

毕业论文



题 目：基于MVC模式的图书管理系统的

设计与实现

姓 名：刘修平

学 号：1605120614

指导教师(职称)：邬少飞（副教授）

专 业：数字媒体技术

班 级：2016数字媒体技术03班

所 在 学 院：计算机科学与工程学院

2020年6月

|  |
| --- |
| **基于MVC模式的图书管理系统的设计与实现** |

|  |
| --- |
| **Design and implementation of book management system based on MVC pattern** |

**学生姓名 刘修平**

**指导教师 邬少飞（副教授）**

毕业设计（论文）诚信声明书

郑重声明：

本人所提交的毕业设计（论文）《基于mvc模式的图书管理系统的设计与实现》是本人在指导教师指导下独立研究、写作的成果。论文中所引用的他人的文献、数据、图表、资料均已明确标注；对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式注明并表示感谢。

学生签名：  日期：2020年 04月 30日

郑重声明：

学生 刘修平 所完成的毕业设计（论文）《基于mvc模式的图书管理系统的设计与实现》是在本人的指导下独立研究、写作的成果。

本人已认真阅读了该生所提交的毕业设计（论文）所有文档，并做了认真批阅。

指导教师签名：IMG_256 IMG_256 IMG_256 日期：2020年5月13日

目 录

[摘 要 II](#_Toc26379)

[Abstract III](#_Toc19918)

[第一章 绪 论 1](#_Toc14381)

[1.1 课题背景 1](#_Toc22986)

[1.2 系统的目的及意义 2](#_Toc23735)

[1.3 国内外研究现状 2](#_Toc16087)

[1.4 主要工作及论文结构 3](#_Toc12591)

[第二章 系统开发技术 5](#_Toc17325)

[2.1 JSP技术 5](#_Toc29844)

[2.2 JavaScript 6](#_Toc31209)

[2.3 B/S结构 6](#_Toc28224)

[2.4 HTML5 7](#_Toc24295)

[2.5 MySQL 7](#_Toc21471)

[第三章 系统分析 9](#_Toc20572)

[3.1需求分析 9](#_Toc30758)

[3.2可行性分析 10](#_Toc23224)

[3.3系统流程分析 11](#_Toc20519)

[3.4系统用例分析 16](#_Toc22422)

[3.5业务流程分析 17](#_Toc3201)

[3.6 数据库分析 18](#_Toc17614)

[第四章 详细设计 21](#_Toc11440)

[4.1 系统概述 21](#_Toc26192)

[4.2 系统功能设计 21](#_Toc16314)

[4.3 数据库设计 22](#_Toc18198)

[第五章 功能模块实现 25](#_Toc25447)

[5.1读者模块 2](#_Toc28600)5

[5.2管理员模块 2](#_Toc32165)7

[第六章 系统测试 33](#_Toc30514)

[6.1测试意义 33](#_Toc11451)

[6.2 测试环境 33](#_Toc16641)

[6.3 测试结果 34](#_Toc21311)

[总 结 3](#_Toc9793)7

[致 谢 3](#_Toc27444)9

[参考文献 4](#_Toc4322)1

摘 要

进入二十一世纪以来，科学技术和经济建设都取得了举世瞩目的成就。我国人民群众的生活水平也在不断的提高，对于精神文明建设的需求也在逐渐的扩大。图书作为历史文化的传承载体之一，对于满足人们的精神渴求有着无法替代的强势作用。可以注意到的是，不仅仅国外的有那些所谓的世界名著，我国各种历史上流传下来的经典传世之作也是不可胜数的，而在现代社会，随着新文化的不断衍生繁荣，各种小说期刊和漫画的出现，这些使得中国的图书的市场在极速的发展，不断的在我们的日常生活中产生影响。为了使得我们人们可以更方便的阅读图书，基于计算机这种先进技术的配套服务越来越多，而针对图书馆的图书借阅问题，图书管理系统就应运而生。

本论文最开始就对基于MVC模式的图书管理系统的研发背景和国内外的发展现状进行了一次详细介绍。通过现如今国内外的具体发展的趋势，对这个图书管理系统进行了需求分析，然后基于具体要实现的功能制定了系统需要进行设计的目标，需要进行方案的预设，然后展示本系统的详细的具体设计以及最后的具体实现情况，最后的关键自然就是系统的测试，在这个环节针对系统的各个具体的功能实现进行了多次测试，最终确定了各个功能是能够完美运行的。

在图书管理系统的详细设计中，本文以当下最规范的的MVC模式开发指导，在系统的具体实现上我们也采取了时下最为流行的技术，JSP技术和MySQL数据库的技术，结合B/S结构进行系统的开发。

关键词：JSP技术；MVC模式；B/S结构；

Abstract

At the beginning of the 21st century, the achievements of science, technology and economic development attracted worldwide attention. The living standard of our people is also constantly improving, and the demand for the construction of spiritual civilization is gradually expanding. As one of the carriers of history and culture, books play an irreplaceable role in satisfying people's spiritual aspirations. Can be noticed that those so-called world more than foreign classics, various history of our country and is a classic masterpiece of also is cannot be counted, and in the modern society, with the continuous derivative of new culture prosperity, all kinds of novel, periodicals, comics, which makes the development of China's book market in speed, constant influence in our daily life. In order to make it more convenient for people to read books, more and more supporting services based on the advanced technology of computer, and in view of the library's book borrowing problem, the book management system came into being.

At the beginning of this paper, the research and development background of library management system based on MVC model and the development status at home and abroad are introduced in detail. Through the concrete development trend both at home and abroad, and now on the books management system management system demand analysis, and then based on the concrete in order to realize the function of the developed system needs to carry on the design goal, need to plan the preset, and then show the detailed design of this system and the concrete implementation situation, the key is the system of natural test, in this link for the realization of the function of the system of various specific repeated testing, finally determine the various function is the ability to run perfectly.

In the detailed design of the library management system, this paper based on the current most standard MVC model development guidance, in the specific implementation of the system we also adopted the most popular technology, JSP technology and MySQL database technology, combined with B/S structure for the development of the system.

**Keywords**：JSP technology；The MVC pattern；B/S structure；

# 第一章 绪 论

在第一章章主要针对本次毕业设计的这个课题的课题背景和研究现状，结合国内外图书管理系统的发展现状来谈谈本次毕业设计的主要工作，通过与当下的发展具体现状的结合来设计这个基于MVC模式的图书管理系统。

## 1.1 课题背景

随着时代的发展，科学技术经常有新的突破，导致社会面貌在每一年都会出现很多不一样的变化。经济水平的提升使得民众对于自己精神生活进行提升的意识越来越强烈，在加上我国对于教育事业的的大力投入，人民群众的知识水平不断上升，文盲率不断的下降。在当下阅读图书不再同以前那般是少数精英分子的特权，而是人人都有书读，因此，对于图书的需求在快速的膨胀[1]。图书馆作为一个收藏图书最为丰富的场所，自然也水涨船高成为人们进行精神文明提升的好去处。随着现在人们的生活质量的提升，人们的精神生活质量不断的自我提高，我国各地出现了各种各样的大大小小的图书馆，不仅各个市县大都设立公立图书馆，各个高校也是都拥有着自己的图书馆。以武汉工程大学为例。武汉工程大学的图书馆已经建立了三十多年了，总建筑面积达到了两万六千八百五十九平方米的，拥有着两千零一十八个阅览座位，在校读者有一万五千人，在库图书更是多达二百六十多万册，已然很大规模[2]。图书馆的不断增加，而人们也都愿意去图书馆看书，因此对于图书管理方提出的要求也就更多，图书管理者的责任也就越大。如何更加有效的进行图书管理就成为了摆在图书馆眼前的问题了。

如今的社会，在进入了二十一世纪以后，网络的建设如火如荼，尤其是近几年几乎是一年一个样，网络普及也已经达到了一个很高的程度，我们的生活与互联网息息相关，我们的日常生活的方式与互联网技术相结合产生了各种各样的改变。在过去我们的图书管理工作基本都是由人力来完成，现在结合计算机技术，我们就可以使用管理系统来代替人工的方式，实现更加高效的工作。这样革命性的改变对我们的生后产生了极大的方便。因此为了进行更加有效的进行图书管理，我们开始尝试图书管理系统[3]。

在图书管理系统出现之前，我们的图书馆使用着人工管理的方式，这种细致而费力的工作往往由人工完成，但是人工完成有着显著的问题，首先人工容易出错，正所谓百密一疏，在面对浩如烟海的书籍，要想把这么多的图书精准的归类，还有需要处理每日剧增的用户的借阅图书和归还图书的个人信息，不出错实在是一件很困难的事情，一旦出现了纰漏，对借阅书籍的读者和管理图书的管理者都会造成不小的困扰。其次，人工的效率也是不高，即便是技术娴熟的管理员。在面对大量的图书和大量的用户的时候也会分身乏术，这直接影响着图书馆的运营，因为必须雇佣大量的工作人员。而说到严谨的工作和高效的工作效率那正是计算机这种机器的强项。就像过去的机器取代手工工人一样，用计算机技术编写出来的程序来取代人工正是时代进步的体现。图书管理系统在这种背景下应运而生，借由缜密的计算机来代替传统而又低效的人工，信息化的方式使得图书的各种信息一目了然，读者方便挑选找寻到自己想要的关键的书籍，而对于管理者来说，不需要雇佣大量的人工去进行那低效而又容易出错的手工操作，绝大多数的细致的工作佑高效的计算机来完成。利用计算机体系化的进行图书管理，这种高度信息化，智能化的管理方式越来越被广大的群众喜欢与接受。管理操作的信息化使得基于MVC模式的图书管理系统变得十分的简便，可以设想依靠这计算机技术的不断进步，基于MVC模式的图书管理系统将会更科学合理有着更多的功能[4]。

## 1.2 系统的目的及意义

图书管理系统的出现，是为了解决利益日益增长的图书需求和有限的图书馆管理资源之间的矛盾。为了实现图书馆更加高效的运作，给与用户更加良好的借阅体验。对于用户而言，用户可以使用图书管理系统来进行查询图书的服务，也可以使用图书管理系统实现图书的借阅和归还。系统通过数据库进行实时的数据处理，实时完成用户提交的操作。用户在使用图书管理系统的时候仅仅需要在十分简洁的前端页面来进行操作，后台的数据处理会直接代替用户完成操作的具体内容。而针对图书管理员的工作，系统提供给管理员许多的权限，最为基础的就是管理员可以对图书的信息实现增删改查，以便图书的管理，而对于借阅图书的读者管理员能够对读者的个人账号实现管理操作，能实现以上的功能，图书管理系统已经能够完成图书馆日常的图书管理工作了[5]。

图书管理系统的出现使得图书馆的图书管理工作变得更加的高效，远比人工低的多出错可能性以及远比人工高效的处理效率，使得以图书馆来代替人工工作变得更加的理所应当。而针对过去一些已有的图书管理系统，对于系统的界面进行优化使其变得更加的简洁容易操作是很重要的问题，过去受制于计算机技术的发展，以及现代用户的审美升级，这就要求现今的图书管理系统不可如同过去一般界面生硬，这样毫无疑问会影响用户进行人机交互的体验，毕竟只有越来越新颖越来越方便操作的图书管理系统才能在用户心中占据优势[6]。

## 1.3 国内外研究现状

在步入新世纪之前，国内外的对于图书管理系统的研究还是有很大的差异的，一切的科学技术促进生产力的发展从而又辐射影响到人们生活的方方面面，但是的中国由于经济还不够发达科学技术远远落后于国外的先进国家，尤其是在计算机领域。现代计算机技术的规范基本都是由外国人进行研究制定的，因此在对于计算机技术应用到图书馆的管理工作上我们的起步远比国外要晚[7]。

国外早在上世纪五十年代就有能够实现自动化操作的管理系统，此后没过几年又是开创了目录检索。在国外图书管理系统的开发十分的早，目前为止已经有很长一段历史。现代化的自动型图书馆的诞生地是在美国，最早的图书管理系统仅仅提供了单词检索匹配功能，这是在一九四五年年由美国海军兵器中心进行实现的。在一九八零年以后，国外已经能够全面实现网络化和现代化而中国此时远远没有这种概念，国外的发展的方式以及讲究的内容远比国内先进与科学，在九十年代，国外更加大力进行图书馆的建设工作，仅仅美国在这几年就投入了三点六四亿美元来支持建设。在九十年代的中后期，由于Windows图形用户界面的推出使用，以及一些列Java技术，数据库技术的出现，许多国外的厂商开始意识到图书管理系统可以用这些技术来支持更加高效的工作。因此，国外就开始出现了具有现代结构的图书管理系统[8]。

而国内关于图书管理系统的研究，在二十一世纪以后开始了突飞猛进的成就，由于我国的经济实力的不断提升，加上不断的科学技术的创新，我国的硬件设施不断的更新换代，软件技术不断的进行升级。这让图书管理系统的发展得到了全面的提升，我们也全面开始摒弃了过去的人工操作的方式，替换成现代化的信息管理系统。到目前为止我国的在图书管理系统方面也取得相当的进步，本次系统的设计就是要在过往的图书管理系统的基础上进行一次完善，在系统的操作界面，和具体的功能上进行一次完善。让图书馆的用户能有一份更好的借阅体验[9]。

本次基于MVC模式进行设计的图书管理系统，我们采用的都是时下最为流行的开发技术，实现的都是一些平时里进行图书管理最为需要的操作。功能多种多样这是国内的图书管理系统的共性。而在国外，他们的管理系统主要是具有针对性，更加专注于那些专业的功能。因此我们需要结合一下，针对核心功能进行优化，打造出功能更加多样的专业化图书管理系统。

1.4 主要工作及论文结构

1.4.1 本课题进行的主要工作

根据用户的需求进行分析。深度研究后,开始设计系统的具体内容,我们从这个系统的需要进行实现的功能开始分析，对比已经存在的图书管理系统我们要实现基本的功能，在我们对于系统有了一个概念性的设计以后，就可以进行更加详细的分析，在IDEA开发工具上对每一个功能的实现代码进行编写，在再整合各个部分以后我们就可以运行出具体的系统。

在开发出系统后，我们还要进行一下测试，以此保证我们的系统是可以在服务器中正常的进行运转的，并且所拥有的功能目前基本可以满足用户需求，我们在对比较市面上已经存在的图书管理系统，比如各个高校的图书管理系统，具体参考武汉工程大学的图书管理系统，我们可以小细节上进行深度的优化，这样才能够使我们的图书管理系统更加的完善，完成出一个功能比较全面的图书管理系统，最后一步要进行毕业论文的撰写工作。在现代这个计算机技术已经高度发达的社会，管理系统的设计已经逐渐变得成熟，这个基于MVC模式的的图书管理系统系统的具体实现，主要借鉴了当今高校的图书管理系统，进行了这个与当下的图书管理工作相差不大的系统实现项目，我们在界面上进行简化，在功能上进行了完善，最为重要的是对我所学的知识进行了一次系统的巩固，通过这样的综合性的实践来检验这四年的学习成果。

1.4.2 论文的主要结构

第一章是绪论，首先针对课题的背景进行一次详细的介绍，通过对人们对于图书的需求以及图书馆的不断建立，来分析我国现在对于图书管理的需求，这就是我们研究的意义，最后我们对比国外与国内的图书管理的历史，来针对国内外现状进行一次十分系统的阐述。以此作为课题研究的基础准备工作。

第二章是系统开发技术，对于此次系统开发的所使用的关键技术进行了一次详尽的介绍，针对系统开发的JSP技术、JavaScript、HTML5等开发技术进行了介绍，对于开发系统采取的B/S结构也进行了介绍，最后对于实现增删改查的数据库方面的知识也进行了仔细的介绍。

第三章是系统分析，在具体的设计我们系统之前我们要针对我们系统进行各方面的分析，首先就是需要实现哪些功能，针对具体的要求我们展开对系统的需求情况做一次分析。对于我们设计这个系统是否可行，我们需要进行可行性的相关分析，从多个角度立体的来确定我们设计的可行性。紧接着就是对于我们系统的流程以及用例进行分析等等，通过这些详细的分析来展示我们系统的轮廓。

第四章是详细设计，在分析我们系统之后，我们对于系统就要展开具体的设计工作了。为了准确的设计出图书管理系统，我们要聚焦到系统的具体功能上来，针对系统的功能模块来进行针对性的设计工作，最后最为重要的是数据库方面的设计，严格按照数据库开发的流程对于数据库进行合理的设计工作。

第五章是功能模块实现，针对具体的功能模块的设计，我们需要在开发工具上编写出能实现具体功能的代码，通过这些来实现我们系统的具体功能。依据系统流程的设计，从管理员和读者两个模块开始对于所有的功能进行一次详细的介绍。

第六章是系统测试，在实现了图书管理系统之后，为了保障系统的稳定性以及安全性方面的要求，因此就需要进行科学的测试，在具体面向各个功能我们分别进行多次重复的测试，反复确认功能的可运行，以此来保障系统的可靠性。

论文的最后就是这次毕业设计的总结与致谢，总结分析一下这个图书管理系统的设计，以及这次设计的不足之处与对未来的展望。在全文的最后我对于培育我的学校以及这几年对我照顾有加的老师同学们进行一一感谢。

第二章 系统开发技术

在本章将会对于开发系统的关键技术进行介绍，主要针对JSP技术和JavaScript等知识进行背景以及应用等方面的介绍。

在开发图书管理系统的工具的选择上，我比较了许多软件，最终选择了目前Java开发最普遍的IDEA进行系统代码的编写，使用的是最适合开发使用的MySQL这种关系型数据库，对于数据库的管理工具则是使用了Navicat，还有Tomcat8.5版本的服务器，最后选择谷歌浏览器作为系统运行的浏览器。

2.1 JSP技术

JSP即Java Server Pages。提到JSP技术的开发，自然又离不开发明Java语言的Sun公司，JSP技术正是这个公司开发的一种Web页面开发技术。JSP是一个在研发动态网页的过程种必须要遵守的以一个技术标准。在开发网页的时候，对于静态的资源我们通常使用HTML就行了，而对于动态资源，就需要更加强大的技术，于是JSP技术就应运而生，也是因为JSP技术十分的强大，几乎所有的服务器都可以兼容使用[10]。

在原有的静态HTML页面种可以写入Java代码片段，这样就构成了动态的JSP页面，在这个动态页面种不仅仅可以正常的显示HTML页面，通过其中的Java代码也可以进行一些数据操作。最开始开发者们使用HTML进行页面的开发，为了实现动态资源的创建于是使用了JSP技术，这说到底就是Servlet技术的升级，在编写完JSP页面之后，我们在运行系统的时候就会对于JSP代码进行编译工作，这个过程就会进我们需要的动态资源加载到服务器中，最终在我们的谷歌浏览器中就会把HTML页面展示出来，所有的动态资源也可以实现，而正是由于需要进行编译执行的特性，所以不同于HTML可以直接在浏览器种运行，JSP页面却无法这样，因为其中的Java代码片段必须要运行在服务器中，因为这个原因JSP也必须借助服务器运行。所以究其本质，JSP就是一个Servlet,不过这个是一个简化的Servlet程序[11]。

在本次基于MVC模式的图书管理系统的开发中，作为页面开发的关键技术，使用了JSP技术加上Java Bean进行开发，这种方式的主要优点如下：

1、JSP技术只要是利用了Java Bean组件进行便捷操作。Java Bean组件其实就是由日常使用的程序写成，我们在实现具体功能的很多时候，有一部分的代码会作为重复的代码，进行不必要的重复编写工作，而作为这种功能性的代码，我们将其封装到Java Bean中的话，那么在需要使用的时候就可以进行直接调用，这么一来就实现了代码复用，我们开发系统的时间就会大大减少。

2、在我们以使用JSP技术后，由于过于简便，开发者经常把Java代码写在HTML代码中，这样反复的嵌套最后越来越多直接导致代码的可读性变得极低，因此为了分离我们就使用了Java Bean，这样一来代码就简化了，开发者维护起来也容易。

2.2 JavaScript

JavaScript与Java等高级语言不一样，是一种较为特殊的脚本语言，在web开发的过程中被进行广泛的应用。在一九九五年，Netscape公司在网景导航者这个浏览器上进行设计首次实现。JavaScript的标准是ECMAScript，这个保准几乎所有的浏览器都进行支持[12]。

JavaScript的运行十分具有特点，它是一种解释性的语言，代码不会先进行预编译，而JavaScript是在程序的运行过程中逐行进行解释。主要的作用就是为HTML页面添加一些实用的交互行为，这些操作使得页面与人的交互体验变得更加的良好。JavaScript有两种引用方式，首先JavaScript可以直接嵌入到HTML页面中，但更多的是同CSS样式表一样直接单独封装成一个JS页面，这种操作有利于结构和行为进行分离[13]。

JavaScript具有事件驱动的特性，具体来说，用户使用鼠标对页面进行操作，再或者使用键盘输入进行操作的时候，这些事件都会驱动JavaScript程序,这些由开发者编写的相关处理这些事件的程序来做出对这些事件的系统响应。JavaScript脚本语言同其他语言一样，有它自身的基本数据类型，表达式和[算术运算符](https://baike.baidu.com/item/%E7%AE%97%E6%9C%AF%E8%BF%90%E7%AE%97%E7%AC%A6/9324947)及程序的基本程序框架。JavaScript提供了四种基本的数据类型和两种特殊数据类型用来处理数据和文字。而变量提供存放信息的地方，表达式则可以完成较复杂的信息处理[14]。

2.3 B/S结构

在软件开发的历史中，主要有着三种体系结构。第一种就是集中式的结构，但是这种结构已经逐渐式微，因为这种结构存在着很多很明显的缺点，首先，信息传输的困难、信息共享的困难、完全封闭的系统无法很容易地与外界交流，系统延展性非常差，因此已经淘汰了这种结构，使用的就慢慢变少了。现在的还广泛进行应用的就是两种结构，一种是B/S结构，另一种则是C/S结构的，也存在两者混合使用的软件开发的情况[15]。

随着时代发展，计算机技术不断的迭代进化，慢慢的有一种Client /Server的结构就被广泛使用了，这种结构简称C/S模式，这种模式就是既需要有服务器，还必要有客户端。基于这种C/S模式开发出了许许多多的应用软件，这种模式有着很多优点。首先在使用C/S结构的软件的开发过程，由于同时存在着客户端和服务器，那么这种软件的整体开销自然就会很合理的分配到两端的硬件配置。基于C/S结构的程序，必须要本地安装好一个完整的客户端，这样维护更新起来就必然造成一定的困难。你在服务器端的更新不够，在客户端也要进行同步的更新，在必要的时候还要进行重新安装，这样实在是不够方便。而用户一旦与主机建立了连接，为了稳定的数据传输，那么这个连接就一直不会自己断开直到用户自己关闭连接，这样无疑是对服务器的负担，假设用户过多，那么就无法建立新的连接，这样从另一个方面来说就是使用户数目受到了极大的限制[16]。

因此我们再设计图书管理系统时采取的是B/S的结构。基于B/S的结构运行的项目，极大减缓了客户机的工作，这样就可以适应千变万化各种各样的客户机，客户机本地仅仅需要配置一些简单的软件。B/S结构显著的一个优点就是跨平台，用户像获取信息在各种各样支持的平台上都能实现。在浏览器种主要使用的就是HTTP协议这种无连接的协议，这种协议的好处就是不需要预先建立好一条连接，在访问请求的时候才进行连接，这样一来即使有成千上万条连接请求服务器也能保障服务，即使服务器无法满足了，也可以适当的对服务器的数量进行一定程度的增加，这样一来就会让负载均衡。而且对于开的成本来说，这样结构也会有着很大的优势，维护起来也十分的简便[17]。

B/S结构相对于C/S结构的优点总结如下：

1、使用广泛：由于不需要安装客户端，用户仅仅打开自己的浏览器输入网址，即可实现对目标的访问，这样的简便操作就直接使得受众更加的广泛；

2、用户在对于系统的维护上也更加方便，不同于更新客户端的工作，由于只有服务器端需要更新数据，浏览器端仅仅展示系统，因此用户就不需要进行多余的更新操作。这就让用户的操作变得很简单了。

3、共享性：随便使用一个浏览器都可以对系统进行，用户只需要使用对应的系统网址就可以在任何的浏览器使用该系统，这样的方式使得系统的共享性极大的增强；

4、开发维护所耗费的的成本相对较低低，十分方便使用者进行操作。

但是不可否认的一点是这种结构不可避免的也存在一定的缺点，首先减轻了客户机的工作，对用户的请求的数量也有很大的包容性，这样一来毫无疑问会增加服务器的工作，不仅如此，通信的开销也会增多，数据安全也有隐患。由于面向与大众就无法做到满足所有人的习惯与审美，毕竟人与人都有着差异。

2.4 HTML5

HTML 是一种语言，但是HTML却不是一门编程语言，而是一种标记语言，就是一套标记的标签，全称超文本标记语言，用于前端页面的编写。而HTML5将成为HTML的新标准，更适合与现在的主流浏览器。使用HTML5进行页面开发，这样的页面简洁明了，方便读者使用[18]。

跨平台是HTML5具有的优势，这就意味着不管用户使用Windows操作还有苹果用户使用MAC或者是移动端的用户使用移动设备，HTML5都能够进行适配。在适配多平台的同时，针对一些设备配置较低的用户，HTML5也能够给予很好的体验，因为HTML5对于硬件的要求其实不高，仅仅依靠低配置的CPU和少量的内存，也能够展示出页面和动画。同时HTML5还进行特性上的丰富，这种丰富就解决了对外部插件比如Flash的依赖，改善用户的体验。在HTML5面世之前，在web开发中要实现的很多特性就必须依赖于Flash或者一些Java插件，但由于HTML5的产生，这种需求就不再被过度依赖了，有的甚至成为了旧时代的产物，已经被抛弃掉了[19]。

2.5 MySQL

在数据库的使用方面，基于MVC模式的图书管理系统使用的是Mysql数据库。MySQL数据库是由一家瑞典的公司进行开发的，但是是从属于Oracle公司旗下的，是一种关系型数据库。公认的在进行web开发时最方便使用的软件其中之一就是这个MySQL数据库。MySQL数据库使用许多的表进行数据的存放，这样使得运行的速度大大提高，系统因此也具有极强的灵活性。

MySQL的管理工作是使用结构化查询语言，这是目前最为标准的数据库语言。MySQL的体积不是很大，运行的速度却是很快，加上其较低的开发的成本，这就使的MySQL十分受开发者的青睐[20]。

在本次基于MVC模式的图书管理系统的设计中，对于MySQL数据库的使用是极为关键的一环，在这个系统具体的功能实现，我们进行操作时候，所有进行增删改查数据都会保存在MySQL数据库的表单当中。本次我们运用XMAPP这个数据库安装软件进行数据库的安装，这样复杂的数据库安装操作就简化了，再使用的Navicat这一款数据库管理工具进行数据库的管理操作，由于直接操作数据库界面不是很人性化，操作起来将会有一定的困难，而我们安装了这种特定的数据库管理工具将极大的方便用户对于数据库的使用，这种数据库管理工具的界面十分简单，用户十分容易学会使用，不同于普通数据库操作的英文界面，Navicat使用了我们更加能够接受的中文界面，我们使用起来难度也是骤降，而且在数据库的安全与稳定方面有着相当好的保证，不会出现数据库故障等事故，因此为了保障本图书管理系统的数据操作能够顺利无误，我选择Navicat作为此次系统设计的的数据库工具。

第三章 系统分析

本章首先针对本次设计结合当下的情况进对系统行一次需求分析，再考虑到技术和经济以及维护等方面，对系统进行一次对可行性分析。在确定系统的实现可行之后，再聚焦系统的流程，通过对于实现具体功能时候的操作流程的分析，我们来确定系统的操作逻辑，而针对系统的不同用例我们也要进行仔细的定位。最后就是关于数据库的问题，再设计系统时候我们要考虑到数据库的的具体设计，对于不同实体的属性也要有清晰的定义。这样一套科学的分析下来，设计出来的系统也会表现出优秀的性能。

3.1需求分析

在准备进行图书管理系统的开发的时候，我们第一步就是了解图书管理的过程，在对图书馆运行原理，以及已有的系统的操作进行观察了解之后，我们要对于系统应该有哪些功能做到心中有数，对于不同的功能确定不同的功能模块，而每种用户能够有什么权限能够操作什么样的模块我们都要分析，最后关于数据库有哪些实体我们也要分析。

在通过已有的系统的观察以及图书馆图书管理操作流程的启发下，分析出一个图书管理系统系统的应该具备以下的需求：

系统的登陆注册：首先作为一个完整的图书管理系统，当用户一进入系统时，自然面对的是我们的登录页面，针对管理员模块和用户模块的不同，也应该有两种登陆方式。用户进行登录的时候，如果是第一次登录，那么就要点击注册进入注册信息界面，在注册信息界面，完成个人相关信息的填写，注册成功之后，才可以登录系统。而管理员使用管理员的超级账号直接进行登录。

图书信息的查阅：读者和管理员进行登录以后都应该有这个模块的功能，只不过具体实现不同。其中最为基础的应该就是两者都能够对图书进行查询找到自己想要的图书的信息。而管理员车则应该有对图书信息的操作权限，对于图书信息可以进行添加和删除，也应该能够实现图书信息的修改。

图书类型的管理：管理员基于图书的类型应该能实现分类功能，这方便读者进行浏览，最好基于不同的分类能够实现数据的排行，在不同的类型的图书按照借阅的次数作为参考依据进行排行，作为读者进行图书借阅的参考。

读者信息的管理：读者可以对自己的个人信息进行管理，如修改自己的电话啊地址啊等等。而管理员则应该有对读者的信息进行增删改的权限，既可以实现对读者信息的增加，也可以实现对读者信息的删除修改。

系统管理：在这个模块里，无论是管理员还是用户都可以现实系统的退出，点击退出系统以后应该要能够实现返回到登录界面。而且还要实现个人账号密码的修改，无论是用户还是管理员都应该能够在这里通过修改密码对自己的密码进行修改然后将修改的结果返回到数据库当中。

3.2可行性分析

3.2.1技术可行性

基于MVC模式的图书管理系统的实现，有着非常多的编程语言可以完成，而我在本次的系统设计开发中选择了对于我自己而言比较熟练的语言Java语言，又在系统的结构上采用了B/S结构进行开发设计。Java语言是目前比较流行的语言，在实现这种管理系统的开发上尤为受欢迎，由于Java的开发历史较久，已经足够成熟了，在开发稳定性与系统的安全性上都能得到保证。前端页面主要采用的是HTML+CSS的设计配合JavaScript来进行动态页面的实现。再结合jQuery框架和BootStrap框架的栅格系统，设计界面十分的简便而且美观。BootStrap有着许多实用的组件，功能十分的完备足够用来设计一个美观的界面。而在后台的数据库采用的是MySQL数据库这种在web开发种也是应用的比较广泛的数据库，通过Navicat进行处理，可以实现数据库对于数据的有效管理。

基于MVC模式的图书管理系统采用MVC的设计规范，采用这种模式使设计模块化，三层结构分离项目，使耦合度大大降低，方便分工工作，使前端代码和后台代码能够分别编写，并且代码的复用更加的频繁，程序的延展性变得更好，维护起来十分方便。总体来说，对于开发人员的技术要求不是很严苛，熟悉了web开发的具体工作流程，开发出一个基础的图书管理系统还是能够实现的，因而从开发的技术上我们进行分析，我们可以得出能够实现的这么一个结论。

3.2.2操作可行性

本次基于MVC模式的图书管理系统，我们充分考虑到用户的使用体感，用户使用时第一直观感受就是来自页面，因此在前端的界面的UI设计上我们采取简约至上的原则，用户可以通过很直观的操作进行系统的使用，不同的人群，怕是计算机知识基础不够的人也能够通过简单的学习对图书管理系统进行快速的掌握，以实现自己的独立操作。而在进行系统的详细设计过程的时候，我在IDEA这个平台上进行开发，这是现今Java开发最常用的工具，这对我们的代码阶段的编写是十分有帮助的。而对于系统所需的数据库的安装工具XAMPP以及管理工具Navicat，个人已经多次使用，颇有些熟练度，而且本身这些软件就是为了在操作上简化对数据库的使用而进行开发的，并没有多大的难度。由此可见，基于MVC模式的图书管理系统系统不管是在前端UI的具体操作上还是在系统详细设计具体实现的的过程操作中，难度都是在可以接受的范围内的，从具体操作上来都是可行的。

3.2.3经济可行性

基于MVC模式所设计的图书管理系统，在进行开发的软件选择上，我选择的软件都是开放性的，不会收取费用。在后续的系统维护以及更新迭代的过程中，由于采用MVC模式，代码分离，逻辑都是十分的清晰明了，因此都仅仅需要少量的人力物力才力就可以完成。在开发系统的关键技术上，例如Bootstrap框架，J2EE技术体系，以及MySQL都是不需要费用的，因此在经济上是绝对可以进行这个项目的。

3.2.4社会可行性

进入二十一世纪以来，尤其是近些年，前往图书馆进行图书阅读已经成为大多数人会选择的优势，这得益于图书馆的环境优异，更多的是人们在现实生活中需要这种仪式感来慰藉自身得到精神文化需求，而图书馆针对络绎不绝的客流量，在面对图书的借阅时，原先的人工方式已经不能够满足如此庞大的需求，而现在的计算机技术的发展恰恰提供了网上管理的可能。现如今，各大高校以及大型公立的图书馆已经开始使用了网上管理模式。所以在已有的社会需求上，设计出这款图书管理系统。加上这款系统完全出自于自己独立的设计，不存在抄袭方面的文婷，用户也是可以按照自己的意愿自己选择使用的，所以社会层面这个图书管理系统完全有需求的市场，而且使用合理合法，因此是具有很强的社会可行性的。

首先分析了技术可行性和操作可行性，然后再经济可行性和社会可行性这几个方面的立体分析，综合这一系列的因素，于是可以得出一个合理的结论：这个项目是可行。

3.3系统流程分析

3.3.1系统开发流程

一次系统的开发流程设计对于我们在设计系统的时候可以提供很大的帮助，有了这么一次系统的分析指导，我们可以很好的分工对于不同的功能模块有针对性的进行研究。在按照顺序有条理的进行系统的开发，对于图书管理系统进行了需求分析以后我们很自然的就知道了系统所需要的功能模块，这些就是我们要具体进行实现的部分，然后根据开发流程制定的逻辑操作过程我们就可以整合各个模块，在代码中编写出各个功能页面的跳转逻辑，实现模块的功能整合，完成一个完整的系统。在系统开发的最后对图书管理系统的界面进行设计以及进行一系列可靠的系统测试。本次系统的的数据库是运用MySQL这个关系型数据库，我们对相应的数据进行了数据库数据表的创立，使用IDEA这个开发工具进行开发，各个模块的代码在具体的编译完成之后尽可能的满足功能需求。在系统测试中，保证系统是可以正常的进行图书管理功能的，整个页面的UI布局，不需要过于的精致花哨但是一点要做到简洁明了，原则上是力求各个类型的用户都尽可能快速的学会操作，使用该系统进行图书管理过的功能，具体开发流程如图3.1所示。

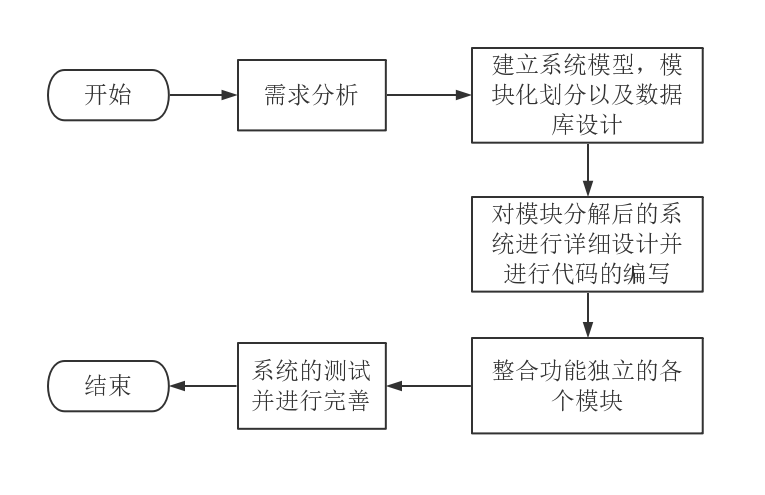


图3.1 系统开发流程图

3.3.2注册登录流程

具体的注册流程参见图3.2所示。

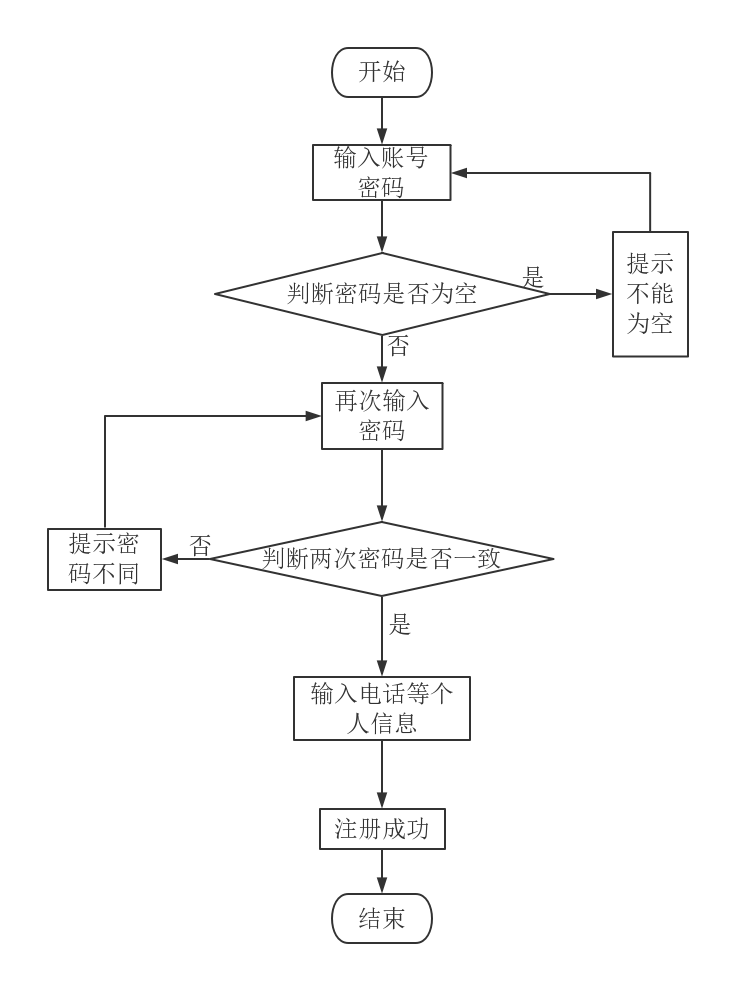


图3.2 注册流程图

用户要在系统中查看图书的相关信息以及进行图书的借阅工作，都必须先登陆到系统中。一个新的用户首次使用本图书管理系统时，要先进行用户账号的注册，只有注册了个人的账号才能够进行相关的操作。而用户进行个人账号的注册需要用户账号的名称、登录密码、姓名性别以及电话邮箱等信息，系统可以对用户输入的密码进行检测确认两次输入的密码一致才可以继续注册。用户输入完自己信息之后，确认信息无误就可以提交自己的信息，在完成了一次新账号的注册工作后，用户就可以使用刚刚注册的账号进行登录。而针对管理员，就不需要进行注册工作，因为账号信息在设计数据库的时候就像进行了预先设定的。管理员账号可以进行读者账号的添加。

用户通过登陆界面进行登录，首先要输入的账号和密码才能提交，如果没有输入信息就点击提交，由JavaScript嵌入的Java代码就会进行逻辑判断，返回不能为空的提示信息。具体登录时读者和管理员都一样，输入好的账户密码信息封装好传至数据库中与数据库中的数据信息对比，确认无误以后允许登录。具体的登录流程图如图3.3所示。

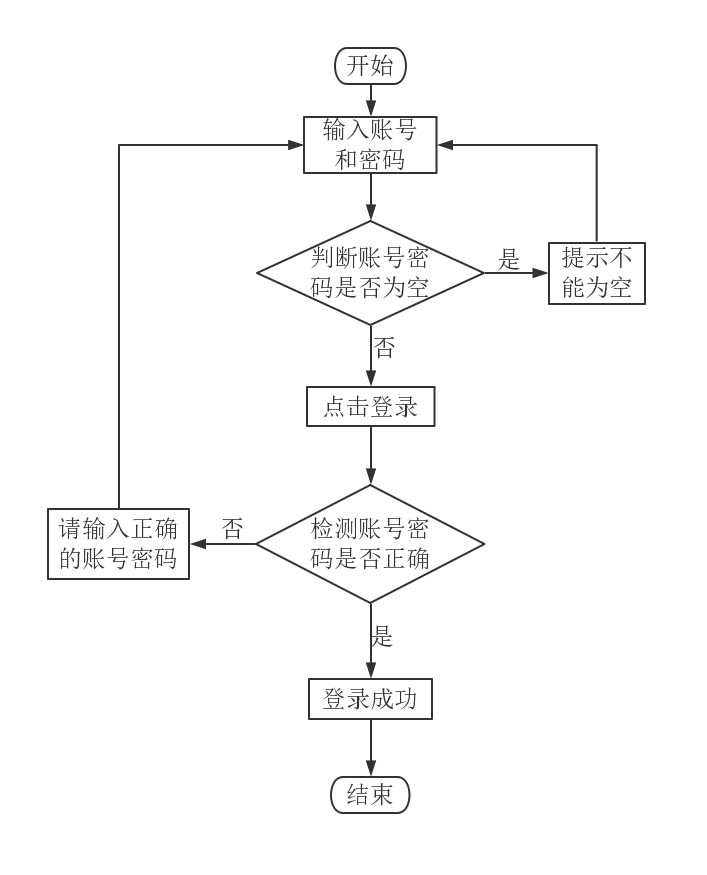


图3.3 登录流程图

3.3.3系统操作流程

用户包括管理员和读者，在输入完自己的账号和密码之后进行一次和数据库中的数据的对比验证。当输入的账号密码信息无误的时候就可以登录跳转到主页面。不过针对读者和管理员具体的功能有所差异，二者能够看到和进行操作的主页面也是有所不同的。系统操作的具体流程如图3.4所示。

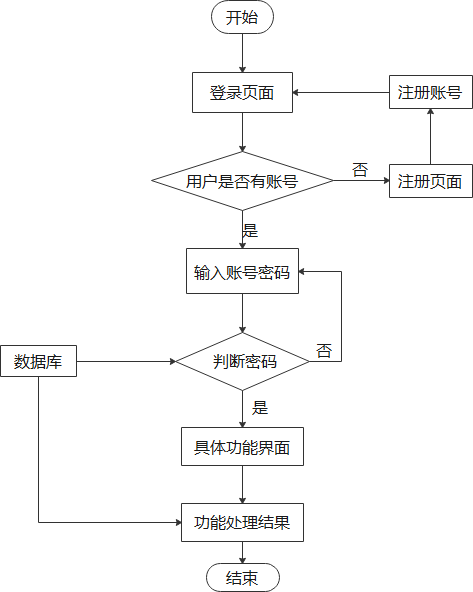


图3.4 系统操作流程图

图书借阅管理的主要能实现的功能有图书的借阅管理部分，还有还书的日期，以及图书可借数量的管理。图书借阅管理部分的功能是这样具体来实现的，在读者借书的时候，添加这个读者的借阅信息，也要生成借阅图书的历史记录。还书的功能就包括还书的日期以及还书信息的记录，还要对相关借阅信息进行删除，在临近还书日期的时候，最好还要有提醒用户还书的功能。

如图所示，管理员登录的时候，直接输入自己的管理员账号和后台数据库的管理员账号信息进行对比，如果一致，则提示登陆成功，如果不一致则提示密码错误。

但管理员使用自己的管理员账号登陆成功的时候，则进入了管理员的主页面，在这个页面也同样有着几个子页面可供管理员选择，不同的子页面对应着管理员权限能够操作的几个功能。首先管理员可以通过点击读者管理界面，查看目前注册了多少个读者，还可以查看读者的个人信息，如读者的电话，读者的邮箱之类的，最主要的可以看到读者借阅图书的情况，读者还有多少书没有还，读者还可以借多少本图书，还可以对读者的账号进行删除。当管理员点击图书管理界面的时候可以看到图书的关键信息，比如图书的类别，图书的出版社，图书的库中还剩数目。对于图书的这些信息，管理员还可以进行修改，也可以删除图书的信息。当管理员点击图书的借阅记录的时候就可以看到哪些图书被借阅，以及借阅图书的读者是哪一个。当然管理员也可以通过点击修改密码来修改自己的密码。

书籍管理的模块要实现每一本书籍的类别，出版社以及图书馆现存数量和可借数量都要有显示，书籍信息可以进行修改，可以增加书籍或者删除书籍。读者的信息管理要实现有管理员权限的用户可以对读者的个人信息删除，而普通的用户则可以实现对自己的姓名电话邮箱等个人信息进行修改。

读者登录的时候，没有账号的读者可以进行个人信息注册，输入自己的姓名，还有密码，然后再输入一遍自己的密码进行确认，只有两次的输入一致才能继续，再输入自己个人电话和邮箱的地址，点击提交就可以注册一个新的账号。

当读者有了自己账号，就可以进行登录了，读者输入自己的账号信息，通过了数据库的验证后就可以跳转进入读者的主界面，如果验证不一致就提示密码错误。

当读者登陆成功，进入读者的主界面时候，可以看到有几个子页面可供读者选择，分别对应着读者权限能够操作的几个功能，首先读者可以点击进去自己的个人信息管理界面，在这个界面读者可以对自己的电话，邮箱等个人信息进行修改。再可以通过点击图书信息界面，可以浏览图书的名字，于是的出版社，图书的作者等图书信息，也可以看见当前图书的可供借阅的数量，找到自己要借阅的图书，读者可以点击借阅。

3.3.4信息管理流程

理员能够进行读者信息的管理修改以及删除读者的账号信息，也可以对图书信息的管理，通过图书名称查询图书信息然后进行图书信息的删除或者修改。读者在读者的界面可以选择个人信息的管理，可以进行个人信息的修改，图书信息模块可以查询图书信息，也可以对图书信息进行借阅，在还书板块可以对图书进行借阅，在借阅历史板块可以对自己借阅得图书进行归还。不管是管理员还是用读者都可以点击修改密码对自己的账号密码进行修改，使用完毕以后可以通过点击退出进行系统的退出从而跳转到登录页面。

管理员可以对图书以及读者的信息进行相关的信息添加修改以及删除一系列的操作进行信息的管理，具体的流程如图3.5所示。

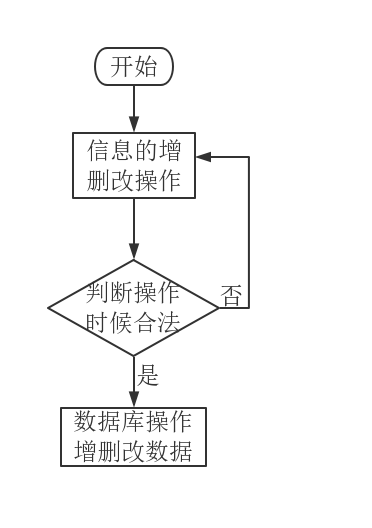


图3.5 信息管理流程图

3.4系统用例分析

3.4.1管理员用例图

管理员使用自己在后台数据库管理系统中预先设定好的管理员的账号密码进行登录就可以跳转到管理员的主页面，在这个页面对应着管理员的权限可以进行操作的几个模块功能。这些功能也就一一对应显示在我们的管理员用例图上，管理员用例参见图3.6所示。

管理员登陆成功之后，则进入了管理员的主页面，在这个页面也同样有着几个子页面供管理员操作，不同的子页面对应着管理员权限能够操作的几个功能。首先管理员可以通过点击读者管理界面，查看目前注册了多少个读者，还可以查看读者的个人信息，如读者的电话，读者的邮箱之类的，最主要的可以看到读者借阅图书的情况，读者还有多少书没有还，读者还可以借多少本图书，还可以对读者的账号进行删除。当管理员点击图书管理界面的时候可以看到图书的关键信息，比如图书的类别，图书的出版社，图书的库中还剩数目。对于图书的这些信息，管理员还可以进行修改，也可以删除图书的信息。当管理员点击图书的借阅记录的时候就可以看到哪些图书被借阅，以及借阅图书的读者是哪一个。管理员也可以通过点击修改密码来修改自己的密码。作为管理员在使用这个基于MVC模式的图书管理系统的时候，首先直接输入自己的管理员账号进行数据库的验证工作，验证完毕后如果信息一致，则提示登陆成功，如果不一致则提示密码错误。具体如图3.6所示。

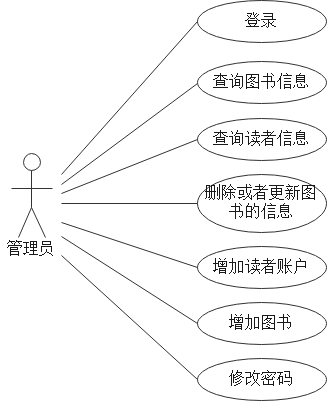


图3.6 管理员用例图

3.4.2读者用例图

读者首先通过注册进行读者账号的注册，注册成功之后返回登录页面通过自己的账号密码进行登录，登陆成功以后可以看到读者的主页面，在这个页面对应着读者的权限可以进行操作的几个模块功能。这些功能也就一一对应显示在我们的读者用例图上，如图3.7所示。

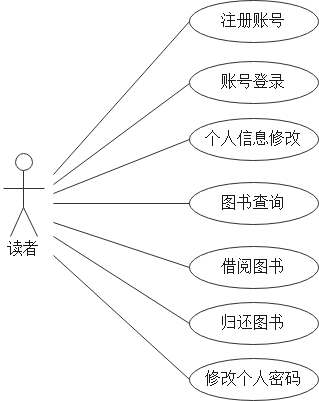


图3.7 读者用例图

3.5业务流程分析

对于各个功能模块进行了分析以后，我们就需要对图书管理系统进行全局的模块功能整合，通过各个模块进行的业务处理的流程，我们在具体的各个功能之间对它们具体是如何联系操作，数据是如何流动的进行分析，我们可以从系统的整体上来分析系统的运行流程，具体流程如图3.8所示。

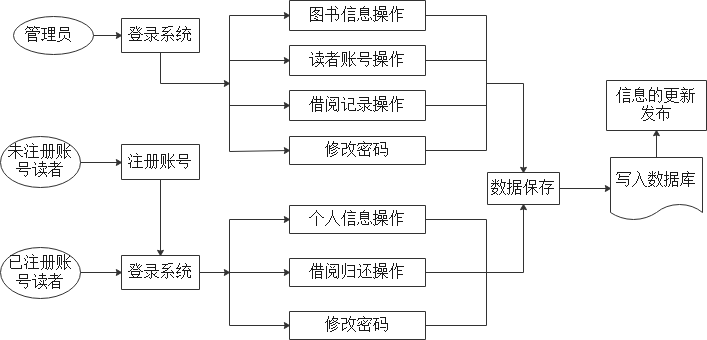


图3.8 业务流程图

3.6 数据库分析

对于这个基于MVC模式的图书管理系统来说，数据库的连接是十分重要的，因为对一个有现实意义的图书管理系统，只有配置上数据库，通过数据库来进行数据的存储数据的查询和比对，这样的图书管理系统才能够起到图书管理的作用。因此，一个图书管里系统的成功与否，与这个系统的数据库的好坏有着不可分割的关系。当你的设计库设计的十分优秀的时候，很显然你的系统的性能与可操作性自然就上升一个档次，要是数据库设计的十分冗余，毫无疑问，这样的系统一定是性能差劲的。所以要想使系统的数据查询，数据存储以及数据修改的效率变高，系统的性能提升，数据库的表单之间的一定要简单清晰，数据结构一定要一目了然。

当然要想设计一个好的数据库，那我就要对数据库中的数据做出细致的解释说明，因此我们就需要有一个数据字典。如果有了数据字典，我们就可以进行数据的有效管理，这样的话，即使在以后数据库要进行调整，也可以进行对应的修改，那我们进行数据的共享的时候，也会有很大的增益。所以我们针对图书管理系统的功能需求，对系统所需要使用到的数据项进行了详细的分析，制作了许多的数据表，将这些数据项作为其中的实体，如此操作一个完整的数据流程就跃然纸上。这样有条理的科学的设计出数据库的各个表单内容，我们系统在数据库的使用上会提高效率。

据数据库设计的规范设计出所需要的各个表，将数据分布到各个对应的表中。要注意的是，在遵循了设计规范的同时我们也要注意到减少数据的重复，各个表的互相关联性要好。而在实现数据库的概念设计的时候，根据实体及其所拥有的属性进而设计E-R模型绘制成图是最好的方式。

1. 图书信息具体拥有的属性有：图书编号、图书名称、出版日期、出版社、作者名称、图书类别，其中图书名称是关键的属性，通过图书名称我们可以进行图书的查询快速定位图书。通过其之间的对应关系我们可以设计出读者的E-R实体属性图，详情参见图3.9所示。

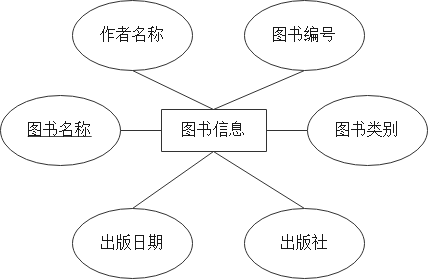


图3.10 图书信息实体图

1. 用户作为图书管理系统的主要使用者，也是一个主体，这个主题具有着用户编号、姓名、性别、账号、密码、联系电话、邮箱地址以及可借阅量这八个属性，根据读者这个用户个体与其属性进行一一对应，可以设计出读者的E-R实体属性图，详情见图3.10所示。

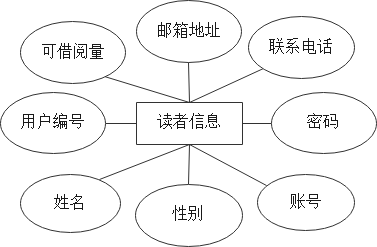


图3.10 读者信息实体图

1. 读者进行图书的借阅，但读者选中了一本图书并且点击借阅就会产生一个借阅的订单，这个订单也是一个实体，其具有这么八个属性，订单编号、借书人名称、借书人编号、借阅图书名称、借阅图书编号、借书日期、截止日期、图书库存数量。根据借阅订单这个实体体与其属性进行一一对应，可以设计出借阅订单的E-R实体属性图，详情见图3.11所示

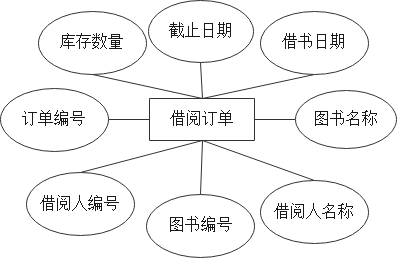


图3.11 借阅订单实体图

1. 图书类型也是一个实体，用于存储图书的类型管理员在修改图书的类型的时候，数据应该保存在这里。图书类型具有两个属性，图书的类型名称以及类型编号。根据这个实体及其两个属性进行一一对应，我们可以设计出E-R实体属性图，详情见图3.12所示



图3.12 借阅订单实体图

通过对于这几个实体之间的分析，显然我们可以得出这么几个结论。图书信息和借阅订单以及读者实体间存在着一定的关系，其中一个借阅订单中只有一本图书的信息也只能有一个读者的信息，但是一个读者能同时有很多的借阅订单，一本图书却只能有一个借阅订单，因此三者之间有着以下关系，读者和订单存在着一对N的关系，而借阅订单和图书信息却是一对一的关系。至于图书类型和图书信息明显具有一对N的关系，因为一本图书只有一个类型，而一个类型却有多本图书对应。基于以上的关系分析，我们可以进行一个总结，然后从总体的角度出发设计出图书管理系统的系统总体E-R图，详情见图3.13。

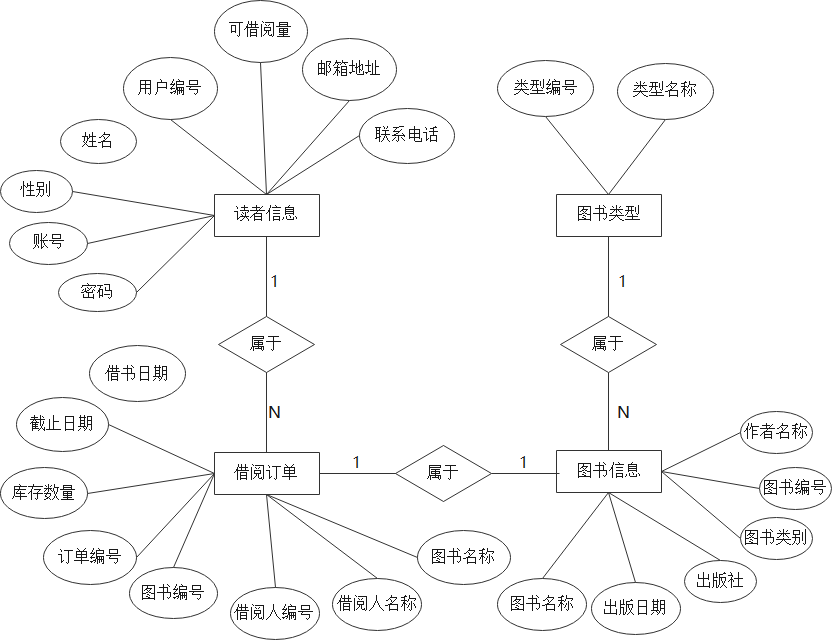


图3.13 系统总体E-R图

第四章 详细设计

进行了具体功能实现上的需求分析以及从各个角度综合进行的可行性分析以后我们又对系统的具体开发的流程包括系统的用例以及数据库的相关设计工作进行了一个详细准确的介绍。在接下来的这一章节，我们就会针对系统的详细设计做一个细致的介绍，具体上就是对于各个模块的代码设计等等，我们期望系统能够达到在这个完整的详细设计的预期效果，能够做到图书管理的操作，并且能够成功的实现运用数据库的操作进行数据的有效管理。

4.1 系统概述

基于MVC模式的图书管理系统，是一个完备的图书管理系统，在现实应用种功能之间应该具有严密的逻辑，顺畅的实现各个功能。当用户打开系统是输入自己的账号和密码，就要配合数据库检查密码和账号的正确与否，通过验证就会登录系统，实现其他的功能，否则就会提示密码或者账号存在问题重新输入。没有账号的用户通过注册功能进行账号的注册，再进行登录，登陆以后就可以进入主界面，然后用户可以选择跳转到图书信息管理界面进行图书信息的浏览选择图书进行借阅。当管理员使用管理者的账号进行登录的时候也是通过后台数据库进行验证，密码错误就提示密码不对，成功登陆以后也跳转到主界面但是功能更加丰富，首先有一个读者的管理功能，点击进入读者管理界面。可以对读者的信息进行删除或者修改。然后点击图书信息的管理功能，就可以进入图书信息的管理界面，可以修改图书的类别，也可以修改图书的信息，也可以删除图书相关信息。

4.2 系统功能设计

首先作为一个完整的图书管理系统，当用户一进入系统时，自然面对的是我们的登录页面，针对管理员模块和用户模块的不同，也应该有两种登陆方式。

1、读者模块，没有账号的读者可以进行个人信息注册，输入自己的姓名，还有密码，然后再输入一遍自己的密码进行确认，只有两次的输入一致才能继续，再输入自己个人电话和邮箱的地址，点击提交就可以注册一个新的账号。

当读者有了自己账号以及密码的时候，输入自己账号和密码，通过和数据库的内保存的账号密码进行对比，如果一致就提示欢迎登录，然后跳转到读者的主界面，如果不一致就提示密码错误。

当读者登陆成功，进入读者的主界面时候，可以看到有几个子页面可供读者选择，分别对应着读者权限能够操作的几个功能，首先读者可以点击进去自己的个人信息管理界面，在这个界面读者可以对自己的电话，邮箱等个人信息进行修改。再可以通过点击图书信息界面，可以浏览图书的名字，于是的出版社，图书的作者等图书信息，也可以看见当前图书的可供借阅的数量，找到自己要借阅的图书，读者可以点击借阅。

2、管理员模块，直接输入自己的管理员账号和后台数据库的管理员账号信息进行对比，如果一致，则提示登陆成功，如果不一致则提示密码错误。

但管理员使用自己的管理员账号登陆成功的时候，则进入了管理员的主页面，在这个页面也同样有着几个子页面可供管理员选择，不同的子页面对应着管理员权限能够操作的几个功能。首先管理员可以通过点击读者管理界面，查看目前注册了多少个读者，还可以查看读者的个人信息，如读者的电话，读者的邮箱之类的，最主要的可以看到读者借阅图书的情况，读者还有多少书没有还，读者还可以借多少本图书，还可以对读者的账号进行删除。具体的系统功能模块图如图4.1所示。

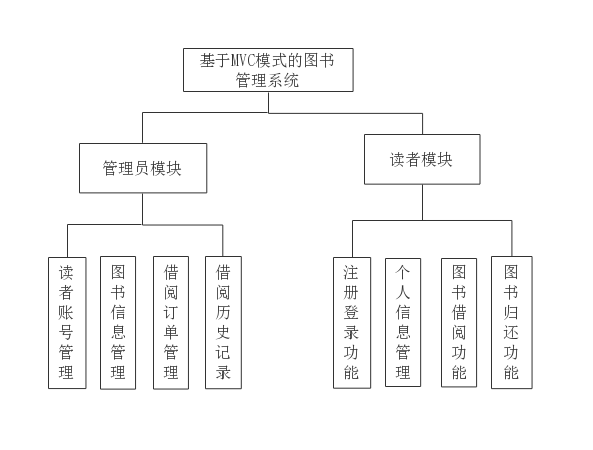


图4.1 系统功能模块图

4.3 数据库设计

对于一个图书管理系统而言，数据库是最为重要的一环，没有数据进行数据的存储和提供一系列的操作，那么这个图书管理系统这能是一个摆设。因此，一个图书管里系统的成功与否，与这个系统的数据库的好坏当然有着不可分割的关系。当你的设计库设计的十分优秀的时候，你的系统的性能和可操作性肯定会上升一个档次，要是数据库设计的十分冗余，这样一定会导致系统性能差劲。所以要想使系统的数据查询，数据存储以及数据修改的效率变高，系统的性能提升，数据库的表单之间的一定要简单清晰，数据结构一定要一目了然。因此我们需要设计出一个十分准确的数据库，在这个图书管理系统数据库中我们要避免产生设计上的错误，这样系统才会拥有高性能。

于MVC模式的图书管理系统使用的是MySQL关系型数据库管理系统，使用的Navicat这一款数据库管理工具进行数据库的管理操作，这个工具是专门为了对数据库的操作难度进行降低以及减少系统管理成本而研发设计的，这个工具极大的方便用户对于数据库的管理。接下来通过细致的表单信息来介绍一下数据库中的表单应该具体设计哪些内容。

1. 图书信息表命名为book，这个表是图书信息表中所具有的各个属性的具体类型以及长度等相关信息的设定，其中各个字段的详情参考表4-1所示。

表4-1 图书信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 长度 | 主键 | 允许空 | 说明 |
| bid | big int | 20 | 是 | 否 | 书籍编号 |
| name | varchar | 255 | 否 | 否 | 书名 |
| card | Int | 11 | 否 | 否 | 序列号 |
| autho | varchar | 255 | 否 | 是 | 作者 |
| num | varchar | 255 | 否 | 否 | 库存数量 |
| press | varchar | 255 | 否 | 是 | 出版社 |
| type | varchar | 255 | 否 | 否 | 书籍类型 |

2、读者账号的信息表命名为reader，这个表单主要是对读者账号的各种信息进行存储，各个字段参见表4-2所示。

表4-2 读者账号信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 长度 | 主键 | 允许空 | 说明 |
| rid | int | 11 | 是 | 否 | 主键 |
| username | varchar | 255 | 否 | 否 | 用户账号名 |
| name | varchar | 255 | 否 | 否 | 用户名 |
| password | varchar | 255 | 否 | 否 | 密码 |
| sex | varchar | 255 | 否 | 是 | 性别 |
| phone | varchar | 255 | 否 | 是 | 电话 |
| email | varchar | 255 | 否 | 是 | 邮箱地址 |
| Led-num | int | 11 | 否 | 是 | 可借数量 |

3、用户订单信息表命名为booktype，这个表单主要是存储书籍的不同的类型，各个字段参见表4-3所示。

表4-3 订单信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 长度 | 主键 | 允许空 | 说明 |
| tid | tinyint | 4 | 是 | 否 | 主键 |
| name | varchar | 255 | 否 | 否 | 留言日期 |

3、用户订单信息表命名为history，这个表单主要是存储借阅书籍订单的记录，各个字段参见表4-3所示。

表4-4 借书订单信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 长度 | 主键 | 允许空 | 说明 |
| hid | tinyint | 4 | 是 | 否 | 主键 |
| rid | tinyint | 4 | 否 | 否 | 读者编号 |
| bid | tinyint | 4 | 否 | 否 | 书籍编号 |
| card | Int | 11 | 否 | 否 | 序列号 |
| bookname | varchar | 255 | 否 | 否 | 书籍名称 |
| readername | varchar | 255 | 否 | 否 | 读者名称 |
| username | varchar | 255 | 否 | 否 | 读者账号 |
| begintime | varchar | 255 | 否 | 否 | 开始日期 |
| endtime | varchar | 255 | 否 | 否 | 截止日期 |

第五章 功能模块实现

在对系统的详细设计进行介绍之后，我们对这个系统的具体实现已经了如指掌了，接下来就是要实现这个系统的运行。本章我们主要是基于MVC模式的图书管理系统的具体系统实现。通过读者功能模块和管理员功能模块两个部分来展开对具体实现的介绍。

5.1读者模块

5.1.1 注册登录功能实现

读者打开这个系统时候会看到一个登录表单。如果读者没有自己的账号密码，那么就可以点击注册按钮，进入如图5.1所示的注册页面进行账号的注册。



图5.1 用户注册界面

在这个注册页面，用户需要填写自己的个人信息，再点击提交就可以在数据库中存入你的个人账号了，然后就可以跳转到登录界面，如图5.2所示。



图5.2 用户登录界面

5.1.2 读者个人信息管理模块实现

用户注册登陆完成之后，就可以进入读者的主页面，通过点击读者的个人信息管理，就可以进入个人信息管理页面，如图5.3所示。

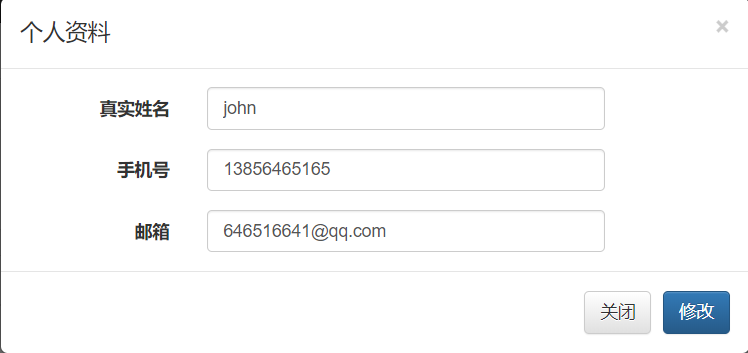


图5.3 个人信息修改界面

实现对个人信息的修改是读者的项很重要的，读者在使用这个系统的时候，有可能会碰到自己的身份信息发生改变。这个时候就要对自己保存在图书管理系统中的信息进行更新，这个有助于进行图书系统管理工作的正常运行。读者对自己的信息进行了一定的修改后，点击修改这一个按钮，那么读者所有更新的信息都会保存到数据库中去。

5.1.3 图书借阅模块

用户点击图书信息即可进入图书借阅模块界面，如图5.4所示。

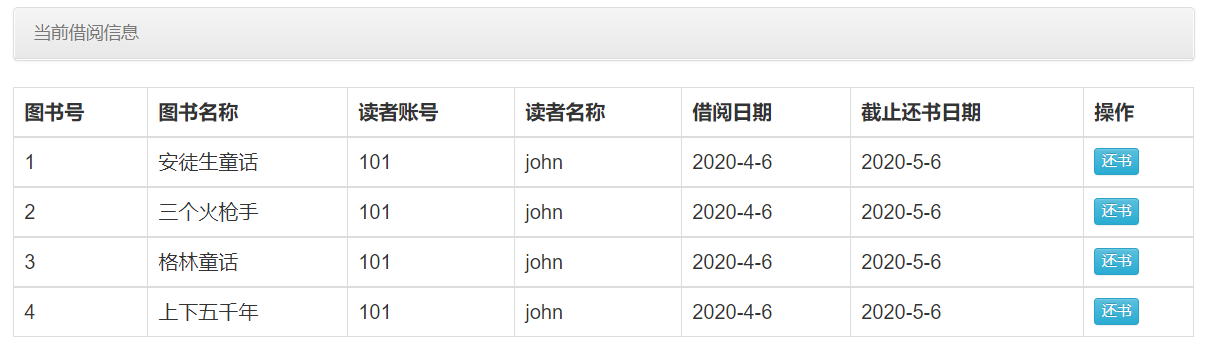


图5.4 图书借阅界面

在这个界面用户首先可以通过搜索框输入图书的名称来进行图书的搜索，如果数据库中有相应的的图书的信息，就会显示出来图书号，图书的类型和图书名称，作者名称出版社等信息，还有图书在库中的总数量也会显示出来，提示你还有多少本能够是借阅，读者想借阅的操作也是十分的简单，仅仅点击借阅就可以完成借阅的工作。

5.1.4 图书归还模块

用户点击图书借阅记录即可进入图书借阅信息模块界面，如图5.5所示。在这个界面，可以查看自己借阅的书籍，以及相关信息，最主要可以看见自己截至还书日期，点击最后的还书操作就可以进行还书。

图5.5 图借阅信息界面

在这个界面读者可以看到自己借阅的图书的相关信息，比如图书号图书名称等信息，还可以看到自己的每本书的借书日期，以及截止日期，然后找到自己想要归还的图书点击还书，就轻松的完成图书归还的工作

5.1.5 修改密码功能

读者通过点击密码修改，弹出密码修改的弹窗界面详情如图5.6所示，在这个界面读者可以通过输入自己的原密码和新密码，点击修改就可以完成自己密码修改工作。



图5.6 密码修改界面

5.2管理员模块

5.2.1 管理员主页面

管理员用户通过登陆界面进行登录，首先要输入账号和密码才能提交，如果没有输入就点击提交，相关的代码就会进行判断，返回不能为空的提示信息。管理员输入账号密码后，与数据库中的一致，就会跳转到管理员的主页面。具体的页面如同图5.7所示。

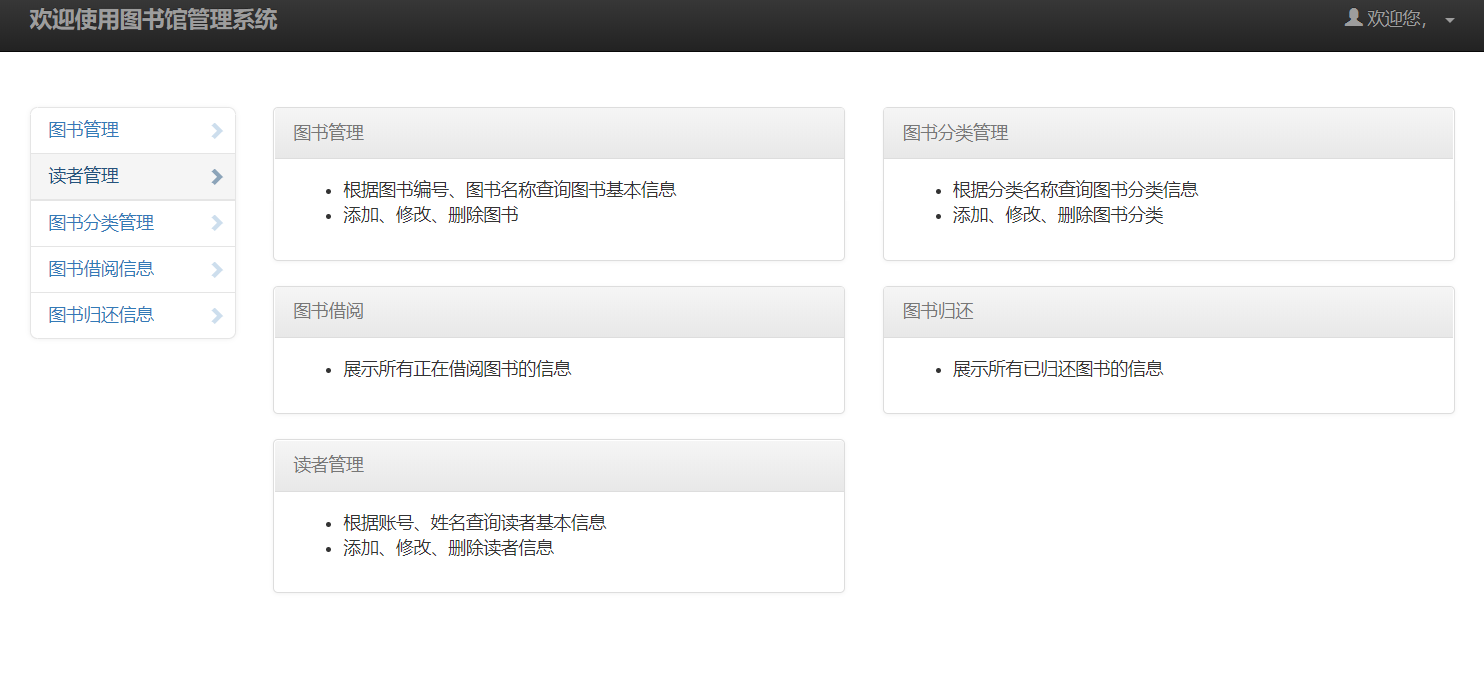


图5.7管理主界面

在管理员的主页面有着很多的子页面，通过点击跳转到不同的页面实现不同的功能。管理员实现对于图书信息的管理可以通过点击图书管理页面，在这个页面实现对图书信息的增删改查等功能，要是需要对读者的信息进行管理，管理员可以点击读者管理进入读者管理界面，在这个界面对读者的信息进行增删操作。同样的对于借阅的历史订单相关的比如借阅信息和归还信息进行管理，也可以通过点击进入相关的界面实现具体的操作。

5.2.2 读者信息管理

管理员登录以后进入管理员主界面，在这个界面管理员通过点击读者管理进入读者管理界面。在这个界面具体如图5.8所示。



图5.8读者管理界面

在这个界面，管理员可以进行一些对于读者的信息进行管理的相关操作，通过点击添加读者就可以通过对读者的信息进行填写然后就会生成一个读者的账号信息，要是想修改读者的信息就可以通过点击修改然后就会有修改信息的弹窗弹出，管理员就可以进行修改，还可以通过删除，对读者的账号信息进行删除。

5.2.3 读者增加功能

管理员通过对读者管理界面的添加读者的按钮进行点击就会弹出添加读者的页面，详情页面如图5.9所示。在这个页面管理员就可以对读者的信息进行添加操作。管理员需要先输入用户的账号姓名邮箱手机号等个人信息。然后还需要对用户的密码进行设置，这个就是用户的初始密码，在后期用户进行登陆的以后可以进行修改。最后就是管理员对于用户的可借阅天数以及最大可借阅数量的填写。这些信息都影响到读者以后的借阅权限。



图5.9读者添加界面

5.2.4 图书信息管理

管理员通过点击图书信息管理进入图书管理界面如图5.10所示.



图5.10 图书管理界面

在这个页面管理员可以进行图书的管理操作，具体的操作是当管理员想增加图书的信息的时候，点击添加图书就可以弹出增加图书的弹窗，在这个窗口输入图书的类型图书的名称以及作者的名称还有图书的出版社这些关键信息，就可以实现图书的增加。对于已经增加成功入库的图书信息，管理员也可以进行查看，也可以通过搜索框进行图书名称的搜索定位到目标图书，然后可以进行修改和删除，想要修改的时候就点击修改然后就弹出图书信息的修改框，输入修改后的图书信息就可以进行图书的修改，修改完毕后点击确认即可将修改的新信息进行入库保存，当你想删除图书的时候就找到目标图书点击删除就你可以进行图书的删除。这些操作都是十分简单容易操作，十分方便管理人员进行图书的管理。

5.2.5 图书添加功能

管理员在进入图书管理的界面后通过点击添加图书就可以进入图书添加的页面进行操作，具体的页面参见图5.11所示。

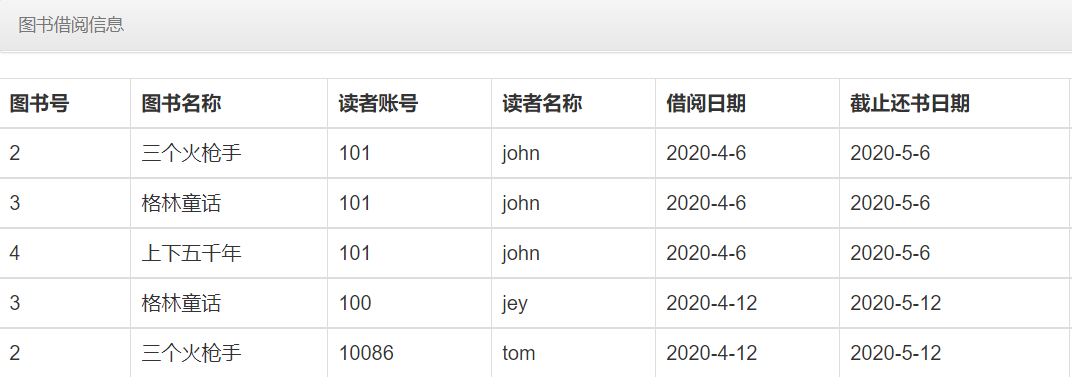


图5.11图书添加界面

在这个页面我们可以看见这么几个图书的关键信息，首先就是图书号，最为图书的唯一标识我们需要通过图书号码来进行图书的准确定位。所以我们先输入图书的号码。再就是图书的名称，我们这个信息也是必须要填的，这个作为读者进行图书搜索时候的标识。再就是输入图书的类型，这个通过下拉框进行填写。最后就是作者名称出版社以及总数量这些图书的属性了。

5.2.6 借阅信息管理

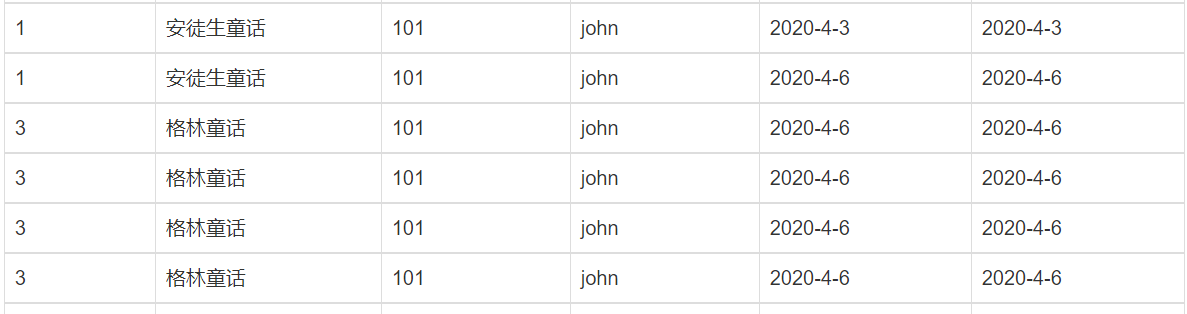
管理员点击借阅历史就可以进入如图5.12所示的借阅历史界面，在这个界面管理员可以查看所有读者的历史借阅记录方便对借阅历史信息的掌握。

图5.12 借阅信息界面

在这个界面，管理员虽然无法进行操作，但是却是一个十分重要的界面。在这个界面管理员可以查看用户借阅信息。比如哪一个用户借阅了，他的账号和名称是什么，他进行借阅的又是什么图书。什么时候进行的借阅，他的借阅的截至日期又是什么时候。这些消息都有利于管理员进行图书的管理，也方便对于读者进行借阅的催还。

5.2.7 归还信息管理

管理员在进入归还信息界面后将会看到如图5.13所示的图书归还信息记录页面。

图5.13 归还信息界面

在这个页面管理员可以清晰的看到，过往的历史借阅记录。比如什么人进行了借阅活动。他借过什么书，借书的日期是什么时候，过了多久还的书这些必要的历史信息均可以保存在数据库里，并且显示在这个页面，以供管理员查询。这些记录都可以为管理员的管理提供有效的保证，比如图书损坏了你就可以知道这是谁借阅的图书，进行赔偿。

5.2.8 图书分类管理

在管理员进入图书分类管理板块后，可以看到如图5.14所示的图书分类管理界面，在这个界面，管理员可以对图书的分类实行有效的管理。他可以通过添加分类的按钮进行分类的添加，所谓的图书分类的添加就是对图书进行一个基于类型上的划分，比如小说故事历史等种类的划分.



图5.14图书分类管理界面

具体的的图书类型的添加页面如图5.15，在这个页面管理员可以进行图书具体的新分类进行系统添加。



图5.15图书分类添加界面

可以看到这两个页面管理员可以进行操作的的功能，管理员在图书分类管理页面可以对图书的分类进行添加修改以及不合适的分类是实现删除这些具体的操作，具体的分类添加的操作，当我们点击添加分类的按钮时会弹出图5.15所示的弹窗，在这个页面我们仅仅需要输入一个信息，那就是分类的名称，点击添加就可以实现图书分类的添加。对于图书分类这个功能，虽然内容并不是很多，却是一个很好用很实用的功能。在平时我们对于图书的管理时，通过具体的分类来管理，这样的管理效率无疑会提升很多，而对管理效率的提升正是我们开发这个图书管理系统的目的之一。对于读者而言实现分类查询，这样也十分的方便于他们找寻图书。

# 第六章 系统测试

在进行完上述所有的步骤之后，基于MVC模式的图书管理系统基本已经开发完毕，但是为了确保系统的稳定性，在这次开发中最不可获取的一环就是要对我们开发的这个图书管理系统进行运行测试，基于各个模块功能，做有针对性的功能测试，只有完成了测试工作，我们才能宣布这个系统真正完工。因此在这一章，针对各个功能进行测试，并且得出相应结论。

6.1测试意义

在我们平时的开发过程中，到了最后一步，万万不能省略的局势对设计完成的系统进行测试，主要是测试预定的功能，要确保设计的各个功能真正能够实现，通过测试我们开发出来的系统，可以找到其中可能存在的bug进行解决，这也能很大程度上可以减少，系统在投入使用后，给用户带来困扰，而开发者要反复的进行补救工作。因此为了开发出一个运行良好，性能有具体保障的系统，在开发完毕之初，我们很有必要性进行多次的准确的测试工作。

6.2 测试环境

本次针对这个基于MVC模式的图书管理系统我们进行了多次的测试，在具体的测试中，我们可以针对性具体的功能模块进行针对性的功能测试，确保我们的系统能够可靠的运行。我们在运行的客户端浏览器的具体选择上依旧选择的是最常使用功能最为熟悉谷歌浏览器，其余的运行系统的电脑的详细硬件配置信息参见表6-1所示。

表6-1 硬件配置

|  |  |
| --- | --- |
| 软硬件环境 | |
| 处理器 | Intel（R）Core（TM）i7-5500CPU |
| 内存 | 8GB |
| 硬盘 | 500GB |
| 运行网速 | 100M宽带 |
| 操作系统 | Microsoft Windows 10 |
| 服务器 | Tomcat8.5 |
| 数据库软件 | Navicat for MySQL |
| 运行平台 | IntelliJ IDEA 2019.3.3 x64 |

6.3 测试结果

6.3.1 注册功能测试

一个新读者在第一次使用我们这个基于MVC模式的图书管理系统的时候，没有账号的他必须要从我们的注册账号这个功能开始。所以在这一次的测试中我们也从我们的注册功能开始进行我们整个系统的具体的功能模块测试。

读者打开这个系统时候会看到一个登录表单。如果读者没有自己的账号密码，那么就可以点击注册按钮进入注册页面。在这个注册页面，用户首先要填写自己的账号，然后输入密码，这是关键的一步当你的密码没有输入的时候，会进行提示读者输入密码，然后就是输入自己的真实姓名，最后就是输入自己的邮箱和手机号码，点击提交就可以在数据库中存入你的个人账号了，而返回登录的按钮是给误点注册按钮的用户可以直接进行跳转到你的登录页面。

这次我们只要进行测试的具体操作是：

1. 当我们没有输入账号和密码的时候点击提交时候会不会提交成功。
2. 当我们输入的密码长度不符合我们的要求时候，会不会有提示我们密码不规范。

经过多次的账号注册测试，我们发现上述提到的问题均未发生，我们能够确定注册功能的确能够实现读者账号的注册并且注册读者账号信息会保存到图书管理系统的数据库中。

6.3.2 登录功能测试

在对我们系统的用户注册功能进行了多次的完整测试以后，我们开始注意到用户登录的第二步，就是登录功能的具体测试，这次的测试我们依旧进行多次测试，看看能不能避免我们预期估计的可能出现的错误，或者一些意想不到的错误。主要的测试要求如下所示：

1. 如果输入的账号不存在于数据库中，系统能否提示用户不存在。
2. 如果输入的账号密码于数据库中已经存在的账号的密码不是相匹配的，系统能否提示密码错误。

可以预想到的是，在进行了多次完整的测试之后，这些预想到可能出现的错误都没有出现，也没有没法其他的意料之外的错误，因此我们可以很肯定的说经过了规范的详细测试之后，我们的基于MVC模式的图书管理系统的登录功能是可靠的，能够实现用户登录

6.3.3 读者借书功能测试

在进行完用户的注册登录两项功能以后，读者也能够成功的进入读者的主页面，作为一个图书管理系统最为关键的功能无疑就是，读者能够进行图书的借阅操作，为了确保这个操作能够实现，而且在实现的过程中稳定不会出现错误我们也对这个功能进行了多次的测试，在进行了具体的测试过程中我们要确保能够做到以下几点：

1. 用户在主页面点击借书页面能够实现页面的跳转，成功进入图书信息界面。
2. 用户在图书借阅界面能够实现图书的查询功能
3. 用户在查询到自己想找的具体图书之后点击借阅的却能够显示借阅成功，而且后台的数据库也显示借阅成功，生成借阅的具体订单并且保存在数据库中。

在经过多次仔细的测试之后，我们发现，基本的流程都是能够实现的，图书借阅的操作的却能够进行。

6.3.4 个人信息管理功能测试

在完成了我们的借阅功能的具体测试之后，针对读者的个人信息的管理功能我们也要进行测试。在这次关于个人信息管理功能的测试中我们主要对以下几点进行测试：

1. 我们在读者主页面点击读者的个人信息页面能够实现跳转
2. 我们在读者的个人信息页面点击修改能够弹出修改个人信息的弹窗
3. 我们修改完我们的个人信息之后，数据能够提交保存到数据库中，我们在数据库中能看到实时更新的数据变动。

经过了多次的严密测试，我们均未发现有故障或者问题出现，因此在个别人信息管理功能上是没有问题的。

6.3.4 管理员功能测试

在之前我们主要是你针对读者的功能模块进行测试，这一系列的测试均通过了，读者使用系统的相关功能没有出现问题，接下来我们一口气对管理员功能进行完整的测试，同样的这项测试也有以下的要求：

1. 管理员使用管理员的特殊账号能够实现登录
2. 管理员登录以后能够跳转到具体的管理员登录页面，在这个主页面，管理员能够通过点击不同的页面从而都能实现跳转
3. 管理员在跳转到读者信息管理界面后，对读者能够实现增删改查，而且数据能够保存到数据库中。
4. 管理员在跳转到图书信息管理界面后，对图书能够实现增删改查，而且数据能够保存到数据库中。
5. 管理员通过点击能够跳转到借阅历史界面，能够看到读者借阅图书的历史记录。

通过进行多次的上述过程的测试我们确定管理员能够正常操作系统。

6.3.5 数据库测试

在之前的测试中我们进行的数据操作，所有的数据改变，包括信息的删除，添加以及修改的操作都能够正确的反映到数据库当中去。这次对数据库的专项的测试时，我们准备从整个系统流程的方向出发，专门对数据库进行一次针对测试。结果发现，整个流程按照顺序走下来，在前端修改的数据都能够在后台数据库中得到改变。

因此我们的得出了数据库也是十分可靠的结论。

6.3.6 测试结果

在进行完所有的的测试步骤之后，我们通过所有测试用例都能够顺利完成得出了结论，我们在进行基于MVC模式的图书管理系统的使用的时候，注册登录的界面都可以进行使用，而且流程都是十分的规范的。而针对读者和管理员两个不同的个体，他们在各自的主页面都能够进行自己权限内的操作，而且操作的流程没有出现问题。在前端页面中我们进行的操作在后台的数据库中都能够得到对应的改变。纵观我们整个系统，我们也的却做到基本功能的实现，而且所有的界面都十分和简约大气。综上所述，我们这次基于MVC模式的图书管理系统的开发基本算取得了圆满的结束。

总 结

在历经了小半年的毕业设计之后，本次基于MVC模式的图书管理系统基本成功的完成了。我的大学四年也接近了尾声，这也成为了我这四年的收官之作。回顾我这个图书管理系统，基于MVC模式进行开发，采用的是三层结构，使用了各种实用的框架技术，结合面向对象进行系统的开发的编程实现，使得这个图书管理系统的各种技术都十分的先进，各种设计都是十分的成熟，这样开发出的图书管理系统合乎设计的规范，科学的流程使得开发过程极具可操作性，后续开发的难度也大大降低，可以轻松的进行功能的更新。结合Java语言的特性，可以一次进行编译，处处能够进行运行，这种灵活的特性，也给这个图书系统增色不少。

在本次图书管理系统的实现过程中，我也不是一帆风顺的。就比如数据库的连接上我就弄了好久，本地数据库在多次的尝试中均未连接上，导致我的登录页面总是无法跳转，显示错误。再后来的资料查询中我多次检查代码，发现原来是数据库的路径没有配置好。经过改正以后，果然连接上了数据库，当时大半夜的真的为之精神一振，根本都不想想睡觉了。

但是由于时间有限，而且作为个人进行开发的能力水平还有所欠缺，这个图书管理系统存在着许多还为解决的缺陷，比如在用户注册的时候使用的用户名时候重复了，注册的数据信息必须先提交到数据库中进行判断处理才能够给用户返回相关结果。这个验证功能还没来得及实现，以及在借阅图书后，图书的库存要进行相应的减少这个问题也没来得及解决，在以后的继续设计中，查询资料以后会解决。其实，还有着一些非常好的功能还没来得及实现，比如图书排行榜，我已经实现了图书的分类，各类的图书可以通过借阅次数来组织一个图书受欢迎程度的排行榜，这样可以给读者一个参考意见。而且系统在真正使用的时候，不管是图书的信息还是读者的身份信息肯定是十分的庞大，这个时候原有的添加方式就会显得十分的低效率，而做成一个文件文档的形式进行整体的输入的想法时候可行，这也是未来继续研究的方向。

以后希望在参考他人对于同类型的系统的设计思路，以及在功能上的想法，可以进一步对自己的系统进行完善，不管是在功能上，还是在界面的设计上或者是与用户的交互体验上争取更加的完善。

# 致 谢

时光荏苒，四年的大学时光弹指间就一晃而过，在这四年的学习生活中，来自于同学老师的帮助实在是太多太多了。回首往昔，种种温情画面不时涌上脑海。这次的研究从选题到开题再到中期检查以及最后的论文，这一切的一切离不开老师同学的帮助，在他们不吝啬的相助下我才能顺利完成这篇论文，在此再次说一声谢谢。

首先我要感谢我的导师，邬少飞老师，三年的导师，一生的恩师。在他和各科课任老师的谆谆教导下，我才会学会许多的知识以及各种学习的方法，感谢他的教导，受用一生。在我这个毕业设计的过程中，是他的建议我选择了这个项目，最终能顺利完成，这第一步多亏了老师的经验支持与知道。在我们不熟悉开题报告的流程的时候，正是老师的帮助我们才能合格的完成开题，这整个毕业设计期间多亏了老师们的指导，我们才能摸着石头过河。

在感谢这次的各位评审老师，你们的认可正是我这四年学习的意义最好的证明。感谢各位不辞辛苦的付出，祝愿各位身体健康。

参考文献

1. 王存娥,杨延宁,吕茕.基于Java的图书管理系统的设计与实现[J].物联网技术,2018,8(04):54-55.
2. 曹定舟,陈云忠,缪毅.医院图书借阅管理系统的设计与应用[J].中国医疗设备,2009,24(04):90-91.
3. 姜浙炜.高校移动图书馆管理系统的设计与实现[J].电脑知识与技术,2016,12(11):60-62.
4. 王惠.基于JSP的企业人事管理系统的设计与实现[J].电脑知识与技术,2018,14(33):75-77.
5. 林琴,王庆伟.基于JavaWeb的图书馆座位预约系统设计与实现[J].电脑知识与技术,2019,15(05):78-79.
6. 周奕.图书管理系统[J].数字技术与应用,2020,38(02):139+141.
7. 王毛毛.基于java的图书管理系统研究[J].信息记录材料,2020,21(1):134-135.
8. Sukumar Mandal, Sushanta Kumar Roy, Abhijit Chakrabarti. Designing an integrated library management and retrieval system using open source tools,2018, 8(3):277-292.
9. 王萍利.基于Java Web的学生成绩查询系统的设计与实现[J].电子设计工程,2013,21(06):18-20.
10. 陈善敏.人工智能技术在图书馆的应用与展望[J].办公自动化,2019,24(18):61-64.
11. Yecies B .Chinese International Film Encounters: Closing the Gaps with Hollywood with Soft Power Appeal at Home and Abroad[J].Creative Media,2014,27(1):1-22.
12. 赵书田,刘海姣.基于JavaScript的动态Web应用系统设计[J].现代电子技术,2017,40(15):44-46+51.
13. 董婷.基于JSP网上购书系统的设计与实现[J].电子设计工程,2014,22(03):11-13.
14. 李萍,李芳.基于B/S结构的医院图书管理系统的设计与实现[J].泰山学院学报,2013,35(03):88-93.
15. 林伟婷.C/S与B/S架构技术比较分析[J].科技资讯,2018,16(13):15-16.
16. 尹全勤.基于B/S架构的图书管理系统研究[J].科技传播,2019,11(06):117-118.
17. Kristiāns Kronis, Marina Uhanova. Performance Comparison of Java EE and ASP.NET Core Technologies for Web API Development. 2018, 23(1):37-44.
18. Ryan F. Buller. An Analysis of “Request It” Print Delivery Service in the Alma Library Management System and Its Impact on Physical Browsing in a Midsized Private Academic Library[J]. Collection Management,2020,45(1):87-98.
19. 李晶莹,王方雄,侯英姿,张闯.基于HTML5/JavaScript的WebGIS功能界面设计与开发[J].微型机与应用,2016,35(07):19-21+24.
20. 张静.图书管理系统数据库的优化设计研究[J].黑龙江科学,2018,9(20):23-24+27.