息，点击提交，保存成功，跳到首页，图书信息录入完成。

当你想要对图书进行编辑和删除操作时，点击图书对应的编辑和删除按钮，如图3-14所示

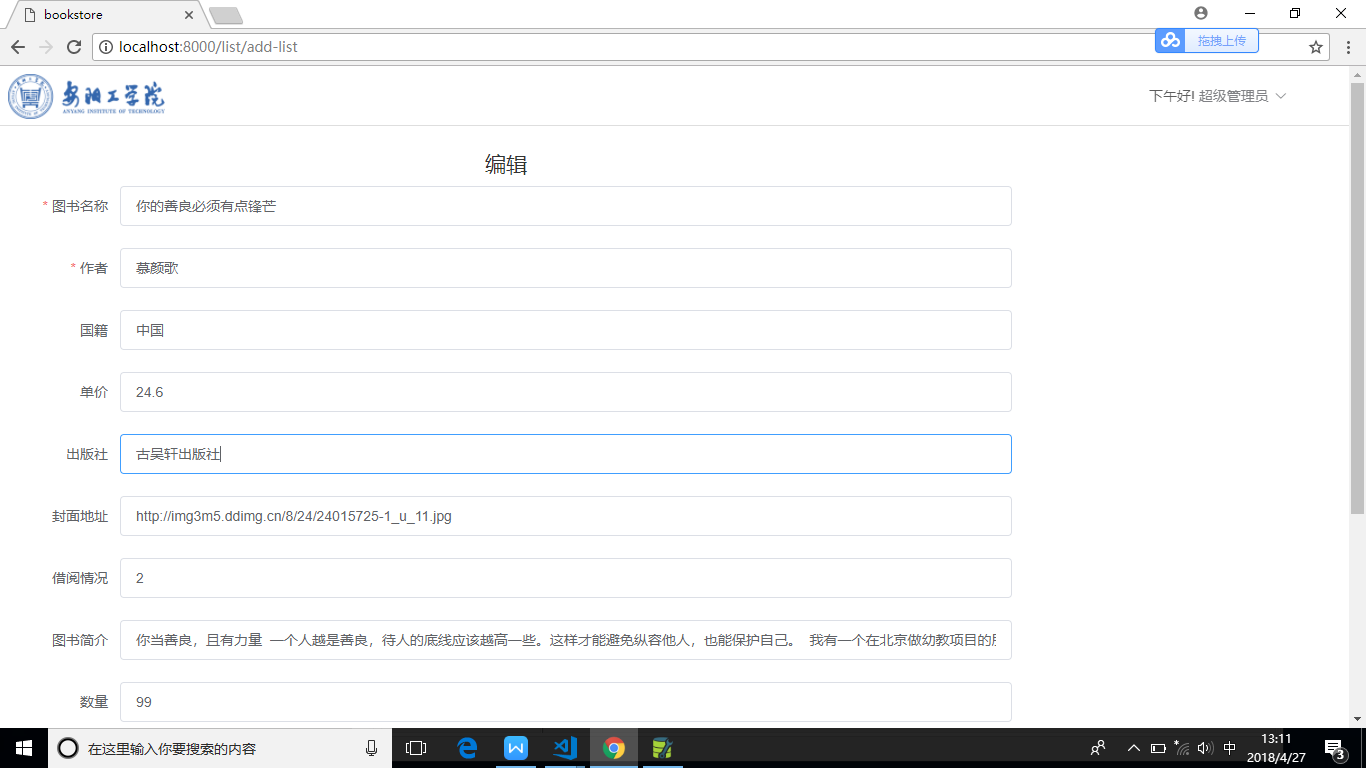


图3-14 已有的图书信息编辑

点击删除“js是如何炼成的”如图3-15所示：

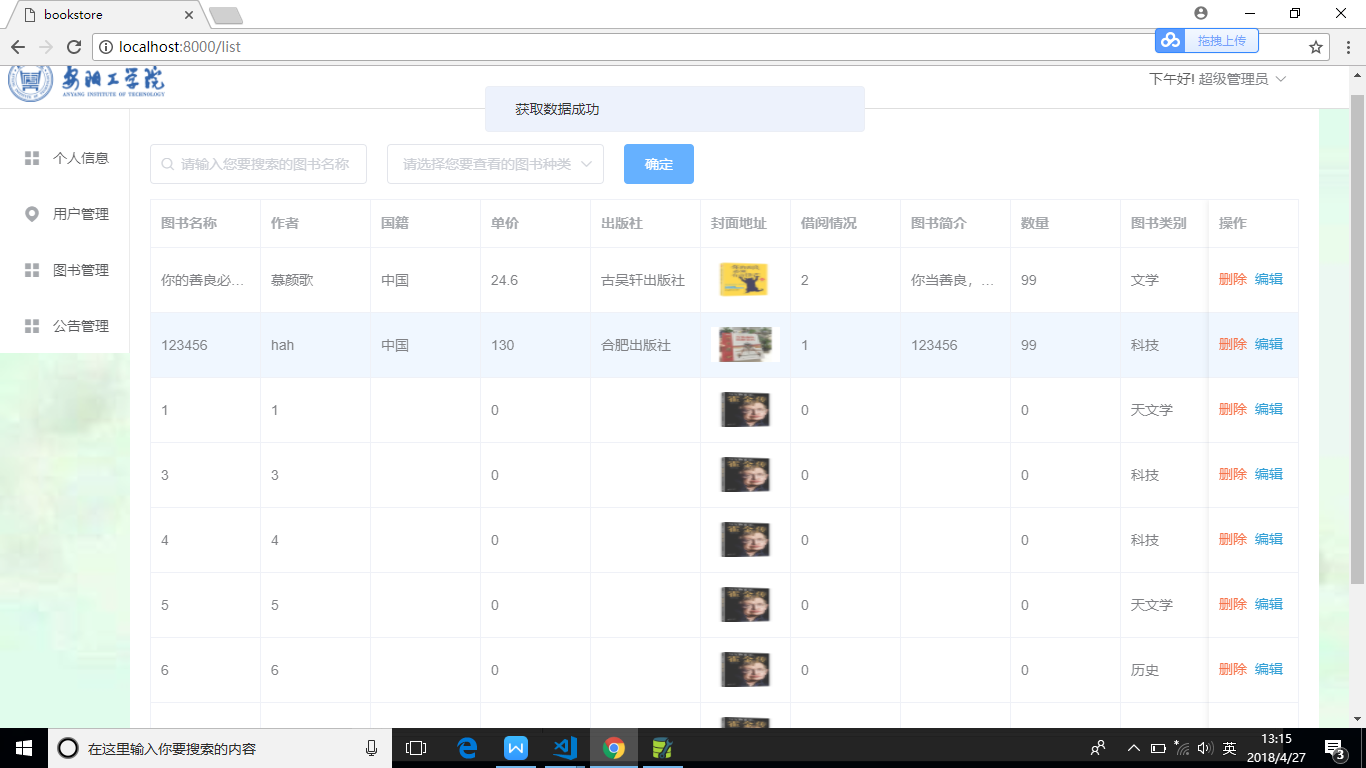


图3-15 已有的图书删除

### 3.2.4 公告管理模块

在该模块中，超级管理员可以发布最新的资讯，所有用户可以在首页查看到这些公告信息，进入公告首页，列出当前系统中存在的所有公告列表，如下图3-16所示

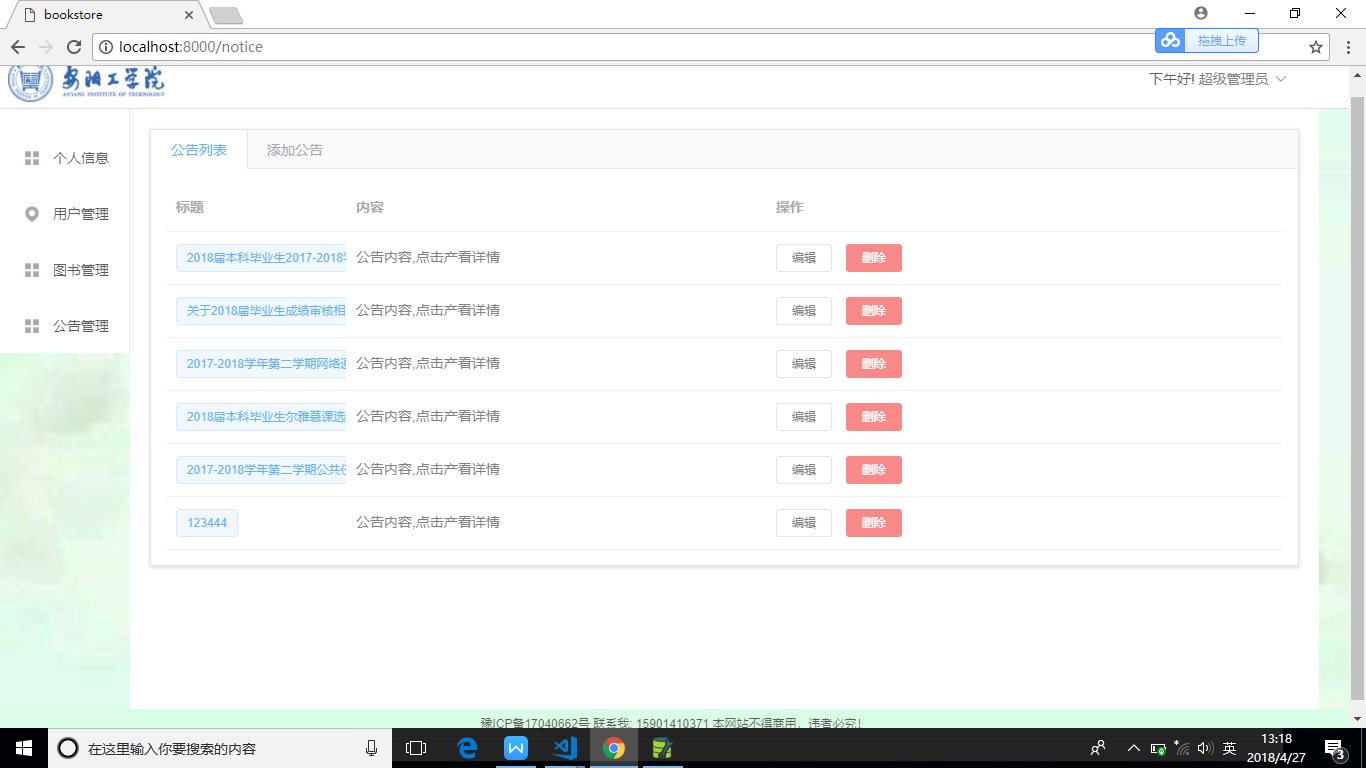


图3-16 超级管理员公告查阅

按照提示操作，可以查看对应公告编辑的详情。

当你想新增一条公告信息时，点击添加公告，进入添加公告页面，如图3-16所示

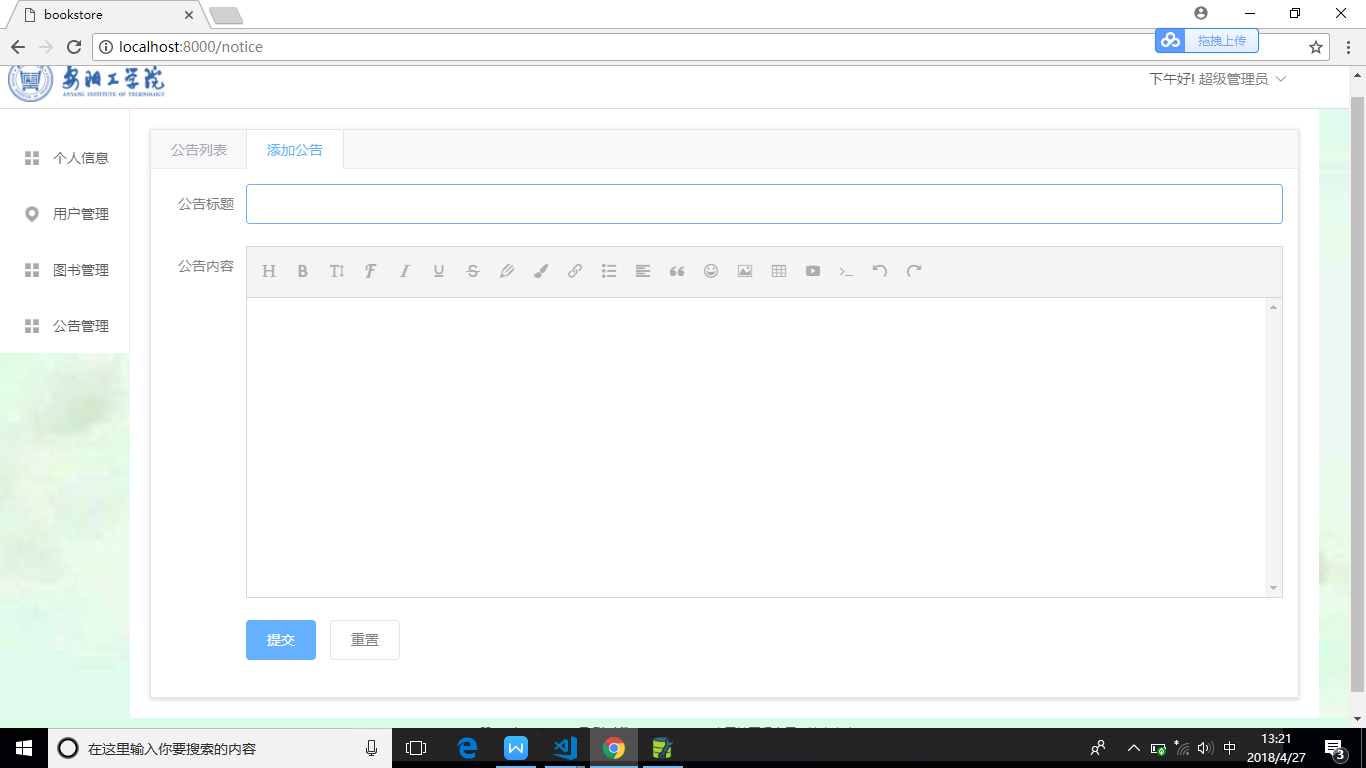


图3-17 公告的添加

输入您要填写的内容，点击提交，你便保存一条新的公告信息，点击公告列表，出现一天新的公告信息，如图3-18所示。

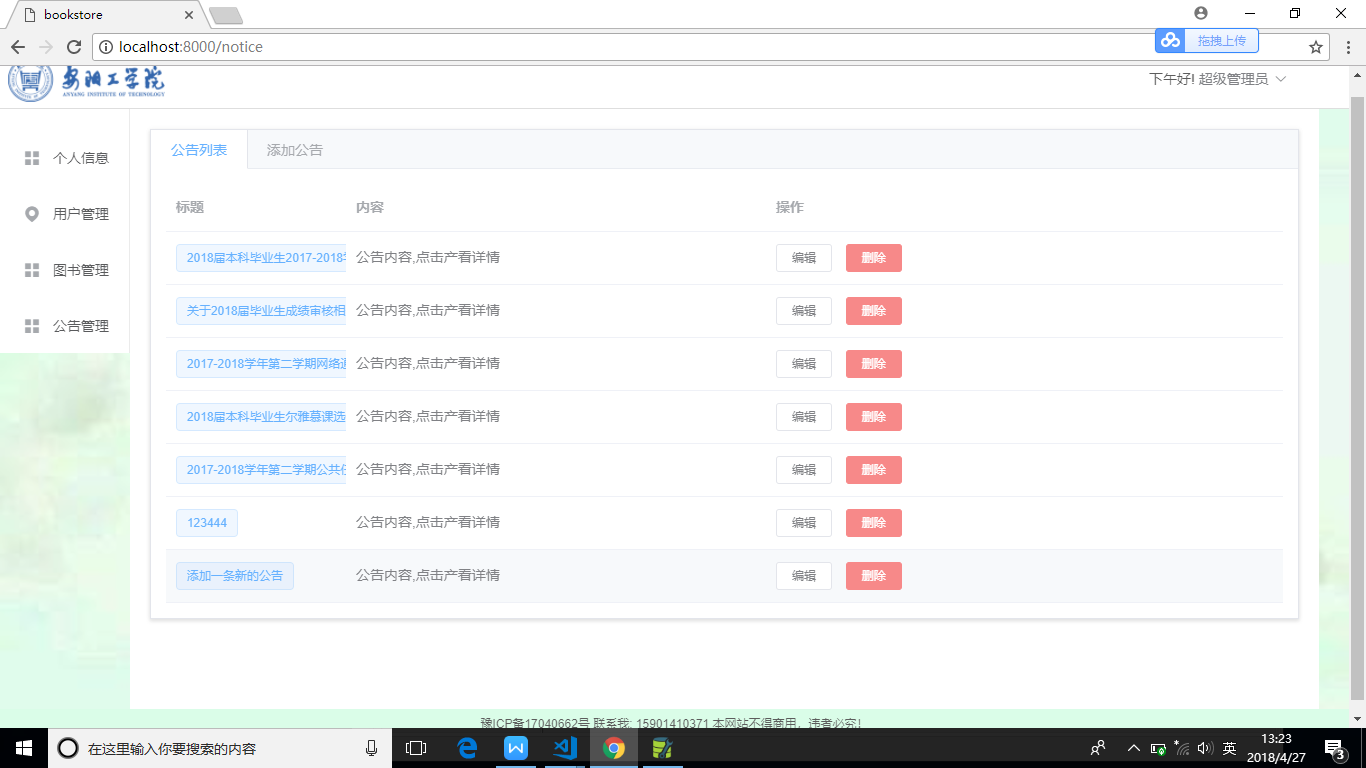


图3-18 公告的添加

进入首页，如下图所示可以看到刚才发布的公告信息，如图3-18所示



图3-18 公告的信息的查阅

## 3.3模块管理员模块

对于超级用户即超管，通过主页的后台管理进入到管理页，可以查看个人的图书借阅情况，以及密码修改情况，设置管理员以及用户的删除，图书管理以及公告管理功能，如图3-19所示

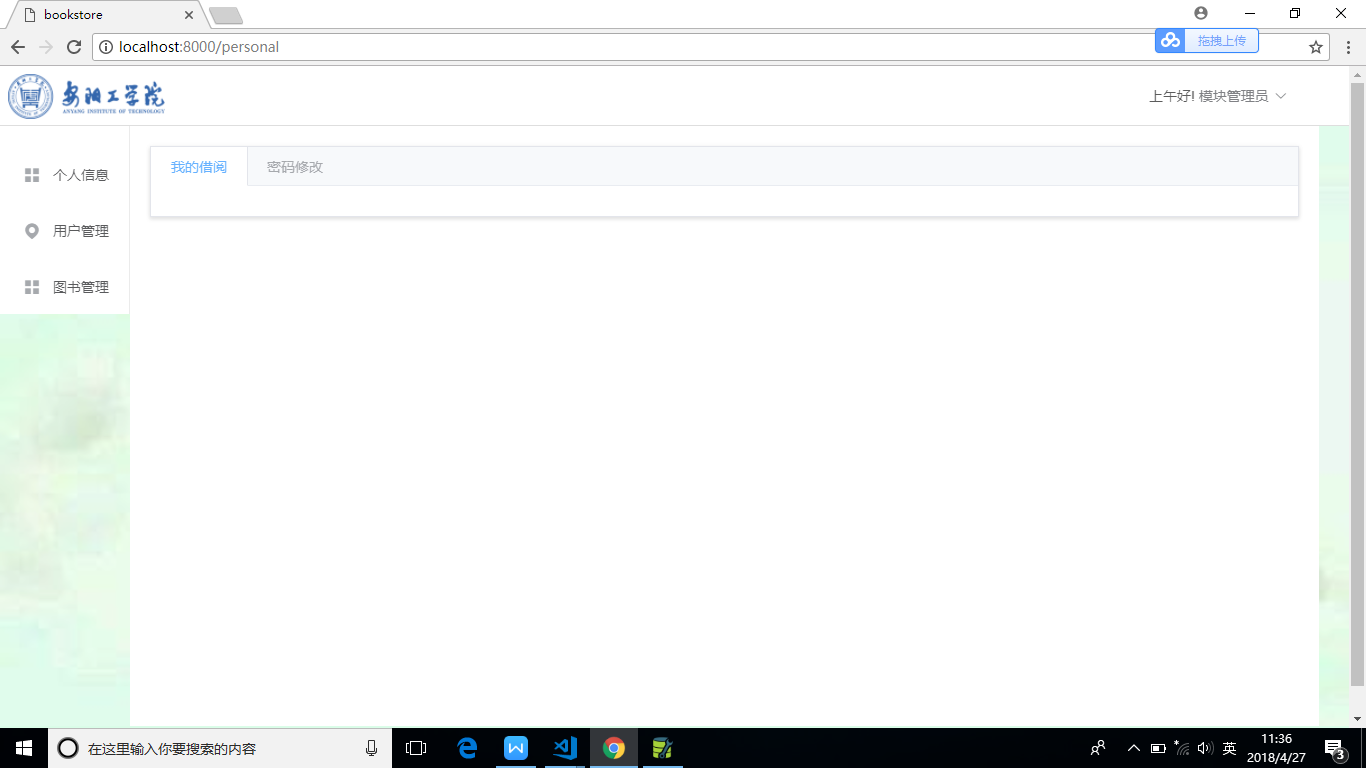


图3-19 模块管理员后台管理页面

模块管理员操作能力与超管的操作能力是一致的，所有效果和上面超级管理员处提到的一致，不同的是模块管理员只能查看人员，不能对人员进行任何操作，具体如下图3-20所示

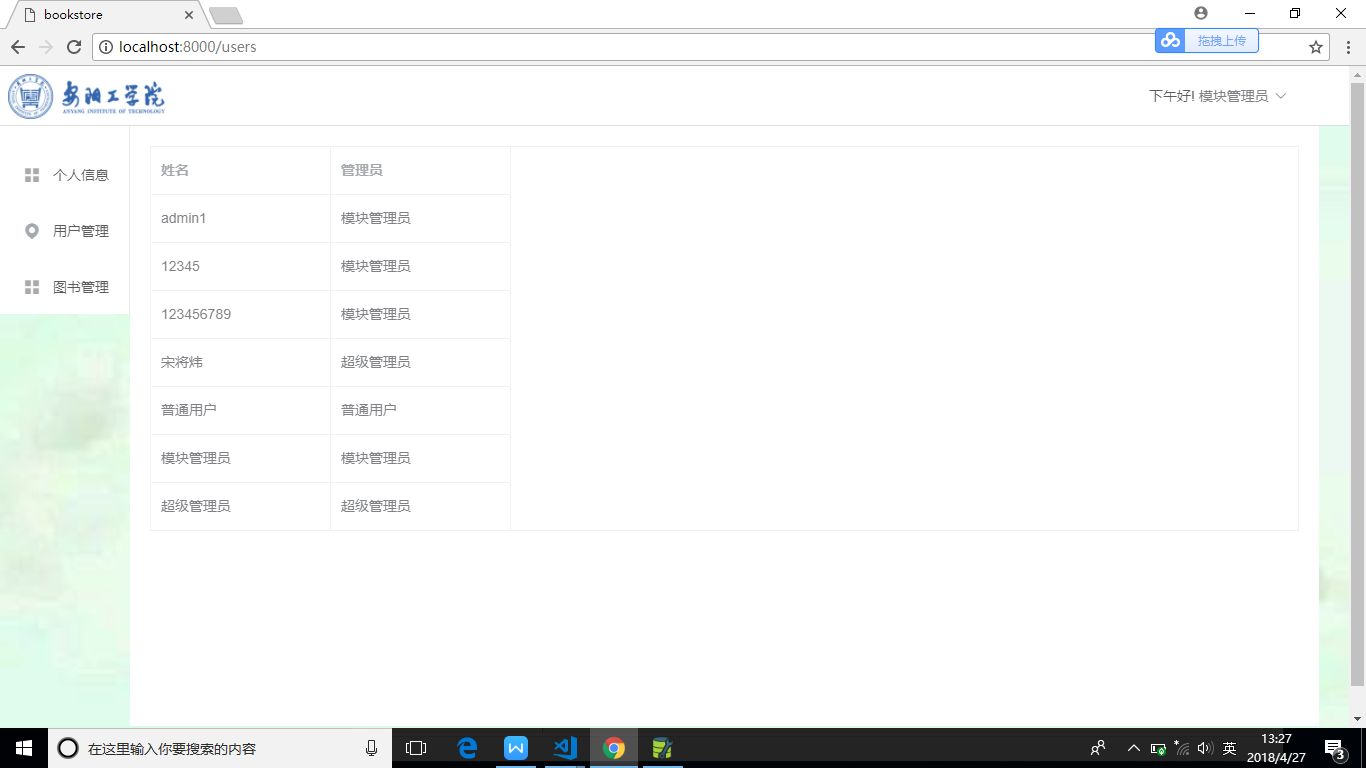


图3-20 模块管理员用户管理模块

## 3.4普通用户模块

对于超级用户即超管，通过主页的后台管理进入到管理页，可以查看个人的图书借阅情况，以及密码修改情况，设置管理员以及用户的删除，图书管理以及公告管理功能。

在首页，用户可以查看图书的详情，以及借阅图书，以及电子版图书下载，具体如下图3-21所示



图3-20 模块管理员用户管理模块

点击图书右下角的借阅，如下图3-22所示

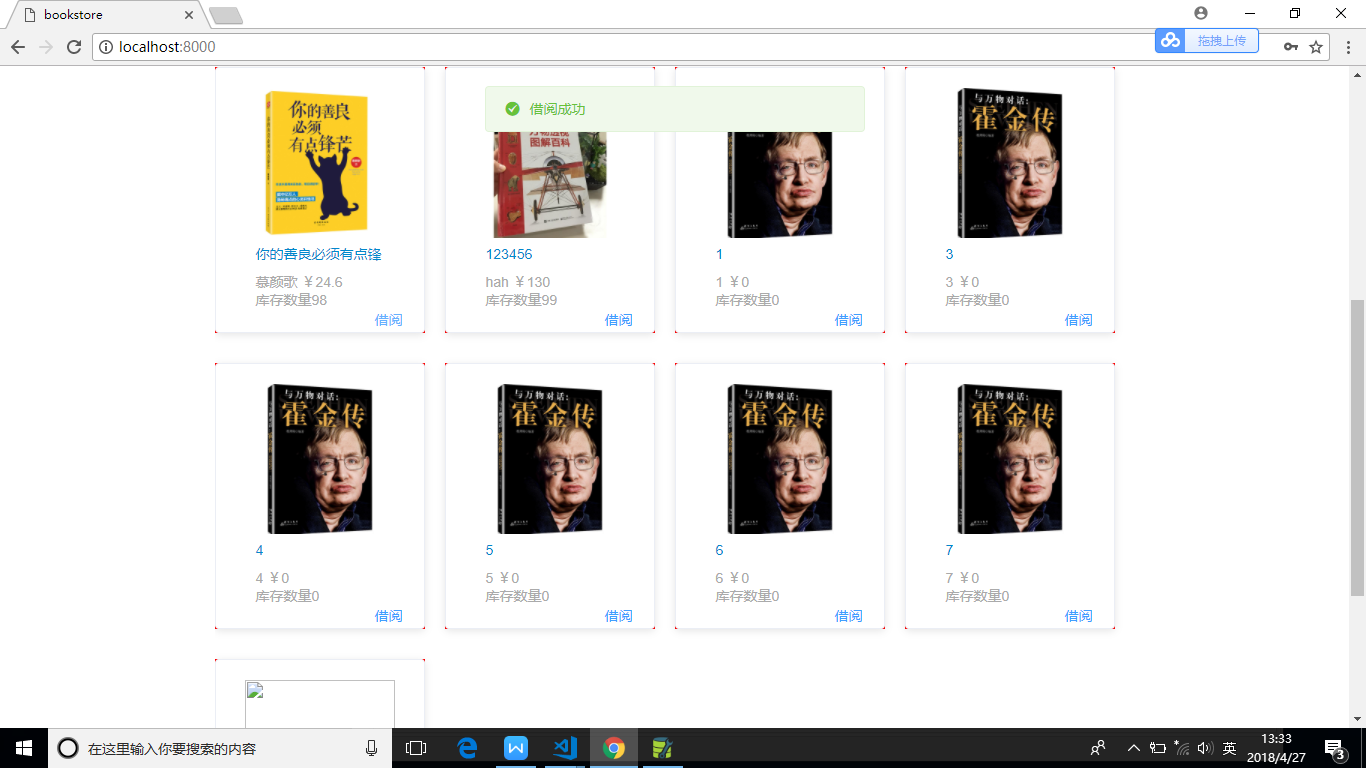


图3-22 用户图书借阅

进入后台管理，可以查看到我所借阅的书籍，如图3-23所示

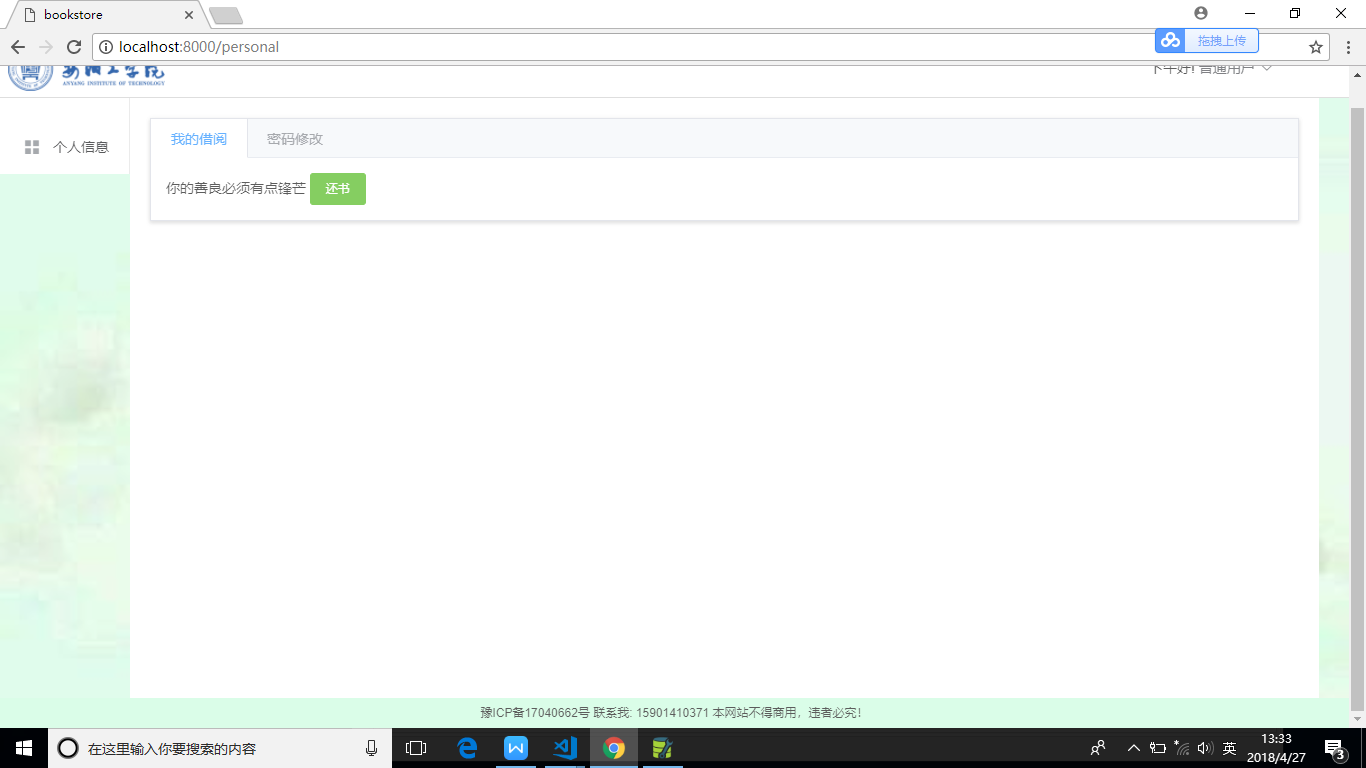


图3-23 用户所借图书展示

点击图书后面的归还，可以将借阅的图书进行归还如图3-24所示

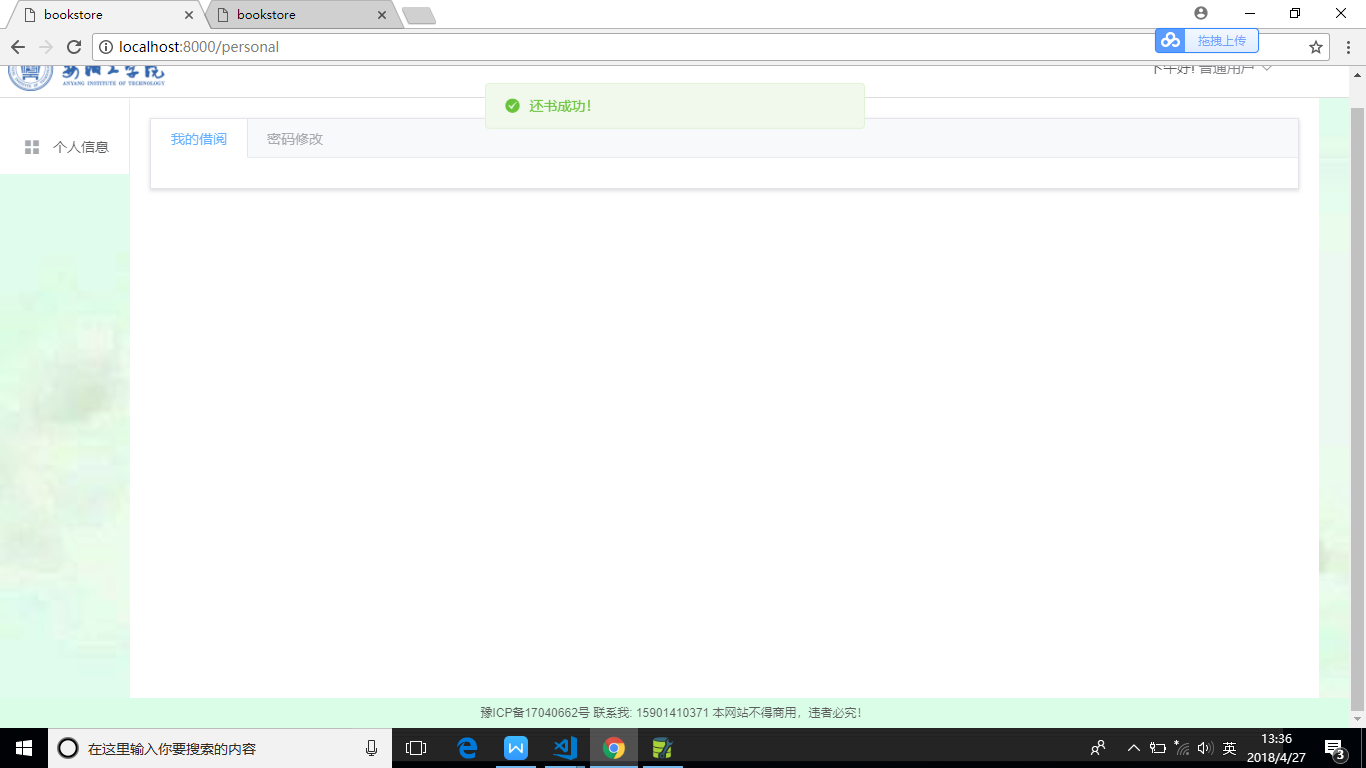


图3-24 用户归还所借图书

当你觉得你的密码不安全时，点击个人信息下修改密码可以修改你的个人密码，需要您输入你刚注册时的密保信息，如果你忘记了你的密保信息，可以到图书馆联系相应工作人员帮你解决。如图3-25，3-26.，3-27所示

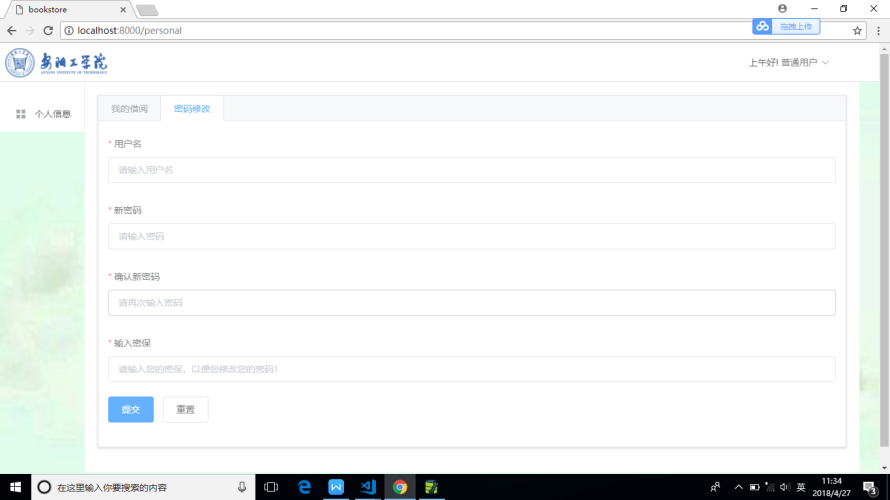


图3-25 用户修改个人密码页面

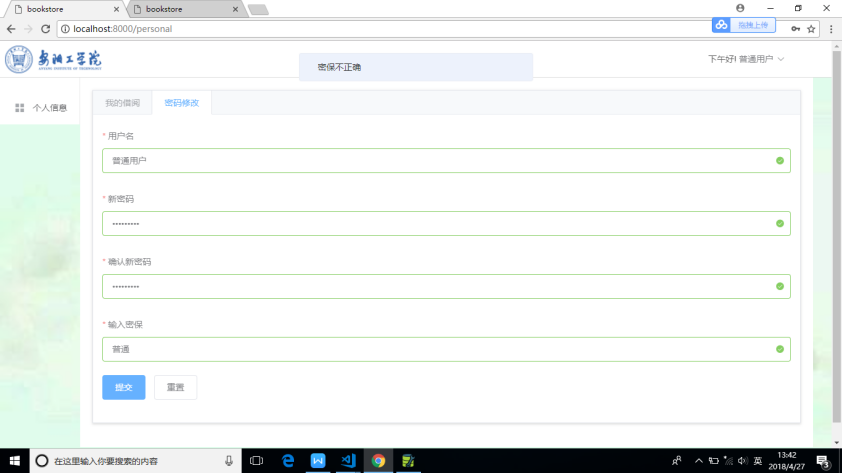


图3-20 用户输入密保错误页面

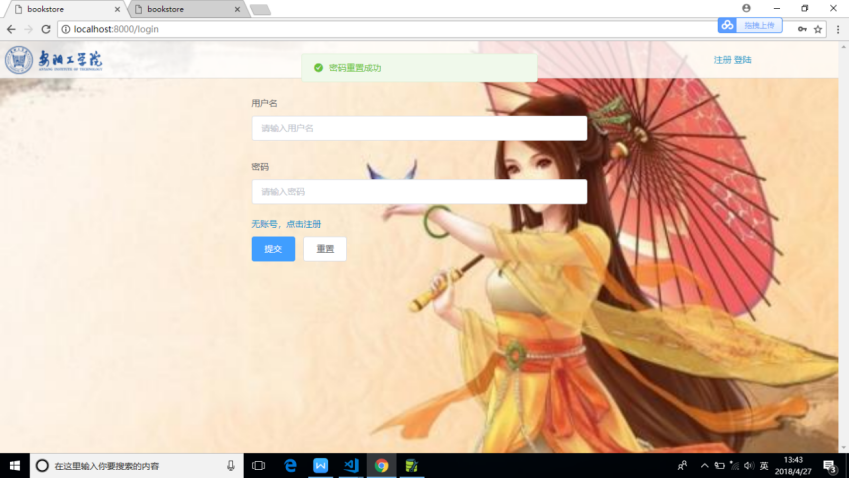


图3-20 密码修改成功页面

# 第4章 系统测试

截止到目前为止，整个系统的功能已经实现完成，接下来便是系统开发中的另一个重要环节，系统整体功能性测试，在公司里面，每一个系统功能在开发完毕时，都会交由测试部分去检测，通过对系统功能的测试，找到系统开发过程中产生的逻辑漏洞，以及权限越权等等，杜绝或者减少上线后出现的功能性问题，避免用户在使用过程中出现问题，减少客户投诉现象的发生。

## 4.1什么是黑盒测试

黑盒测试法也称[功能测试](https://baike.baidu.com/item/%E5%8A%9F%E8%83%BD%E6%B5%8B%E8%AF%95/10921202" \t "https://baike.baidu.com/item/%E9%BB%91%E7%9B%92%E6%B5%8B%E8%AF%95%E6%B3%95/_blank)或[数据驱动测试](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E9%A9%B1%E5%8A%A8%E6%B5%8B%E8%AF%95/1069827" \t "https://baike.baidu.com/item/%E9%BB%91%E7%9B%92%E6%B5%8B%E8%AF%95%E6%B3%95/_blank)，它是在已知产品所应具有的功能，通过测试来检测每个功能是否都能正常使用，在测试时，把程序看作一个不能打开的黑盒子，在完全不考虑程序内部结构和内部特性的情况下，测试者在程序接口进行测试，它只检查程序功能是否按照需求规格说明书的规定正常使用，程序是否能适当地接收输入数锯而产生正确的输出信息，并且保持外部信息（如数据库或文件）的完整性

## 4.2 黑盒测试环境

本系统的测试为常规的功能性测试，需要一台4G以上内存的WIN10操作系统的电脑，配置有NODE，MongoDB，软件及相应的运行环境，Chrome浏览器或者常规浏览器，由于选用的技术栈比较新，目前不支持IE9以下或者老式浏览器版本。

## 4.3 黑盒测试方式

本次系统采用黑盒测试方式去系统整体功能进行测试，真个测试步骤围绕权限步骤来进行，检测项目开发是否与需求预期寄过一致，测试根据需求文档的描述进行功能测试，看看开发是否实现了这些功能，有没有出现与文档描述不一致的地方，有没有越权问题的出现，以及主体功能是否阻塞，对于测试人员来讲，需要懂一点浏览器测试的基础知识，能够看得懂报错信息，给出正确的反馈。

## 4.4 测试环节

### 4.4.1超管用户的权限及系统功能

对该模块的主要测试是按照需求文档的要求进行测试，主要包含管理员是否可以将用户进行设置模块管理员，模块管理员设置为普通用户，用户的删除，图书管理模块图书的录入，编辑以及删除，公告的添加，编辑与删除等，其测试的结果如下表4-1所示

表4-1 超管用户的权限及系统功能

测试结果表

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 功能要求 | 测试结果 | 测试结论 |
| 个人信息 | 归还图书以及密码修改 | 功能实现 | 通过 |
| 人员管理 | 设置模块管理员，设置普通用户，人员删除 | 功能实现 | 通过 |
| 图书管理 | 图书录入，编辑，删除，搜索，筛选 | 功能实现 | 通过 |
| 公告管理 | 公告查阅，添加，编辑以及删除 | 功能实现 | 通过 |

### 4.4.2 模块管理员的权限及系统功能

对该权限的管理员模块进行测试主要是测试能否修改个人的信息，添加新的书籍，对自己添加的书籍进行删除、更新、查看操作，能否查看用户信息列表。该模块的测试如表4-2所示：

表4-2 “admin”权限管理员模块测试列表

测试结果表

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 功能要求 | 测试结果 | 测试结论 |
| 个人信息 | 归还图书以及密码修改添加 | 功能实现 | 通过 |
| 人员管理 | 系统中人员查看 | 功能实现 | 通过 |
| 图书管理 | 图书录入，编辑，删除，搜索，筛选 | 功能实现 | 通过 |

### 4.4.3 普通用户模块功能测试

对普通用户模块进行测试主要是测试能否修改个人的信息，查看系统中的公告信息，图书借阅，图书归还，密码修改，查看图书详情，该模块的测试如表4-3所示

表4-3 普通用户模块功能测试

测试结果表

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 功能要求 | 测试结果 | 测试结论 |
| 公告 | 查看系统中所有的公告信息 | 功能实现 | 通过 |
| 搜索图书 | 输入书名，进行图书的模糊搜索 | 功能实现 | 通过 |
| 筛选图书种类 | 选择分类，进行图书种类的查询 | 功能实现 | 通过 |
| 借阅图书 | 选择图书，点击右下角进行图书借阅 | 功能实现 | 通过 |
| 查看图书详情 | 点击图书名称，查看图书详情 | 功能实现 | 通过 |
| 电子版图书下载 | 点击下载链接，下载电子版图书 | 功能实现 | 通过 |
| 借阅图书归还 | 在个人信息模块里归还借阅的书籍信息 | 功能实现 | 通过 |
| 密码修改 | 在个人信息里输入密保进行密码修改 | 功能实现 | 通过 |

### 4.4.3 登陆注册模块测试

对该模块采用最简单直接的测试，测试新用户的注册以及注册账号的登陆，其测试结果如图表4-4所示

表4-4 登陆注册模块测试

测试结果表

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 功能要求 | 测试结果 | 测试结论 |
| 注册 | 新用户注册用户名不能重复 | 功能实现 | 通过 |
| 登陆 | 注册用户及现有用户登陆到主系统中 | 功能实现 | 通过 |

## 4.4 测试结果

本次测试环节已经大体完成，对测试环节中所有模块进行系统的测试没有发现任何问题，功能都完全得到实现，所有的测试环节结果都与预期结果一致。

## 4.5 本章小结

## 本章主要向大家阐述系统开发过程中测试功能环节，详细的向大家介绍了黑盒测试的概念以及对整个系统进行功能性的测试，对每一个测试环节给出对应的测试结果，整体功能在运行上没有任何问题，由于本次测试采用的是黑盒测试，并没有对数据库以及代码逻辑进行测试，所以测试深度还需要进一步的深入，需要一名专业的测试人员操作。结 论

本系统基本完成了智慧图书管理平台的预期功能，该系统是一个智慧型图书馆里平台，旨在学生能够参与进来，方便快捷的查阅图书馆中典藏的书籍以及快速借阅，该平台通过用户层面将系统中管理员与普通用户两种群体合理的分割开发，管理员负责维护图书馆平台的运转与稳定，用户可以快速找到自己想要书籍，节省了自己借书的时间，同时，也是得图书馆的作用得到充分的发挥，真是一举多得，在该系统中，最高级用户超级管理员需要到数据库里面去操作，在系统中只能设置普通管理员也即是模块管理员的权限，这样防止一些不必要的事件的发生，模块管理员除了不能操作用户及发布公告外，其他的权限和超级管理员是一致的，同时超级管理员和普通管理员，用户都享有图书借阅的权利，都可以借阅书籍，最多可以借10本书籍，作为普通用户，享有公告查阅，图书借阅，图书归还，密码修改，电子版图书下载等功能。

该平台最大限度让学生作为图书馆的一员参与进来，学生可以了解到图书馆的最新信息，而且还可以在平台上进行图书的借阅，归还，查阅所需书籍信息状况，大大减少了学生借阅书籍，查找书籍所消耗的时间，为了保证系统的可扩展性，多数地方采用枚举的方式实现，只需修改数据库的值就可以达到相应的效果，对后期的二次开发十分有利。

在本系统实现过程中，多次进行系统测试，保证系统功能的稳固，防止用户在使用过程中出现一些问题，给用户带来损失及不便，总体来说，这次智慧图书管理平台完美完成。

**致 谢**

从刚开始的选题到现在整体项目的成型，毕业论文的撰写，已经差不多半年的时间了，早在大三时我就开始跟着老师学习前端方面的知识，因为主攻前端技术方向，对数据库以及后台语言的了解非常的少，基本是一张白纸的状态，在我开发过程中，遇到了很多关于后台以及数据库方面的问题，我一边查阅资料，一边向老师和懂该技术的同学咨询，在他们的帮助下和指导下，我才得以顺利将整个系统的开发工作完成。

说实话，在开始选择这个题目的时候我也是捏了一把汗，这完全是接触一个新的知识，新的技术，刚选定题目的几天我很惶恐不安，害怕自己没有能力完成这个下项目的开发，在张炜老师的鼓舞下，我选择接受挑战，人生就是不断的勇攀高峰，如果永远只追逐容易得到或容易实现的事务，那么生活还有什么意义，我在开始的一个多月里，查阅很多网上的资料和视频去学习这些项目中用到的后台技术，周末时间找张炜老师询问学习过程中不明白的地方，老师不惜消耗自己宝贵周末的时间一一为我解答，每每想到此时，就觉得非常感动，选择张炜老师做我的毕业导师，我真的很幸运！感谢张炜老师在这大四一年里的辛苦奉献，我将时刻铭记于心，谢谢！

毕业设计顺利的完成，同时也离不开大学四年里教育我的老师们，是他们给我指引了前进的道路，也许只是他们不经意间的举动，却将我的整个星空燃动，在此，我要对他们说声谢谢，同时也感谢在我前进道路不断遇到的困难，是它们教会了我成长，是它们教会了我要坦然面对，激流勇进，最后，感谢所有帮助过我的人，特别是我的那些可爱的室友，是你们在背后默默的支持才让我走到了今天，谢谢你们，对于你们默默无闻的奉献，我将永远铭记于心，最后，衷心的祝愿所有人都事事顺心，万事如意！。

# 参考文献

1. Node.JS中文网 <http://nodejs.cn/api/>
2. MongoDB 教程 <http://www.runoob.com/mongodb/mongodb-tutorial.html>
3. Vue.js <https://cn.vuejs.org/>
4. Vuex <https://vuex.vuejs.org/zh-cn/>
5. Vue-router <https://router.vuejs.org/zh-cn/>
6. ES6 <http://es6.ruanyifeng.com/>
7. Html5教程 <http://www.w3school.com.cn/html5/index.asp>
8. Express http://www.expressjs.com.cn/
9. Moment时间转换插件 <http://momentjs.cn/>
10. Element前端UI框架 <http://element.eleme.io/#/zh-CN>
11. 百度百科
12. npm <https://www.npmjs.com/>
13. GitHub
14. 陈明 《软件工程》 中国铁道出版社 2011年3月
15. 百度文库 <https://wenku.baidu.com/>
16. 思否 <https://segmentfault.com/>
17. 有道翻译 <http://fanyi.youdao.com/>
18. javascript高级程序设计 (美)(Nicholas C.Zakas)扎卡斯 中国邮电大学出版社
19. 郭家宝．《Node.js开发指南》[M]．北京：人民邮电出版社，2012年7月
20. （美）休斯-克劳奇（Hughes-Croucher，T.）,（英）威尔逊（Wilson，M.）．《Node即学即用》．北京：人民邮电出版社，2013年1月
21. 朴灵．《深入浅出Node.js》[M]．北京：人民邮电出版社，2013年12月