

**江 苏 科 技 大 学**

**基 于 Three.js 的**

**3D 模 型 共 享 平 台**

学 院： 计算机学院

专 业： 计算机科学与技术

姓 名： 朱庆壁

学 号： 192210701144

指导教师： 宦婧

**2023年4月**

摘 要

随着3D模型在工业设计、建筑设计、游戏开发等领域的应用越来越广泛，3D模型共享成为了一个重要的话题。本文提出了一个基于Three.js的3D模型共享平台，旨在为用户提供高效、便捷的3D模型分享和浏览方式。

当前，WebGL技术得到了广泛的应用，基于WebGL的3D库如Three.js、Babylon.js等也日益受到关注。其中，Three.js是一个非常流行的基于WebGL的3D库，它通过提供简单易用的API，使得在Web中创建和展示3D图形变得更加容易和高效。随着HTML5在2011年成为W3C推荐标准，WebGL技术也随之被广泛支持。同时，移动设备的普及让越来越多的用户通过手机和平板电脑来访问Web，使得基于Web的3D应用需求日益增加。在这种背景下，基于WebGL的3D库应运而生，Three.js也成为了其中最受欢迎的一种。

本平台使用了基于Three.js的WebGL技术，能够在现代浏览器中运行。用户可以通过上传自己的3D模型或浏览其他用户上传的3D模型。同时，本平台还支持3D模型的在线预览、旋转、缩放等基本操作，使用户能够更好地了解和展示自己的作品。

在实现中，本平台采用了Vue 3框架和Spring MVC作为前后端分离的架构，同时使用了jQuery封装的Ajax技术来实现客户端和服务器之间的异步通信，以实现实时的数据传输和页面更新。此外，本平台使用MySQL数据库来存储用户信息、3D模型存放位置等信息，使得数据的组织和管理更加高效、可靠。还使用了MyBatis-Plus框架来简化数据持久化操作。

目前，本平台已经完成了所有的开发和测试，希望能为3D模型的分享和展示提供新的途径。

Abstract

With the increasingly widespread use of 3D models in fields such as industrial design, architectural design, and game development, 3D model sharing has become an important topic. This paper proposes a 3D model sharing platform based on Three.js, aiming to provide users with efficient and convenient ways to share and browse 3D models.

Currently, WebGL technology has been widely used, and 3D libraries based on WebGL such as Three.js, Babylon.js, etc. are also receiving increasing attention. Among them, Three.js is a very popular WebGL-based 3D library that makes it easier and more efficient to create and display 3D graphics in the web by providing simple and easy-to-use APIs. With HTML5 becoming a W3C recommended standard in 2011, WebGL technology has also been widely supported. At the same time, the popularity of mobile devices has led to an increasing demand for web-based 3D applications. In this context, 3D libraries based on WebGL have emerged, and Three.js has become one of the most popular ones.

This platform uses WebGL technology based on Three.js and can run in modern browsers. Users can upload their own 3D models or browse other users' uploaded 3D models. Additionally, the platform also supports basic operations such as online preview, rotation, and scaling of 3D models, allowing users to better understand and showcase their work.

In terms of implementation, this platform adopts Vue 3 framework and Spring MVC as the front-end and back-end separation architecture, and uses Ajax technology encapsulated by jQuery to realize asynchronous communication between the client and server, achieving real-time data transmission and page updates. In addition, this platform uses MySQL databases to store user information, 3D model storage locations, and other information, making data organization and management more efficient and reliable. The MyBatis Plus framework is also used to simplify data persistence operations.

Currently, this platform has completed all development and testing, and hopes to provide a new way for sharing and showcasing 3D models.

目录

[摘 要 I](#_Toc125963863)

[目录 II](#_Toc125963864)

[第一章 绪论 1](#_Toc125963865)

[第二章 项目相关技术介绍 2](#_Toc125963866)

[2.1 后端开发技术 2](#_Toc125963867)

[2.1.1 JAVA语言 2](#_Toc125963868)

[2.2 前端开发技术 2](#_Toc125963869)

[2.2.1 HTML、CSS与JavaScript 2](#_Toc125963870)

[2.3 数据库技术 4](#_Toc125963871)

[2.3.1 MySql 4](#_Toc125963872)

[第三章 系统分析 5](#_Toc125963873)

[3.1 可行性分析 5](#_Toc125963874)

[3.1.1 经济可行性 5](#_Toc125963875)

[3.1.2 管理可行性 5](#_Toc125963876)

[3.1.3 技术可行性 5](#_Toc125963877)

[3.2 系统综合要求 6](#_Toc125963878)

[3.3 功能性分析 6](#_Toc125963879)

[3.4 系统开发总体需求 9](#_Toc125963880)

[第四章 总体设计 11](#_Toc125963881)

[4.1 系统数据库设计原则 11](#_Toc125963882)

[4.1.1 概念结构设计 11](#_Toc125963883)

[第五章 详细设计 15](#_Toc125963884)

[5.1 方案选择 15](#_Toc125963885)

[5.2 数据库和Java代码连接实现 15](#_Toc125963886)

[5.3 HttpServlet接口和tomcat框架的连接实现 16](#_Toc125963887)

[5.3.1 HttpServlet接口 16](#_Toc125963888)

[5.3.2 Tomcat框架 16](#_Toc125963889)

[5.4 后端总体代码结构和层次 17](#_Toc125963890)

[5.4.1 项目分层的意义 17](#_Toc125963891)

[5.4.2 pojo层 17](#_Toc125963892)

[5.4.3 mapper层 18](#_Toc125963893)

[5.4.4 service层 18](#_Toc125963894)

[5.4.5 utils层 19](#_Toc125963895)

[5.5 JavaScript和HttpServlet的连接实现 19](#_Toc125963896)

[5.5.1 HttpServlet通信 19](#_Toc125963897)

[5.5.2 JavaScript数据通信 21](#_Toc125963898)

[5.6 前端不同页面用户保持登录状态功能实现 22](#_Toc125963899)

[第六章 系统实现与测试 26](#_Toc125963900)

[6.1 登录和注册模块 26](#_Toc125963901)

[6.1.1 登录模块 26](#_Toc125963902)

[6.1.2 注册模块 26](#_Toc125963903)

[6.2 主界面 27](#_Toc125963904)

[6.2.1 登出功能 28](#_Toc125963905)

[6.2.2 搜索功能 29](#_Toc125963906)

[6.3 搜索界面 29](#_Toc125963907)

[6.4 管理员管理书籍界面 30](#_Toc125963908)

[6.4.1 书籍添加功能 30](#_Toc125963909)

[6.4.2 书籍修改功能 32](#_Toc125963910)

[第七章 结语 33](#_Toc125963911)

[**参 考 文 献** 34](#_Toc125963912)

1. 绪论
   1. 选题的目的与意义

Web开发在过去几年中发生了很大变化。JavaScript库已经成熟，浏览器变得更加强大，并且大多数系统都很好地支持HTML5规范。不过，目前Web主要关注2D来表示信息、创建游戏和构建网站。但是，当前也有一种创建3D图形的出色且标准化的方法。那就是WebGL；它提供了一个API来创建硬件加速的3D图形。

然而，WebGL API并不是很容易使用。它需要使用C语言进行编程，并且需要确实地深入了解WebGL的内部工作方式才能使用此API。不过幸运的是，有许多可用的JavaScript库隐藏了WebGL的复杂性，并且提供了一个易于使用的API来创建3D应用程序和可视化。目前，这些JavaScript 库中最优秀的，便是Three.js。Three.js提供了一个易于使用的API和一系列可用于访问WebGL的高级材料和形状。

在现代社会，随着3D技术的不断发展，3D模型应用范围越来越广泛。在工业设计、建筑设计、游戏开发等领域，3D模型已经成为必不可少的工具。然而，由于3D模型的复杂性和体积较大，如何高效地共享和浏览3D模型成为了一个重要的话题。

基于此，本文提出了一个基于Three.js的3D模型共享平台。该平台能够为用户提供高效、便捷的3D模型分享和浏览方式。用户可以通过上传自己的3D模型或浏览其他用户上传的3D模型，同时支持在线预览、旋转、缩放等基本操作，以及灯光、材质等高级设置。该平台不仅能够满足用户的3D模型共享需求，同时还能为用户提供更好的展示自己作品的途径。

本文的选题目的是为了解决3D模型共享的问题，提供一个高效、便捷的3D模型共享平台。该平台的意义在于为用户提供了一个全新的3D模型共享和展示方式，可以帮助用户更好地展示自己的作品，同时也可以促进3D模型在各领域的应用发展。

* 1. 国内外研究现状及存在的问题

当今社会，随着科技的不断进步，3D技术应用领域也在不断拓展。3D模型作为3D技术应用的基础，被广泛应用于工业设计、建筑设计、游戏开发等领域。在这些领域，3D模型的共享和交流已经成为一项重要的工作。而基于Web技术的3D模型共享平台，成为实现3D模型共享的一种有效途径。

在国内，已经有一些基于Web技术的3D模型共享平台，如3D溜溜、3D鱼塘等，但这些平台存在一些问题。首先，这些平台的3D展示效果不够优秀，用户体验较差。其次，这些平台对于上传的3D模型的格式和大小有一定的限制，不够灵活。此外，这些平台缺少对3D模型的高级操作，如灯光、材质等的设置，使得用户无法充分展示自己的作品。总之，国内的基于Web技术的3D模型共享平台仍有一些不足之处。

与此相比，在国外，有很多公司和组织已经开发了各种类型的3D模型共享平台。例如，Sketchfab是一个总部位于法国的平台，用户可以上传、展示和共享自己的3D模型，同时也支持一些高级特性，如3D模型在线编辑和VR浏览等。此外，Thingiverse是一个致力于3D打印的社区，用户可以上传和分享自己的3D打印模型，同时也可以浏览和下载其他用户共享的模型。此外，这些平台还支持社交化的功能，使得用户可以方便地分享和交流自己的作品。

然而，国外的这些平台在国内并不十分流行，主要是因为这些平台的语言和界面都是英文的，不够贴近国内用户的需求。同时，这些平台在国内的使用可能会面临一些网络访问问题，如访问速度慢、不能访问等。

针对上述问题，国内外学者们已经进行了大量的研究和探索，提出了许多解决方案。

首先，有一些研究集中在3D模型压缩和传输技术方面，旨在减小3D模型的文件大小并提高传输效率。例如，研究人员在3D模型的网格数据中使用分层细节技术，使得3D模型在不同的分辨率下都可以呈现出良好的效果，从而减小文件大小。另外，也有研究利用多边形网格简化算法来减少模型的三角面数，从而进一步降低文件大小。

其次，还有一些研究集中在3D模型共享和交互方面，旨在提高用户对3D模型的共享和交互体验。例如，研究人员利用虚拟现实技术和手势识别技术来实现对3D模型的自由浏览和操作，从而提高用户的交互体验。同时，也有研究提出了基于社交网络的3D模型共享方案，使得用户可以在社交网络上分享和交流自己的3D作品。

尽管在上述领域中已经取得了一些进展，但是仍然存在一些问题和挑战。例如，在3D模型压缩和传输方面，如何在减小文件大小的同时保证模型的质量和细节，仍然是一个难以解决的问题。在3D模型共享和交互方面，如何实现多用户之间的实时共享和交互，以及如何提高用户对共享模型的信任度和安全性，也是亟待解决的问题。在3D模型版权保护方面，如何避免数字水印和加密算法被破解或绕过，从而保护作者的知识产权，也是一个需要考虑的问题。

* 1. 主要研究内容

1. Three.js技术的应用：此毕业设计通过使用基于WebGL的Three.js技术，实现了一个3D模型共享平台，可以在现代浏览器中运行。为此需要掌握前置知识。它们包括基础的三维数学和几何学知识，如矩阵、向量、坐标系转换等；Three.js 库的基本结构和使用方法，包括如何创建场景、相机、灯光、材质等；还有Three.js 中常用的几何体、贴图、动画等相关知识。
2. 前后端分离架构：此毕业设计采用了Vue 3框架和Spring MVC作为前后端分离的架构，通过使用jQuery封装的Ajax技术来实现客户端和服务器之间的异步通信，以实现实时的数据传输和页面更新。采用前后端分离架构可以将前端和后端的开发分开，降低代码的耦合度，使得代码维护和开发更加方便和高效。
3. MyBatis-Plus框架的应用：这其中包括MyBatis-Plus框架的基本使用方法，包括配置、实体类映射、查询、更新、删除等操作；MyBatis-Plus框架与Spring框架的整合，包括配置、事务管理等；MyBatis-Plus框架与MySQL数据库的整合，包括数据库连接配置、数据源管理等以及MyBatis-Plus框架的高级特性，例如Wrapper查询。
4. MySQL数据库系统与Linux或Windows文件系统的结合设计：这需要研究SQL语言，包括SQL的基本语法和常用的操作命令，如查询、插入、更新、删除等；数据库设计和规范，包括数据表设计、数据字段设计、数据完整性约束等；Linux和Windows系统的文件结构，包括Linux文件系统的树状结构和Windows文件系统的分层结构等。
5. 3D模型共享平台的设计与实现：本研究设计并要实现一个基于Three.js的3D模型共享平台，用户可以通过上传自己的3D模型或浏览其他用户上传的3D模型，同时还支持3D模型的在线预览、旋转、缩放等基本操作，使用户能够更好地了解和展示自己的作品。
6. 系统总体设计
   1. 系统设计思路
      1. 服务器架构

Vue 3和Spring MVC和Nginx和用于传输文件的服务器，分Linux系统和Windows系统

* + 1. 用户模型文件的管理

MySQL中存路径，实际文件存储在文件系统中

* + 1. 模型在线展示功能

返回模型访问路径，DRACOLoader

* + 1. 数据库设计
    2. JAVA语言

Java是一门[面向对象](https://baike.baidu.com/item/%E9%9D%A2%E5%90%91%E5%AF%B9%E8%B1%A1?fromModule=lemma_inlink)的编程语言，不仅吸收了[C++](https://baike.baidu.com/item/C%2B%2B?fromModule=lemma_inlink)语言的各种优点，还摒弃了C++里难以理解的多继承、[指针](https://baike.baidu.com/item/%E6%8C%87%E9%92%88/2878304?fromModule=lemma_inlink)等概念，因此Java语言具有功能强大和简单易用两个特征。Java语言作为静态面向对象编程语言的代表，极好地实现了面向对象理论，允许程序员以优雅的思维方式进行复杂的编程。

Java具有简单性、面向对象、[分布式](https://baike.baidu.com/item/%E5%88%86%E5%B8%83%E5%BC%8F/19276232?fromModule=lemma_inlink)、[健壮性](https://baike.baidu.com/item/%E5%81%A5%E5%A3%AE%E6%80%A7/4430133?fromModule=lemma_inlink)、[安全性](https://baike.baidu.com/item/%E5%AE%89%E5%85%A8%E6%80%A7/7664678?fromModule=lemma_inlink)、平台独立与可移植性、[多线程](https://baike.baidu.com/item/%E5%A4%9A%E7%BA%BF%E7%A8%8B/1190404?fromModule=lemma_inlink)、动态性等特点。Java可以编写[桌面应用程序](https://baike.baidu.com/item/%E6%A1%8C%E9%9D%A2%E5%BA%94%E7%94%A8%E7%A8%8B%E5%BA%8F/2331979?fromModule=lemma_inlink)、Web应用程序、[分布式系统](https://baike.baidu.com/item/%E5%88%86%E5%B8%83%E5%BC%8F%E7%B3%BB%E7%BB%9F/4905336?fromModule=lemma_inlink)和[嵌入式系统](https://baike.baidu.com/item/%E5%B5%8C%E5%85%A5%E5%BC%8F%E7%B3%BB%E7%BB%9F/186978?fromModule=lemma_inlink)应用程序等。

Java作为当下网络软件开发的主流语言，有着极为强大的实用功能，其跨平台的特点也极大的扩大了Java的应用面。

* 1. 前端开发技术
     1. HTML、CSS与JavaScript

（1）HTML

HTML的英文全称是 Hyper Text Markup Language，即超文本标记语言。HTML是由[Web](https://baike.baidu.com/item/Web/150564?fromModule=lemma_inlink)的发明者[Tim Berners-Lee](https://baike.baidu.com/item/Tim%20Berners-Lee/1836386?fromModule=lemma_inlink)和同事Daniel W Connolly于1990年创立的一种[标记语言](https://baike.baidu.com/item/%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80/5964436?fromModule=lemma_inlink)，它是标准通用化标记语言SGML的应用。用HTML编写的超文本文档称为HTML文档，它能独立于各种操作系统平台(如[UNIX](https://baike.baidu.com/item/UNIX/219943?fromModule=lemma_inlink)，[Windows](https://baike.baidu.com/item/Windows/165458?fromModule=lemma_inlink)等)[4]。使用HTML，将所需要表达的信息按某种规则写成HTML文件，通过专用的[浏览器](https://baike.baidu.com/item/%E6%B5%8F%E8%A7%88%E5%99%A8/213911?fromModule=lemma_inlink)来识别，并将这些HTML文件“翻译”成可以识别的信息，即我们所见到的网页。

HTML5是互联网的新一代标准，是构建以及呈现互联网内容的一种语言方式．被认为是互联网的核心技术之一。HTML产生于1990年，1997年HTML4成为互联网标准，并广泛应用于互联网应用的开发。

HTML5是Web中核心语言HTML的规范，用户使用任何手段进行网页浏览时看到的内容原本都是HTML格式的，在浏览器中通过一些技术处理将其转换成为了可识别的信息。HTML5在从前HTML4.01的基础上进行了一定的改进，HTML5仍处于完善之中。然而，大部分现代浏览器已经具备了HTML5的支持。

（2）CSS

层叠样式表(英文全称：Cascading Style Sheets)是一种用来表现[HTML](https://baike.baidu.com/item/HTML?fromModule=lemma_inlink)（[标准通用标记语言](https://baike.baidu.com/item/%E6%A0%87%E5%87%86%E9%80%9A%E7%94%A8%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80/6805073?fromModule=lemma_inlink)的一个应用）或[XML](https://baike.baidu.com/item/XML?fromModule=lemma_inlink)（标准通用标记语言的一个子集）等文件样式的计算机语言。CSS不仅可以静态地修饰网页，还可以配合各种脚本语言动态地对网页各元素进行格式化。

从HTML被发明开始，样式就以各种形式存在。不同的浏览器结合它们各自的样式语言为用户提供页面效果的控制。最初的HTML只包含很少的显示属性，随着HTML的成长，为了满足页面设计者的要求，HTML添加了很多显示功能。但是随着这些功能的增加，HTML变的越来越杂乱，而且HTML页面也越来越臃肿。于是CSS便诞生了。

CSS为HTML标记语言提供了一种样式描述，定义了其中元素的显示方式。CSS在Web设计领域是一个突破。利用它可以实现修改一个小的样式更新与之相关的所有页面元素。CSS具有丰富的样式定义、易于使用和修改、多页面应用、层叠、页面压缩等特点。

（3）JavaScript

JavaScript（简称“JS”）是一种具有函数优先的轻量级，解释型或即时编译型的[编程语言](https://baike.baidu.com/item/%E7%BC%96%E7%A8%8B%E8%AF%AD%E8%A8%80/9845131?fromModule=lemma_inlink)。虽然它是作为开发[Web](https://baike.baidu.com/item/Web/150564?fromModule=lemma_inlink)页面的[脚本语言](https://baike.baidu.com/item/%E8%84%9A%E6%9C%AC%E8%AF%AD%E8%A8%80/1379708?fromModule=lemma_inlink)而出名，但是它也被用到了很多非[浏览器](https://baike.baidu.com/item/%E6%B5%8F%E8%A7%88%E5%99%A8/213911?fromModule=lemma_inlink)环境中，JavaScript 基于原型编程、多范式的动态脚本语言，并且支持[面向对象](https://baike.baidu.com/item/%E9%9D%A2%E5%90%91%E5%AF%B9%E8%B1%A1/2262089?fromModule=lemma_inlink)、命令式、声明式、[函数](https://baike.baidu.com/item/%E5%87%BD%E6%95%B0/301912?fromModule=lemma_inlink)式编程范式。

JavaScript最初由[Netscape](https://baike.baidu.com/item/Netscape?fromModule=lemma_inlink)的[Brendan Eich](https://baike.baidu.com/item/Brendan%20Eich?fromModule=lemma_inlink)设计，最初将其脚本语言命名为[LiveScript](https://baike.baidu.com/item/LiveScript?fromModule=lemma_inlink)，后来Netscape在与Sun合作之后将其改名为JavaScript。JavaScript最初受[Java](https://baike.baidu.com/item/Java/85979?fromModule=lemma_inlink)启发而开始设计的，目的之一就是“看上去像Java”，因此语法上有类似之处，一些名称和命名规范也借自Java，但JavaScript的主要设计原则源自Self和[Scheme](https://baike.baidu.com/item/Scheme?fromModule=lemma_inlink)。JavaScript与Java名称上的近似，是当时[Netscape](https://baike.baidu.com/item/Netscape?fromModule=lemma_inlink)为了营销考虑与Sun微系统达成协议的结果。微软同时期也推出了[JScript](https://baike.baidu.com/item/JScript?fromModule=lemma_inlink)来迎战JavaScript的脚本语言。Vue.js框架

Vue.js是一套构建用户界面的渐进式框架。与其他重量级框架不同的是，Vue采用自底向上增量开发的设计。Vue的核心库只关注视图层，并且非常容易学习，非常容易与其它库或已有项目整合。另一方面，Vue完全有能力驱动采用单文件组件和Vue生态系统支持的库开发的复杂单页应用。

Vue.js的目标是通过尽可能简单的API实现响应的[数据绑定](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E7%BB%91%E5%AE%9A/2884939?fromModule=lemma_inlink)和组合的视图组件。Vue.js由于只聚焦于视图层，所以自身不是一个全能框架。但是其非常容易学习，非常容易与其它库或已有项目整合。另一方面，在与相关工具和支持库一起使用时，Vue.js也能驱动复杂的单页应用。

* 1. 数据库技术
     1. MySql

MySQL是一个[关系型数据库管理系统](https://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E5%9E%8B%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F/696511?fromModule=lemma_inlink)，由瑞典[MySQL AB](https://baike.baidu.com/item/MySQL%20AB/2620844?fromModule=lemma_inlink)公司开发，属于[Oracle](https://baike.baidu.com/item/Oracle?fromModule=lemma_inlink)旗下产品。MySQL是最流行的[关系型数据库管理系统](https://baike.baidu.com/item/%E5%85%B3%E7%B3%BB%E5%9E%8B%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F/696511?fromModule=lemma_inlink)之一，在[WEB](https://baike.baidu.com/item/WEB/150564?fromModule=lemma_inlink)应用方面，MySQL是最好的[RDBMS](https://baike.baidu.com/item/RDBMS/1048260?fromModule=lemma_inlink) (Relational Database Management System，关系数据库管理系统)应用软件之一[5]。

MySQL是一种关系型数据库管理系统，关系数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。

MySQL所使用的SQL语言是用于访问[数据库](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%BA%93/103728?fromModule=lemma_inlink)的最常用标准化语言。MySQL软件采用了双授权政策，分为社区版和[商业版](https://baike.baidu.com/item/%E5%95%86%E4%B8%9A%E7%89%88/1817444?fromModule=lemma_inlink)，由于其体积小、速度快、总体拥有成本低，尤其是[开放源码](https://baike.baidu.com/item/%E5%BC%80%E6%94%BE%E6%BA%90%E7%A0%81/7176422?fromModule=lemma_inlink)这一特点，一般中小型和大型网站的开发都选择MySQL作为[网站数据库](https://baike.baidu.com/item/%E7%BD%91%E7%AB%99%E6%95%B0%E6%8D%AE%E5%BA%93/6399264?fromModule=lemma_inlink)。

1. 系统分析

可行性，是在当前状况下，是否满足开发系统的条件。在系统开发之前，我们要针对该系统做详细的调查，然后根据调查报告来分析和总结，总结出系统如果开发会出现的问题，最后决定是否有必要开发这个系统，这个过程叫做可行性分析。管理系统实现的条件和可行性是息息相关的。系统的可行性分析与系统的风险分析是成反比的，开发的系统风险越大，那么此管理软件的可行性就会非常小。我通过对学院的调研和分析，对该管理系统做的可行性分析如下。

* 1. 可行性分析
     1. 经济可行性

目标系统开发所需要求比较低，且系统不是十分复杂，开发的周期较短，人员经济支出有限。当系统开发完实际运行后，将会改变学校原有的图书手工管理，给许多读者带来方便，并且系统的开发将提高读者的时间利用率。

* + 1. 管理可行性

管理可行性主要考虑的内容主要有：①管理员是否支持此系统软件的开发；②学院是否具备管理的条件；③系统的后期管理工作是否能保证正常。伴随着电子信息的飞速发展，计算机已经全民普及，普通人员都具有一定的计算机知识，同时开发设计人员将数据信息标准化，系统界面简单化，所以系统的后期管理是非常方便的。

学校每个职工均有借书证，每个学生的学号也都不相同，同时每个读者在馆里都有自己的个人信息，因此此系统在管理上是可行的。另外，管理员非常支持开发此系统软件，同时在软件正式使用之前也会有相关的培训计划。

由上可知，开发此图书管理系统在经济、技术、管理上面都是可行的。

* + 1. 技术可行性

学校只需要建立一个局域网，并引入适当量的硬件设备就可以实现图书管理系统的应用，目标系统准备使用Java技术实现，目前这种技术已经普遍，因此在技术手段上实现本系统成为可能，高校也有计算机师资力量，对一定的软件师生有能力在一定时间内掌握。综上所述，目前实现目标系统的条件已经较为成熟。

* 1. 系统综合要求

系统分析是在了解系统思想的基础上把复杂的模块分为多个简单的部分，然后把每个模块的属性了解清楚，每个模块之间的关系也需要知道。系统需求分析就是要详尽的阐述系统的功能和性能，保证开发的系统和用户的需求是非常统一的。需求分析可分为三个阶段来进行，有问题分析、需求阐述和需求审核。在问题分析的最初阶段，设计人员可以对用户进行访谈，也可以以会议的形式来了解用户的需求，设计人员也可以到实地去考察，去图书馆工作一段时间观察和了解管理员的工作。问题分析的关键技术有问题的抽象、问题的分析和需求建模。需求阐述时产生的需求说明的格式必须非常标准。需求规格的主体是由需求模型组成。在需求审核阶段，设计人员在审核需求说明时必须遵循一定的标准：需求一致性、需求正确性、需求可修改性、需求完整性、需求无岐义性、可验证性、可理解力性、可追踪性。

分析用户的需求是软件需求分析中一个重要的环节，也是最困难的，因为不是所有用户对计算机都十分了解，所以开发人员要尽可能的多与用户沟通,明确用户的真实需求。对用户的需求了解的越详细，软件后期维护就越容易。对于图书管理系统来说，读者足不出户就可以查看新书公告、查看和修改自己的个人信息、借阅的图书信息、已过期的图书信息、浏览所需要的图书，可以查看最近一段时间借阅率高的图书。管理员所需要的功能如下：第一，能够对图书进行管理（包括新书的录入和删除以及图书的分类管理）；第二，对读者进行管理（添加、修改和删除读者信息）；第三，发布公告信息；第四，有借书、还书、续借功能。

* 1. 功能性分析

功能需求分析是设计人员对管理员和借阅者需要的系统功能进行分析，它是建立系统的基础，需求分析是否到位，决定这个系统在今后的使用效果。经过对我院图书馆的详细调研，在满足用户需求的基础上合理规划系统，经过优化整理确定系统最终要运行的主要功能包括：图书管理、借阅者管理、系统管理、统计排行、图书借还、图书检索等功能。

1. 系统管理

系统管理实现的功能主要包括本馆介绍功能、管理员的设置功能、操作权限设置功能、口令更改功能、系统参数设置功能、系统提醒功能、数据维护功能等功能，用例图如图3.1所示。

图示

描述已自动生成

图3.1 系统管理用例图

1. 图书管理

图书管理实现的功能包括新书录入功能、图书出库功能、图书修改功能、新书公告功能。用例图如图3.2所示。

图示

描述已自动生成

图3.2 图书管理用例图

1. 借阅者管理

借阅者管理实现的功能有：查询个人信息功能、修改个人信息功能、借阅证挂失功能、密码修改功能。借阅者管理用例图如图3.3所示。

图示

描述已自动生成

图3.3 借阅者管理用例图

1. 图书借还

图书借还实现的功能包括：预约功能、预借功能、借阅功能、续借功能、归还功能。用例图如图3.4所示。

图示

描述已自动生成图3.4 图书借还用例图

1. 图书检索

图书检索实现的功能包括：按书名检索、类别检索、作者检索、出版社检索、图图示

描述已自动生成书编号检索、关键字检索等功能。用例图如图3.5所示。

图3.5 图书检索用例图

1. 图书推荐

图书推荐实现的功能有图书借阅数量排行、图书借阅分类排行、读者借阅分类排行功能。用例图如图3.6所示。

图示

描述已自动生成

图3.6 图书推荐用例图

* 1. 系统开发总体需求

建设图书管理系统会涉及用户的实际操作和使用体验，因此建设过程要遵循这些原则：

（1）易用性：尽可能设计简单易操作的图书管理系统来达到用户的标准。用户偏向功能全面、简单上手、通俗易懂的图书管理系统，当然，系统的设置也要符合读者和图书馆工作人员的日常使用习惯，以降低用户掌握系统的困难指数。

（2）安全性：图书管理系统的本质是一套能够对大量数据进行存储管理的系统，所以本系统需要相当高的安全性能和针对数据的严密防护体系，每一个登录系统的用户都需要进行严格监管，以防信息外泄带来不可预估的重大损失。

（3）稳定性：图书馆的用户群体较大，所以系统的稳定性方面也有一定要求，系统应该采取故障恢复和自我检查等方法来保证其稳定性。

（4）经济性：图书管理系统主要是给图书馆工作人员和在校学生使用，因此系统设计的要尽可能的节约成本。

（5）灵活性：图书管理系统应可以对图书馆实际情况的变化及时进行调整，以提高系统灵活的部署能力。

1. 总体设计

总体设计中首先寻找实现目标系统的各种不同的方案，需求分析阶段得到的数据流图是设想各种可能方案的基础。在总体阶段确定软件的结构，确定系统中每个程序是由哪些模块组成的以及这些模块相互之间的关系。

* 1. 系统数据库设计原则

在设计数据库时要以功能需求中的数据为衡量标准，遵循以下的原则：

（1）一致性。要求在系统的全程过程中表示同一事物的数据具有正确性和唯一性，数据结构的一致性，尤其在关系数据库中，更加要求当事务完成时，必须使所有数据都具有一致的状态。

（2）通用性。系统中的数据不仅仅是一个模块对它操作，是所有的模块都能对它进行操作（查询、修改等），这就要求数据具有很好的通用性。

（3）规范性。规范性是大多事物都具有的性质，数据也不例外，只有保证数据的规范性才能确保数据不会冗余、修改正常等问题，才能不影响系统运行。

（4）扩展性。用户的功能需求不会一成不变，也包括前期的需求分析工作再全面也可能会有遗漏的问题，所以必须在设计的时候保证数据的可扩展性、伸缩性和移植性等。

（5）可维护和管理。系统中的数据是可以随时变化的，所以要确保后期数据具有可维护和方便管理的性质。

（6）安全性。要保证系统的正常运行，不仅数据要正确，而且要确保数据的安全，确保数据不能被恶意篡改。

* + 1. 概念结构设计

概念设计就是在数据需求分析后，运用自底向上的方法建立的概念结构。概念模型是站在用户的角度把用户需要的信息和数据转化成看得见的结构，这里概念模型我们用 E-R 图来表示。E-R 图中矩形表示实体集，椭圆则表示属性，菱形表示实体之间的联系。

图书管理系统涉及的实体主要有读者、管理员、图书等，各实体的属性如下所示：

（1）借阅者信息的数据项有：借阅者姓名、借阅者密码、借阅者学号、性别、班级、电话号码、已借书数量等。

（2）管理员信息的数据项：管理员姓名、工号、密码等。

（3）图书信息的数据项有：书名、图书编号、作者、出版社、类别、入库日期、图书的总数量、图书当前数量等。

（4）借书信息的数据项有：借阅者学号、图书编号、借书日期等。

（5）图书检索的数据项有：书名、类别、关键字、图书编号、作者、出版社等。

（6）图书推荐的数据项有：图书借阅数量、图书借阅分类、读者借阅分类等。根据上面得到的实体和实体之间的联系，就能得出系统的E-R图。

图示

描述已自动生成（1）借阅者 E-R 图设计

图4.1 读者E-R图

（2）管理员E-R图

图示

描述已自动生成

图4.2 管理员E-R图

（3）图书信息E-R图

图示

描述已自动生成

图4.3 图书信息E-R图

（4）借书E-R图

图示

描述已自动生成

图4.4 借书E-R图

（5）图书检索E-R图

图示

描述已自动生成

图4.5 图书检索E-R图

（6）图书推荐E-R图

图示

描述已自动生成

图4.6 图书推荐E-R图

（7）综合以上模型，得到的整个系统的E-R图设计

图示

描述已自动生成

图4.7 系统E-R图

1. 详细设计
   1. 方案选择

本项目采用前后端分离的设计，前端主要采用HTML和jQuery框架，通过JavaScript实现前端页面和后端服务器间的交互。后端运用了Apache Maven框架和Java EE提供的HttpServlet接口、由Apache Tomcat提供运行环境。利用mybatis框架实现Java和MySQL的交互。

* 1. 数据库和Java代码连接实现

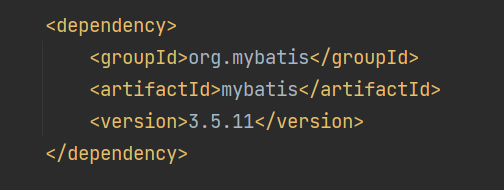
这一部分内容在Java代码中主要用到了mybatis框架，使用了MySQL数据库。由于项目采用的是Maven框架，我们可以通过该框架的pom.xml文件很方便的导入我们项目所需要的各种依赖，而该小节中的mybatis依赖就是通过pom文件所添加的。依赖添加代码如图5.1所示。

图5.1 依赖代码

依赖添加完后，下一步便是配置依赖。mybatis的配置需要借助xml配置文件来实现首先我们需要选定将Java代码连向MySQL的数据库驱动，而这里我们可以选择com.mysql.cj.jdbc.Driver，其次我们需要向mybatis指明数据库的包含端口连接路径和对数据库进行操作的用户，需要包含用户名和登录密码最终的主要配置代码如图5.2所示。

所谓用Java代码操作数据库，其实并不是真正用Java代码来直接与数据库做交互，而是需要将特定的Java代码映射为对应的MySQL代码，而这就要引入我们的mybatis映射文件了。和配置文件一样，映射文件同样通过xml文件进行配置。就像前面提到的一样，映射文件的主要作用就是将Java代码映射为开发者想要的SQL语句。而这就是简单的将SQL语句写进映射文件，写好之后我们还要将该映射文件在项目中的路径添加到配置文件中。

 最终，我们便可以通过自己定义的mapper接口来实现通过Java代码来对数据库中的数据进行增删改查操作。

图5.2 数据库配置

* 1. HttpServlet接口和tomcat框架的连接实现
     1. HttpServlet接口

所谓HttpServlet，就是JavaEE提出的13个规范之一，其中上节提到的xml文件也是JavaEE的规范之一。看到Http这个词便可以联想到网页。而这个接口这是用来实现Java代码和网页之间的交互的。

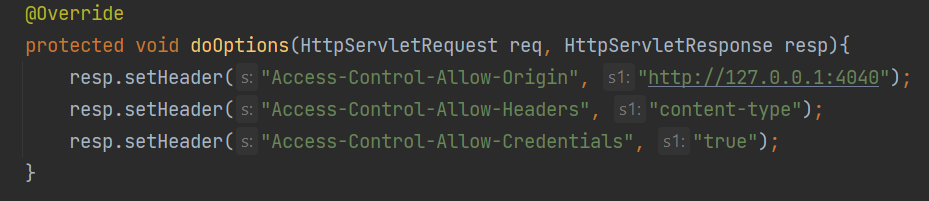
而说到Http，就不得不提我们最常遇到的几个数据请求方式了。分别是GET，POST和OPTIONS请求方式。而HttpServlet接口就帮我们对前端以不同的请求方式发送过来的数据分别提供了不同抽象方法。我们对不同的请求方式只需要分别重写HttpServlet接口不同的方法便可以接受到对应的数据。不如说一个简单的OPTIONS方式的重写如图5.3所示。

图5.3 OPTIONS方式的重写

图5.3

重写好不同的方法后还不足以做到Java代码和Web之间的交互，我们还要给不同的Service类分别绑定一个路径，我们可以注解的方式来绑定路径，同样也可以通过xml配置文件进行配置。

* + 1. Tomcat框架

前面提到的HttpServlet接口，我们讨论的一直都是接口，而不是实现类。众所周知，接口是无法直接实现功能的，Java语言中真正实现各种各样的功能是它内部丰富的实现类。所以仅凭HttpServlet还是无法实现Java代码和Web间的交互。真正实现这类功能的实现类被继承到了tomcat中，而这就体现了tomcat的重要性。

最终我们的Web项目要运行起来，就是要将项目部署到tomcat服务器上。而一般的Java项目最终会被封装为jar文件，而由于tomcat是一个Web服务器，我们若想将项目部署到该服务器上需要将我们的Java项目封装为war文件。而这只需要在Maven框架的pom文件内添加一句项目的封装方式就能实现了。得到了项目的war文件后，我们需要将该文件放置到tomcat文件内的webapps目录下。到这一步，我们的Java代码便可以接收到Web传来的各类请求数据了。

* 1. 后端总体代码结构和层次
     1. 项目分层的意义

项目分层是为了符合“高内聚，低耦合”思想，把各个功能模块划分为表示层（UI）、业务逻辑层（BLL）和数据访问层（DAL）三层架构，各层之间采用接口相互访问，并通过对象模型的实体类（Model）作为数据传递的载体，不同的对象模型的实体类一般对应于数据库的不同表，实体类的属性与数据库表的字段名一致。

分层后开发人员分工更明确，将精力更专注于应用系统核心业务逻辑的分析、设计和开发，加快项目的进度，提高了开发效率，有利于项目的更新和维护工作。一个开发中的项目层次结构大致如图5.4所示。

* + 1. pojo层

该层的内容即为数据库表对应的实体类，与数据库中的属性值基本保持一致。该层也可以称为model层或者domain层。而所谓实体类，便是数据库中Table在Java代码中的具体体现，而由于Java中的一切内容都是以类为单位的，所以该层的内容也被封装为一个个不同的类，每个类都与数据库中某个表之间有一个唯一的对那个关系。

* + 1. mapper层

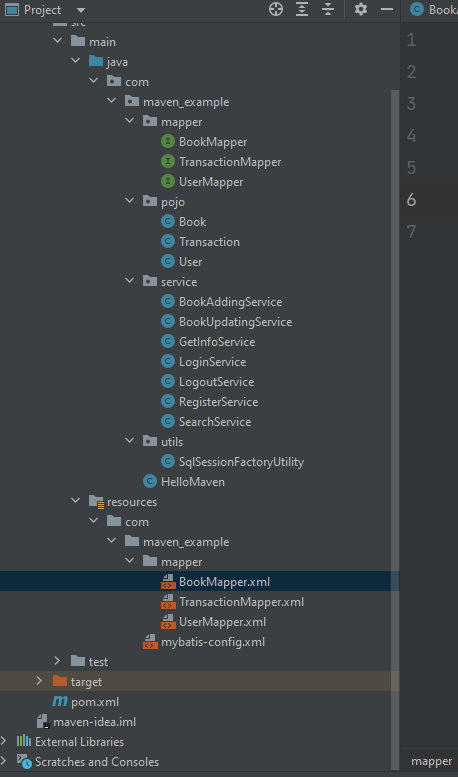
又被称作DAO层，中文叫数据访问层，全称为Data Access Object。某个DAO一定是和数据库的某一张表一一对应的，其中封装了CRUD（增加Create、检索Retrieve、更新Update和删除Delete）基本操作，DAO只做原子操作。无论多么复杂的查询，DAO只是封装增删改查。至于增删查改如何去实现一个功能，该层并不负责，该层只需要将每一个细节操作封装为可以被上层的类调用的函数。

图5.4 项目文件结构

在“数据库和Java代码连接实现”一节中也提到了mapper层，在本项目中，mapper层主要包含了mybatis的映射文件对应的多个接口，接口内实现类对各各数据库表的各种操作的抽象方法，具体实现操作数据库的MySQL代码存放在mybatis的映射文件中。

* + 1. service层

service层属于业务层，一般项目的业务逻辑就放在该层中。该层的类会调用上面提到的mapper层内的接口。在本项目中给层中的一个类就对应着一个业务功能，比如添加图书的功能就对应着该层中的BookAddingService类。在类中，一个标准的流程就是先通过mybatis的SqlSessionFactory类来建立一个该服务类当前与数据库的一个连接，也就是SqlSession类，通过这个连接获得mapper层接口中对应的类以便通过类间接对数据库进行操做。

为了保证能顺利和Web进行沟通，这期间可能还会遇到一些跨域问题或者关于Session或者Cookie的传递，我们还需要对响应头添加一些设置，向接收响应的Web说明该响应具有哪些性质以便Web能够正常接收响应。

* + 1. utils层

准确来说该层并不属于业务处理的层次，该层的主要任务是提供一些其他层可能会经常用到的工具，方便其他层的类来对该层的类进行调用，这样可以大量的减少项目中的重复代码。比如在本项目中，utils层中放有一个SqlSessionFactory的工具类，因为service层中每个需要操作数据库的功能都需要先与数据库建立一个连接，而建立连接需要SqlSessionFactory对象，因此该对象会被service层中的大多数类所调用，而如果我们分别在每个service类中都分辨建立一个SqlSessionFactory，除了会导致大量的重复代码外，更为严重的是该类会占用系统大量的资源，所以我们应该在utils层中建一个工具类，此后每当项目中的某个类需要到对应工具时，就可以直接通过该工具类获得该工具，大量提升了系统运行的效率和项目代码的可阅读性。

* 1. JavaScript和HttpServlet的连接实现
     1. HttpServlet通信

前后端想要相互通信，就必须事先约定好发送的数据的编码和解码的格式，而在这点上HttpServlet似乎做得和其他很多框架或者工具不太一样。因为该项目中几乎所有框架的默认编码方式都是“UTF-8”，而HttpServlet的默认编码方式却是“ISO-8859-1”，这就导致了一个很严重的问题。比如说在添加图书的时候，该图书管理系统内几乎所有的书都是中文的，包括书名、作者、描述等信息，同样存在数据库中的这些信息也是中文的。当我们在实现查询功能的时候，我们需要根据用户的输入来向Web返回一个包含大量书本信息的列表，也就是一个List。这时候如果我们采用HttpServlet默认的编码方式，Web端收到的信息就全部以‘？’符号呈现，其中一个结果如图5.5所示。

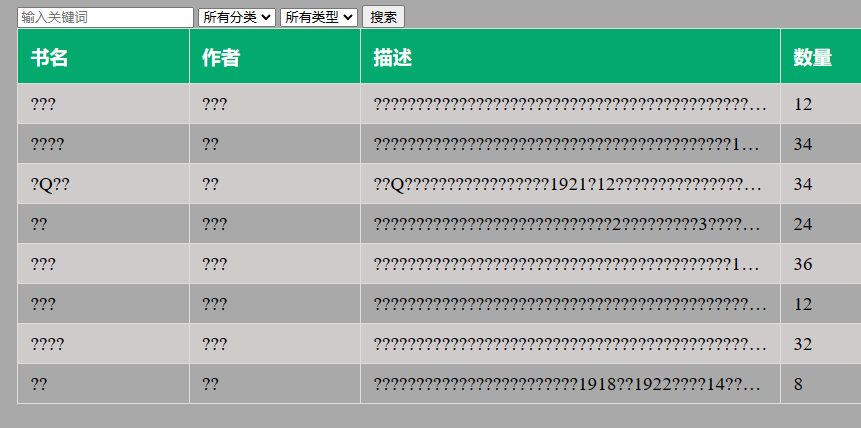
因此，前后端在相互通信前必须事先约定好请求和相应格式，不仅仅是从编码上，还要从传递的数据类型上。至于编码上，我们只需要给HttpServlet的返回的响应头和接收到的请求的相应编码类型手动第设置为“UTF-8”的字符集即可。在数据类型上，本项目中前后端约定除了简单的字符串响应数据外，其他所有信息均通过JSON来传递，前端将数据封装成JSON字符串，后端便通过解析该JSON字符串获得对应的pojo层的类，并通过该类向mapper层的接口传递参数。

图5.5 错误书籍信息列表

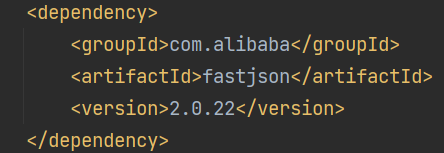
而为了解析由JavaScript传递过来的JSON字符串，本项目中引入了阿里云的一个依赖，叫做fastjson，如图5.6所示。面对由JSON字符串，我们可以调用该依赖的JSON.parseObject方法，将收到的JSON字符串和想要得到的类对象的字节码文件作为参数传递给该方法就可以顺利的收到前端传来的请求数据了。相应的，在返回数据时，我们可以将当前的数据或者列表作为参数传递给fastjason依赖的JSON.toJSONString方法，便可以得当当前数据对应的JSON字符串，随后我们再将该字符串写入响应的打印流当中就可以让前端收到后端响应的数据了。

图5.6 配置JSON依赖

* + 1. JavaScript数据通信

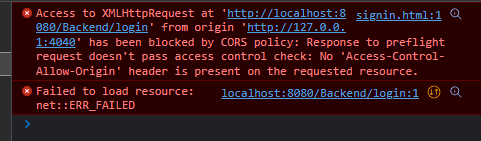
这里以用户的登录功能为例。用户输入用户名和密码后，我们需要将用户名和密码传递给后端服务器。首先通过Document方法取得对应输入框的值，即value属性，然后使用POST的数据请求方式将两个字符串封装为JSON字符串向后端发送数据。但是，仅仅这样是完全不能让后端接受刚刚发送的请求参数的。我们这时候可能会收到浏览器返回的类似图5.7的错误信息，出现这个错误信息的原因在于这里涉及到了前后端相互通信的跨域问题以及preflight的相关知识。

图5.7 JSON数据接受错误

首先来介绍一下什么是跨域问题。跨源资源共享（CORS，或通俗地译为跨域资源共享）是一种基于 HTTP 头的机制，该机制通过允许服务器标示除了它自己以外的其它源（域、协议或端口），使得浏览器允许这些源访问加载自己的资源。跨源资源共享还通过一种机制来检查服务器是否会允许要发送的真实请求，该机制通过浏览器发起一个到服务器托管的跨源资源的“预检”请求。在预检中，浏览器发送的头中标示有 HTTP 方法和真实请求中会用到的头。

跨源 HTTP 请求的一个例子：运行在 https://domain-a.com 的 JavaScript 代码使用 XMLHttpRequest 来发起一个到 https://domain-b.com/data.json 的请求。

出于安全性，浏览器限制脚本内发起的跨源 HTTP 请求。例如，XMLHttpRequest 和 Fetch API 遵循同源策略。这意味着使用这些 API 的 Web 应用程序只能从加载应用程序的同一个域请求 HTTP 资源，除非响应报文包含了正确 CORS 响应头。

CORS 机制允许 Web 应用服务器进行跨源访问控制，从而使跨源数据传输得以安全进行。现代浏览器支持在 API 容器中（例如 XMLHttpRequest 或 Fetch）使用 CORS，以降低跨源 HTTP 请求所带来的风险。

跨源资源共享标准新增了一组 HTTP 标头字段，允许服务器声明哪些源站通过浏览器有权限访问哪些资源。另外，规范要求，对那些可能对服务器数据产生副作用的 HTTP 请求方法（特别是 GET 以外的 HTTP 请求，或者搭配某些 MIME 类型的 POST 请求），浏览器必须首先使用 OPTIONS 方法发起一个预检请求（preflight request），从而获知服务端是否允许该跨源请求。服务器确认允许之后，才发起实际的 HTTP 请求。在预检请求的返回中，服务器端也可以通知客户端，是否需要携带身份凭证（例如 Cookie 和 HTTP 认证相关数据）。

从图5.7的信息中我们可以得知该错误产生的原因是预检请求（preflight request）没能通过访问控制检测。而预检请求是使用OPTIONS方法发起的，所以我们在接受用户输入的信息时，不能只重写HttpServlet的doPost方法，还需要重写doOptions方法以对预检请求进行处理。就像图5.7所提示的那样，我们需要在重写的doOptions方法中给响应头设置Access-Control-Allow-Origin参数，将该参数与前端的url地址进行绑定。另外还应注意到我们的数据是以JSON字符串来进行传递的，所以只设置Access-Control-Allow-Origin参数还是不够的，这依然不能让预检请求通过访问控制检测。我们还需要给HttpServlet的响应头设置Access-Control-Allow-Headers参数，以向浏览器表明该服务器可以接受什么样的请求头。同样，在重写的doPost方法中我们同样要给响应头设置这两个参数。

至此，我们前端和后端就可以相互发送数据并保证对方能正常接收到了。

* 1. 前端不同页面用户保持登录状态功能实现

在上一节通过系统的登录功能描述用户输入的用户名和密码在前后端之间传输大致说明了在跨域情况下前后端之间通信会遇到的几个问题，但我们收到用户输入的数据并向浏览器返回登录成功的信息后，仅仅通过JavaScript和HTML是没法让用户在除了登录界面的其他页面保持登录状态的。如果我们希望在用户登陆成功后页面自动跳转到主页，同时保持用户的登录状态，比如在页面右上角显示当前用户的用户名和头像之类的功能，像这种功能在目前介绍到的技术下还是没法实现的。概括地说就是我们希望让前端收到数据后能将数据存储下来。而这，就要介绍一下Cookie和Session的相关知识了。

首先说一下Cookie。HTTP Cookie（也叫 Web Cookie 或浏览器 Cookie）是服务器发送到用户浏览器并保存在本地的一小块数据。浏览器会存储 cookie 并在下次向同一服务器再发起请求时携带并发送到服务器上。通常，它用于告知服务端两个请求是否来自同一浏览器——如保持用户的登录状态。Cookie 使基于无状态的 HTTP 协议记录稳定的状态信息成为了可能。

Cookie 主要用于以下三个方面：（一）会话状态管理：如用户登录状态、购物车、游戏分数或其它需要记录的信息；（二）个性化设置：如用户自定义设置、主题和其他设置；（三）浏览器行为跟踪：如跟踪分析用户行为等。

从前面的介绍可以看出仅仅使用Cookie我们便可以实现保持用户的登录状态这一功能，但是Cookie的一个局限性在于：它是存储在客户端的，而存储在客户端就会带来很大的安全隐患。尤其是在对用户名和密码这些敏感数据而言，将这些信息存储在客户端无疑给项目带来了极大的安全隐患。因此我们需要一种更安全的方案来存储用户的敏感数据。而这种更安全的方案便是Seesion。

与Cookie类似，Session同样是用来存储各各客户端的信息的。但有一个根本性的区别就是：Cookie存储在客户端，而Session是存储在服务器端的，正是因为这一点，同样的数据，使用Session存储会比使用Cookie存储安全的多。而服务器判断不同的请求是否来自同一个浏览器的功能的实现就通过浏览器向服务器返回的SessionID来判断。

当用户第一次登录服务器时，服务器会向浏览器返回一个SessionID，SessionID实际上是一个字符串，每一个在一定时限内访问过服务器的浏览器都会有一个唯一SessionID来唯一地标识自身的身份。而SessionID是通过Cookie最开始由服务器像浏览器发送的，所以说Session是基于Cookie实现的。与直接将数据存储在Cookie中不同，通过Session我们可以将数据存在服务器，而从客户端所看到的，就是一串SessionID。而此后每当浏览器像访问服务器时，一般情况下浏览器都会自动携带SessionID一项服务器表明自己的身份。

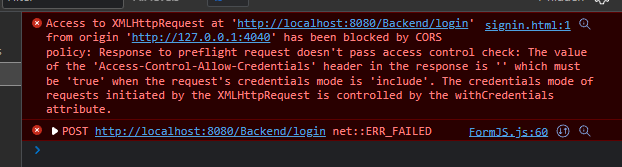
可在跨域的情况下，浏览器在向后端传递数据时默认是不会带上Cookie或者SessionID等验证信息的，我们需要给XMLHttpRequest对象设置withCredential属性。将该属性的值设置为true，即表明我们需要在跨域的情况下向服务器发送携带Cookie或SessionID等用于验证身份的数据。而这时我们再向服务端发送数据依然没能实现保持用户登录状态的功能。我们很可能收到如图5.8所示的错误提示。

图5.8 用户登录错误

根据提示我们可以知道，若想在跨域模式中传递身份验证信息，不能只有客户端或者服务器端表示能够发送或接受该类型的数据。我们还需要在服务器的给响应数据设置Access-Control-Allow-Credentials参数。

Access-Control-Allow-Credentials 响应头用于在请求要求包含 credentials（Request.credentials 的值为 include）时，告知浏览器是否可以将对请求的响应暴露给前端 JavaScript 代码。

当请求的 credentials 模式（Request.credentials）为 include 时，浏览器仅在响应标头 Access-Control-Allow-Credentials 的值为 true 的情况下将响应暴露给前端的 JavaScript 代码。

Credentials 可以是 cookies、authorization headers 或 TLS client certificates。

当作为对预检请求的响应的一部分时，这能表示是否真正的请求可以使用 credentials。注意简单的GET 请求没有预检，所以若一个对资源的请求带了 credentials，如果这个响应头没有随资源返回，响应就会被浏览器忽视，不会返回到 web 内容。

Access-Control-Allow-Credentials 标头需要与 XMLHttpRequest.withCredentials 或 Fetch API 的 Request() 构造函数中的 credentials 选项结合使用。Credentials 必须在前后端都被配置（即 Access-Control-Allow-Credentials header 和 XHR 或 Fetch request 中都要配置）才能使带 credentials 的 CORS 请求成功。

至此，我们从数据库到浏览器的连接功能算是基本完成了。

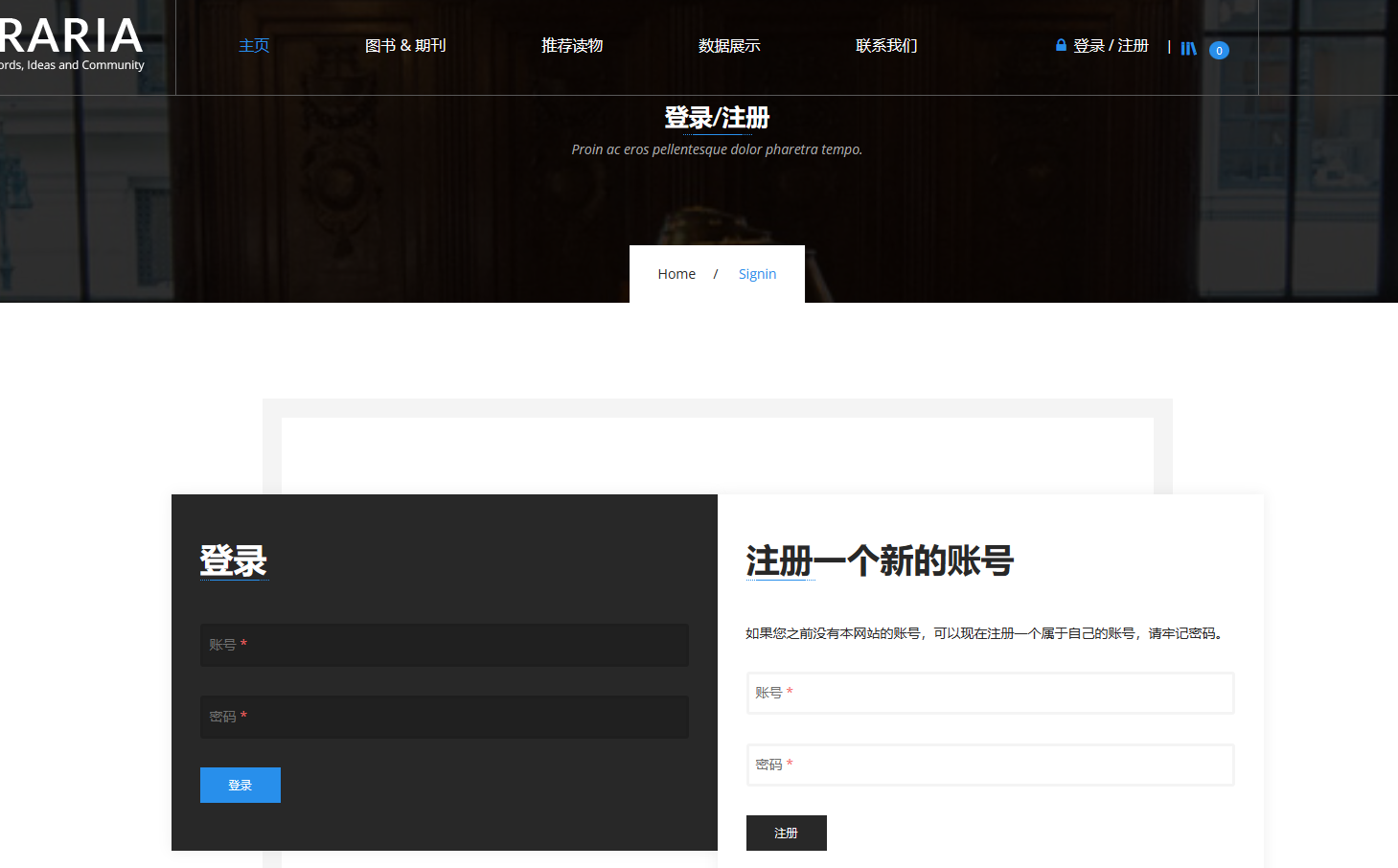
1. 系统实现与测试
   1. 登录和注册模块

图6.1 登录与注册

* + 1. 登录模块

从图6.1可以看出，在本系统中，我们将登录和注册功能设置在了同一界面。用户在输入框中分别输入用户名和密码。而且用户输入的信息并不会立马交给后台进行检验。用户输入完后，会首先在前端检测用户输入的用户名和密码是否符合规范，比如用户名的长度应大于等于三，密码的长度大于等于六。输入框中还限制了用户的输入长度，用以防止有人蓄意提升服务器运行压力。

当用户输入的用户名不符合规范时，浏览器给出的提示如图6.2所示。而当用户输入的用户名和密码不匹配时，浏览器给出的提示如图6.3所示。若用户输入了正确的信息，浏览器会自动重定向到网站的主页。

* + 1. 注册模块

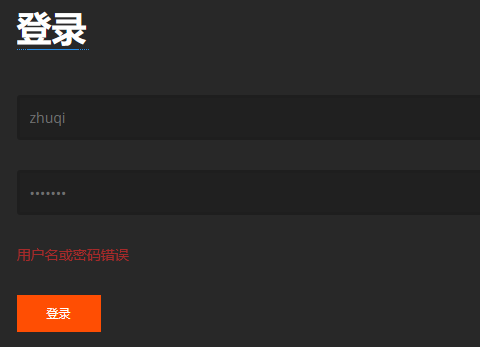
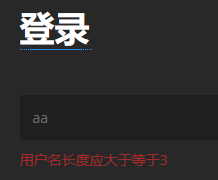
与登录模块类似，注册模块也会在将数据交给后台处理前先验证用户的输入是否符合规范当用户输入的注册信息长度不符合规范时，浏览器给出的提示如图6.4所示。同时，若用户未对不规范的数据进行修改，此时的“注册”按钮点击后是无效的。

图6.2 用户名提示

图6.3 用户信息错误提示

图6.4 注册账号信息提示

由于用户名是表示用户身份的标识符，所以用户输入的用户名不能与数据库中已有的用户名出现重复。若当前用户输入的用户名已经被其它用户注册过了，浏览器给出的提示如图6.5所示。若用户注册成功，浏览器给出的提示如图6.6所示。注册成功后，用户便可以用刚注册的账户进行登录并跳转到主页了。

* 1. 主界面
     1. 登出功能

主界面如图6.7所示。得益于上一章对Cookie和Session的分析，观察该页面的右上角，可以发现右上角给出了当前用户的用户名。看到用户名右边向下的箭头可以到该处为一个下拉菜单，当我们把鼠标停留在其上方时，弹出的菜单如图6.8所示。

图6.6 注册成功

图6.5 用户名重复

若用户点击了“Logout”按钮，则浏览器右上角显示的信息变为“登录/注册”。如图6.9所示。此时该按钮的功能为一个连接，用户点击该链接页面就会跳转到上一节所介绍的登录/注册界面。

图6.7 登录跳转页面

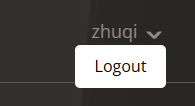
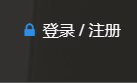
* + 1. 搜索功能

图6.9 退出后展示信息

图6.8 Logout按钮

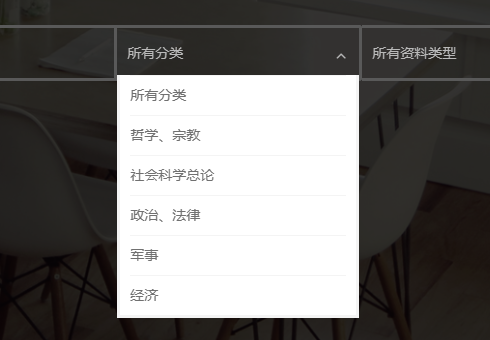
主页正中央有三个搜索框和一个提交按钮，该模块用于完成用户的书籍搜索功能，用户可以在左边第一个搜索框内输入关键词，可以是作者的关键词也可以是书名的关键词，同时该搜索功能还支持模糊查询，用户不需要记得作者或书籍的全名也能够搜索出自己想要的书籍。右边两个为选项框，当用户分别点击两个选项框后浏览器给出的提示分别如图6.10和图6.11所示。

图6.10 图书分类

图6.11 资料类型

用户可以在其中选择自己想要的分类或类型，这样服务器就会对返回的书籍进行筛选。当用户点击“SEARCH”按钮后，浏览器会自动跳转到搜索界面并展示出相应的搜索结果。

* 1. 搜索界面

搜索界面如图6.12所示。该页面列出了刚刚在主页搜索的返回结果。在该界面可以看到相应书籍的书名、作者、描述和剩余量信息。右边的“借书”和“还书”按钮可以实现用户借书和还书的功能。

同时，列表左上角的搜索模块还支持动态查询，用户不需要对所有选项框进行选择也可以不在输入框中输入关键词也能实现搜索功能。若用户什么都不输入便点击“搜索”按钮，则浏览器返回的是图书馆中所有书籍的列表。

另外关键词输入框还支持模糊查询。样例演示如图6.13所示。可以看到在只输入了一个“迅”字的情况下，浏览器便返回了书名或者作者名中有“迅”字的书籍。

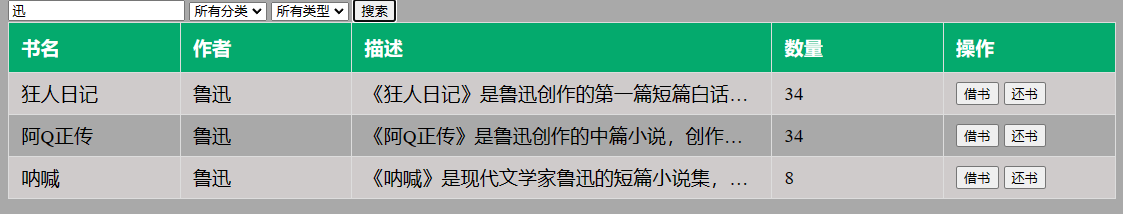
* 1. 管理员管理书籍界面

图6.12 搜索页面

图6.13 模糊词汇搜索

* + 1. 书籍添加功能

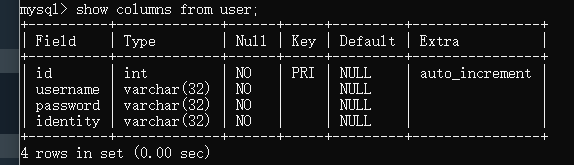
首先我们先观察一下用户数据表，如图6.14所示。可以看到该表除了存储用户的用户名和密码外还存储了用户的身份信息。而管理员管理书籍的界面是只有身份是“管理员”的用户才能进入的。管理员能在该页面内进行书籍的添加和删除操作。该界面与搜索界面类似，不同的地方在与浏览器提供的操作类型。如管理员项进行“添加书籍”的操作则浏览器会展示出一个弹窗如图6.15所示。

图6.14 用户数据表

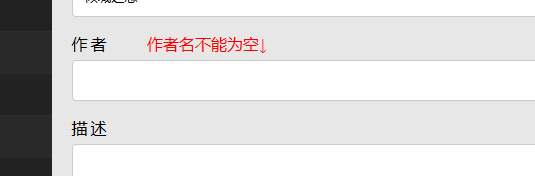
从图中可以看到弹窗左上角会提示管理员当前正在进行的操作。管理员可以在输入框内输入书籍的相关信息。当管理员输入的信息不符合规范时，浏览器也会给出相应的提示，比如作者名不符合规范的提示如图6.16所示。

图6.15 添加书籍

图6.16 空缺信息提醒

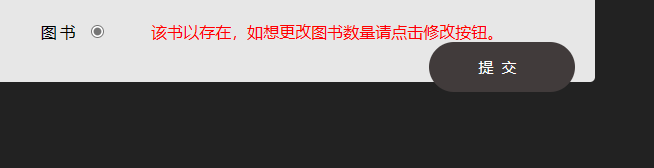
最后当管理员输入完信息并提交按钮浏览器也会给出对应的提示信息，分别如图6.17和图6.18所示。

图6.17 重复信息提醒

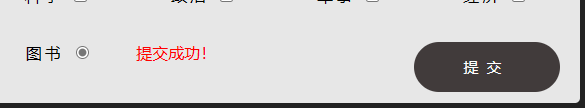
* + 1. 书籍修改功能

图6.18 图书信息提交

点击对应书籍的修改按钮，浏览器同样会有一个对应书籍信息的弹窗，如图6.19所示。可以在左上角看到管理员当前正在进行的操作为修改数据，并且输入框内以自动填充好了对应书籍的相关信息以便管理员进行修改

图6.19 数据信息修改

1. 结语

本系统是使用java技术实现的，对于自身而言很是困难，所以在这个过程中基本处于一边学习一边设计的情况，通过观看各种java、web前端开发视频来进行学习并设计。尽管如此还是有许多不足，除了程序本身还是有很多不足，自身的基础知识也不是很牢固。程序自身还有很多繁琐并且重复的地方，由于编写前期受到视频的影响将前端代码写到的后端，采用了输出流的方式，使得程序异常繁琐，导致添加功能时期的进度十分缓慢，对自己造成了很大的困难，走了很多弯路。经过这段时间的努力，对java编程、web前端开发和mysql数据库有了更深的理解，对自己以后的学习工作道路大有裨益。

通过开发这个设备管理系统学到了很多java全栈知识，例如springboot框架、git分支、数据库、前端等等知识点，使我进步了许多，对后端开发有了一个全新的认识，主要是把基础设施代码和业务代码尽可能的分开，各自不要干扰，而且能把BEAN都统一到spring container里面去，这样，bean的生老病死都由spring来管理，开发者就只需要关心业务怎么实现就好了，别一会实现功能，中间还要来段事务处理，后面还要加个数据库错误处理啥的。总而言之一句话，spring解决的问题就是尽可能的业务代码归业务代码，基础设施代码（日志、事务，异常，对外接口......）归基础设施代码，搞定解耦的问题，希望在以后的学习生涯中可以了解更高效的技术，从浅入深，环环相扣，每一步都会对照着官方文档结合自己的见解进行讲解，同时也会编码实现，理论与实践相结合。

**参 考 文 献**

* 1. WOLF H. Austrian Library Network and Next Generation Library System: Alma [J]. Bibliothek Forschung und Praxis, 2016, 9(10): 341-346.
  2. YAHIA Z.M. Library expense control: a system dynamics approach [J]. TheElectronic Library, 2012, 9(10): 492-506.
  3. JOY K. How to work with Sessions in ASP.Net [J]. Info World.com, 2016, 9(10):123-126.
  4. DING D., RICH C. Building a SQL Server Inventory Report [J]. SQL Server Magazine, 2010, 5(10): 35-39.
  5. ADAM F., STEVEN S. Pro ASP.NET MVC 3 Framework [M]. 3rd ed. Apress, Beijing: The People’s Posts And Telecommunications Press, 2013.