

1 引言

1.1 课题背景

随着互联网技术的快速发展，图书管理系统已经成为了图书馆、书店等机构不可或缺的一部分。它能够帮助管理员更好地管理图书信息，方便读者查找和借阅图书。在过去，许多图书管理系统都是基于传统的客户端/服务器架构开发的。这种架构虽然稳定可靠，但也存在一些问题。例如，它需要安装专门的客户端软件，并且难以扩展和维护。

为了解决这些问题，越来越多的开发人员开始使用 Web 技术来开发图书管理系统。Web 技术具有跨平台、易于访问和维护等优点。它能够让用户通过浏览器直接访问系统，无需安装任何客户端软件。Vue 和 Node 是两种非常流行的前端和后端开发技术。Vue 是一种渐进式 JavaScript 框架，它能够帮助开发人员快速构建交互式用户界面。Node 则是一种基于 Chrome V8 引擎的 JavaScript 运行时环境，它能够让开发人员使用 JavaScript 来编写服务器端程序。^[1]结合 Vue 和 Node，我们可以开发出一套高效、易用、可扩展的图书管理系统。该系统能够满足管理员和读者的需求，并且具有良好的用户体验。首先，在前端方面，我们可以使用 Vue 来构建一个清晰、简洁、易用的用户界面。Vue 提供了许多强大而灵活的功能，例如组件化、数据绑定、计算属性等。这些功能能够帮助我们快速实现各种复杂交互效果。其次，在后端方面，我们可以使用 Node 来搭建一个高性能、可扩展、易于维护的服务器程序。Node 具有非常高效地处理 I/O 操作（如网络请求）的能力，这对于图书管理系统这类需要处理大量数据的应用来说非常重要。^[2]此外，Node 还提供了丰富的模块和包，可以帮助我们快速实现各种功能。在数据库方面，我们可以使用 MongoDB 等 NoSQL 数据库来存储和管理图书信息。NoSQL 数据库具有高性能、可扩展、易于使用等优点。它能够帮助我们快速实现复杂的查询和统计功能。

总之，基于 Vue+Node 的图书管理系统具有许多优点。它能够提供良好的用户体验，并且具有高效、可扩展、易于维护等特点。在未来，随着 Web 技术的不断发展，这类系统将会得到更广泛的应用。

1.2 国内外研究现状

1.2.1 图书管理系统

图书管理系统是指利用计算机技术和网络技术，对图书馆的各项业务进行自动化

管理的系统。图书管理系统的发展与研究现状主要包括以下几个方面。

1.2.2 国外图书管理系统的发展与研究现状

发达国家的电子技术产业早在上个世纪 70 年代后期就开始研究信息检索技术。之后,信息检索、自动分类和自动索引等技术的研究工作陆续开始,推动了图书馆信息管理系统的发展。当前,在国外图书馆的管理系统中,ExLibris 公司开发的 Aleph500 系统和澳大利亚 DYNIX 公司开发的 Horizon 系统比较受欢迎。这两个系统都开始研究图书馆信息技术很早,Aleph500 系统是 20 世纪 80 年代初期面世,Horizon 系统是在 80 年代中期推出。这些国外的图书馆管理系统,特别是 Aleph 和 Horizon,在信息检索、分类编目、公共服务等方面,都已经形成相对成熟的产品和服务。它们在海外图书馆,特别是研究型图书馆中得到广泛的应用,对海外图书馆的自动化和网络化起到了很大的推动作用。这些系统具有高度集成、高度可定制、高度互操作、高度智能等特点,能够满足不同类型和规模图书馆的需求。

1.2.3 国内图书管理系统的发展与研究现状

我国的图书馆管理系统起步于上个世纪 80 年代初,经过 30 年的发展,现已取得了长足的进步。目前,我国绝大部分图书馆已经将计算机技术应用于图书管理当中。图书馆管理已经发展到计算机系统的自动化、网络化和数字化操作模式。80 年代初,我国图书馆信息技术还处于初始阶段,计算机仅用于少数图书馆的局部业务处理。经过 30 年发展,信息技术已经广泛用于我国各类型图书馆的管理运营。大多数图书馆实现了自动化管理,读者可以查询馆藏资源和进行预约等操作;许多图书馆实现了网络化,与外部数据库和其他图书馆实现了互联互通;还有部分图书馆的藏书资源已经实现电子化,可以为读者提供数字资源的访问服务。我国图书馆信息技术发展迅速,自动化水平和网络化程度不断提高,部分图书馆的数字化建设也在积极推进。这使得我国的图书馆管理更高效便捷,为读者提供更丰富的服务,已经成就了一定的成就,这是值得肯定的。同时,也需要看到,总体上我国图书馆的信息技术与发达国家相比还存在差距,还需要不断学习和改进。^[3]该系统采用了日志文件技术,保障了数据一致性和安全性。

(2) 汉王智能卡片式数字化阅览室解决方案。汉王智能卡片式数字化阅览室解决方案是一种基于智能卡片技术和无线网络技术实现数字资源共享和阅读服务的新型解决方案。该方案通过智能卡片实现读者身份认证和权限控制,并通过无线网络将服务器上存储的数字资源传输到读者手中持有的智能卡片上进行阅读。(3) 超星数字资源平台(DRM)。超星数字资源平台(DRM)是一个基于 Web 服务架构设计并实现的

分布式数字资源服务平台。该平台提供了从数字资源获取、加工、存储到检索、浏览、下载等全过程 的服务，并支持多种格式 和类型 的数字资源。

1.2.4 未来图书管理系统可能存在问题及其对策

随着社会的发展和科技的进步，图书管理系统也需要不断地更新和完善，以适应读者的需求和图书馆的管理。未来图书管理系统可能存在以下问题及其对策：

系统安全性问题。随着网络技术的普及，图书管理系统也需要与互联网相连，以提供更多的服务和功能。^[4]但是，这也带来了一些安全风险，如黑客攻击、病毒感染、数据泄露等。为了保证系统的安全性，需要采取以下措施： 建立完善的防火墙和加密机制，防止外部入侵和数据窃取。定期备份数据，并存储在安全的地方，以防数据丢失或损坏。加强用户身份认证和权限管理，避免非法操作或误操作。增加系统日志记录和审计功能，监控系统运行状况和异常事件。 系统功能不足问题。随着读者需求的多样化和个性化，图书管理系统也需要提供更多的功能和服务，如在线预约、电子资源、智能推荐等。但是，这些功能可能超出了现有系统的设计范围或技术水平，导致系统功能不足或不稳定。为了解决这个问题，需要采取以下措施： 调查分析读者需求和满意度，确定优先开发或改进的功能模块。引入新技术或新平台，如人工智能、云计算、移动端等，扩展系统功能和服务范围。定期更新维护系统软件和硬件设备，保证系统性能和稳定性。^[5]系统资源利用率低问题。随着图书馆藏量的增加和信息化建设的推进，图书管理系统需要处理更多的数据和信息。但是，如果没有合理地配置资源或优化算法，则可能导致资源浪费或效率低下。

为了提高资源利用率和效率，则需要采取以下措施： 根据业务特点和数据特征进行合理地数据库设计与优化，并使用索引、分区等技术提高查询速度。 根据负载情况进行动态地资源分配与调整，并使用缓存、负载均衡等技术提高响应速度。根据用户行为进行智能地数据分析与挖掘，并使用推荐、分类等技术提高服务质量。^[6]

1.3 论文的结构安排

本论文共分为 6 章。

第一章：引言，首先阐述图书管理系统设计的背景、再介绍本课题现状的调研，最后介绍本文的结构安排。

第二章：对课题中用到的工具和技术进行简单的介绍。包含 Vue、Node、Koa 所用到的一系列技术。本章围绕技术的基础概念、特点、适用场景等维度展开介绍。

第三章：需求分析，分析系统的功能性需求及系统的其他需求。

第四章：图书管理系统的概要设计，设计基于 Vue+Node 的图书管理系统的设计以及 MongoDB 的集合文档设计

第五章：根据需求分析和概要设计内容，对图书管理系统进行各个功能模块的详细设计与实现。

第六章：系统测试

第七章：对课题和论文的总结。及对未来工作的展望和致谢。

2 相关技术和工具

在设计一个系统的过程中，对系统使用的技术和工具进行充分的研究是实现一个系统的前提，本章主要介绍了系统使用的语言、技术和工具。本项目主要针对目前热门的 Vue3 和 Node 后端框架进行研究。^[7]其中包括 Vue3 前端渐进式开发框架、Node 后端环境、Koa2 后端框架、MongoDB 非关系型数据库等相关工具和技术。

2.1 Vue3

Vue 3 是一款流行的 JavaScript 前端框架，它带来了很多新功能和改进，使得开发者能够更加高效地构建交互式 web 应用程序。

以下是 Vue 3 的一些主要技术框架：Vue CLI 是一个命令行工具，可以帮助你快速地创建、管理和构建 Vue 项目。它提供了一个模板和插件系统，可以让你轻松地扩展和自定义项目。Vuex 是一个状态管理库，用于管理 Vue 应用程序中的状态。它提供了一个集中化的存储，可以跨组件共享状态，使得应用程序的状态管理更加简单和可维护。^[8]Vue Router 是一个官方的路由管理库，用于管理应用程序中的路由。它提供了一种简单而灵活的方式来定义和处理路由，使得应用程序的导航更加直观和流畅。Composition API 是 Vue 3 中的一项新功能，它提供了一种更加灵活和可组合的方式来编写组件逻辑。它可以帮助开发者更好地组织和重用代码，使得应用程序更加易于维护和扩展。Teleport 是 Vue 3 中的另一项新功能，它提供了一种灵活的方式来在组件之间传输内容。它可以帮助开发者更好地控制 DOM 结构和组件关系，使得应用程序更加易于扩展和重构。

2.2 Vite

Vite 是一个基于 ES modules 的快速开发工具，它被设计为用于现代化的 web 应用程序开发。以下是一些 Vite 的技术特点：

快速的冷启动：Vite 的开发服务器采用了类似于 Snowpack 的技术，可以在几毫秒内启动，使得开发者可以更快地进行开发和调试。**基于 ES modules：**Vite 基于 ES modules 构建，可以通过浏览器原生支持的方式加载模块，避免了传统构建工具中的打包和转换过程，可以提高开发的效率和性能。^[9]**零配置开发：**Vite 支持零配置开发，开发者可以不用进行繁琐的配置，直接使用默认的配置即可进行开发。**插件化架构：**Vite 的插件化架构可以让开发者方便地扩展 Vite 的功能，例如添加 TypeScript 支持、CSS 预处理器等功能。**支持 Vue 3：**Vite 对 Vue 3 提供了深度

支持，可以快速构建和调试 Vue 3 应用程序。总之，Vite 提供了一种快速、现代化的开发体验，可以帮助开发者更快地构建和调试现代化的 web 应用程序。

2.3 Node

Node.js 是一个基于 Chrome V8 引擎的 JavaScript 运行时，能够让 JavaScript 代码在服务器端运行，使得 JavaScript 可以脱离浏览器环境，应用于服务器端的开发。^[10]Node.js 提供了丰富的 API 库，使得开发者能够方便地进行网络应用程序的开发，包括但不限于 Web 应用程序、API、后端服务等等。

以下是 Node.js 技术的一些常见应用：服务器端开发：Node.js 可以用于构建服务器端应用程序，例如 Web 应用程序、API、即时通讯服务等等。前端开发工具：Node.js 可以用于搭建前端开发的自动化构建工具、包管理器等等。数据库访问：Node.js 可以用于访问多种数据库，包括 MySQL、MongoDB、PostgreSQL 等等。实时通讯：Node.js 可以用于实现 WebSocket、Socket.IO 等实时通讯协议和应用程序。命令行工具：Node.js 可以用于开发命令行工具，例如 npm、gulp、webpack 等等。机器学习和人工智能：Node.js 可以与 TensorFlow.js 和 Brain.js 等机器学习框架配合使用，实现机器学习和人工智能应用。总的来说，Node.js 技术具有高效、可伸缩性强、易于学习和应用广泛等优点，已经成为了现代 Web 开发不可或缺的技术之一。

2.4 Koa

Koa 是一个基于 Node.js 平台的 Web 开发框架，它由 Express 的原班人马打造，旨在提供一种更简单、更强大、更具表现力的 Web 开发体验。Koa 的特点是使用了 ES6 的 async/await 特性，使得异步代码的编写和错误处理更加优雅，同时也更加符合 JavaScript 开发者的编码习惯。^[11]

以下是 Koa 技术框架的一些特点：轻量级：Koa 的核心代码只有不到 600 行，它不依赖任何第三方库，所以非常轻量级。中间件机制：Koa 采用了一种灵活的中间件机制，允许开发者通过串联多个中间件函数，处理各种 Web 请求和响应，这样代码的组织和重用都变得非常简单。异步处理：Koa 采用了 ES6 的 async/await 特性，使得异步代码的编写和错误处理更加优雅。错误处理：Koa 非常重视错误处理，它提供了一种统一的错误处理机制，使得代码的健壮性得到了很大提升。路由功能：Koa 本身并没有路由功能，但是可以使用 koa-router 这个第三方中间件库，非常方便地实

现路由功能。总的来说，Koa 技术框架是一种优雅、简单、可靠的 Web 开发框架，适合用于构建各种 Web 应用程序，特别是对于需要高度定制化的场景，Koa 的灵活性和可扩展性非常有优势。

2.5 MongoDB 数据库

MongoDB 是一个基于分布式文件存储的开源数据库，它采用了面向文档的数据模型（Document-Oriented），以 BSON（Binary JSON）格式存储数据，支持丰富的数据查询语言和索引机制，同时具备高度可扩展性、高性能、高可用性等优点。

以下是 MongoDB 数据库的一些特点：面向文档：MongoDB 的数据模型是面向文档的，每个文档都是一个键值对（Key-Value）的 JSON 对象，文档可以嵌套文档和数组，非常灵活。^[12]分布式文件存储：MongoDB 的数据存储是分布式的文件存储方式，支持水平扩展和负载均衡，可以存储海量数据。查询语言和索引机制：MongoDB 支持丰富的数据查询语言和索引机制，包括多种查询方式和索引类型，如全文索引、地理空间索引等等。高度可扩展性：MongoDB 支持水平扩展，可以在多个服务器上部署数据库，构建高可用性的集群环境，同时也可以很方便地添加新的节点和扩展存储容量。高性能：MongoDB 使用了一些高效的技术，如内存映射文件、写时复制等，以提高数据库的性能。

总的来说，MongoDB 是一种非常强大、灵活、可扩展、高性能的数据库系统，适合用于各种规模的应用程序，特别是对于需要处理大量文档型数据的场景，MongoDB 的优势非常明显。

2.6 Axios 前后端通信

Axios 是一个基于 Promise 的 HTTP 客户端，它可以在浏览器和 Node.js 中使用，用于发送 HTTP 请求和处理响应数据。Axios 可以支持多种请求类型和数据格式，可以设置请求参数、请求头等，同时还提供了一些高级特性，如拦截器、取消请求、自动转换请求和响应数据等。

下面是 Axios 的一些特点：基于 Promise：Axios 是基于 Promise 的异步编程模型，支持链式调用，可以使用 then、catch、finally 等方法处理响应结果，使得代码更加清晰易懂。支持多种请求类型：Axios 支持 GET、POST、PUT、DELETE 等多种请求类型，可以通过传递请求参数和请求头来进行设置。支持多种数据格式：Axios 支持多种数据格式，如 JSON、FormData、URLSearchParams 等，可以根据实际需求

选择适合的数据格式。^[13]拦截器：Axios 提供了请求和响应的拦截器，可以在请求发送前和响应返回后进行拦截处理，如设置请求头、处理错误等。自动转换请求和响应数据：Axios 可以自动将请求和响应数据转换为 JSON 格式或其他格式，简化了开发过程。取消请求：Axios 提供了取消请求的功能，可以在请求还未完成时取消请求，避免不必要的资源浪费。

总的来说，Axios 是一种非常方便、灵活、可靠的 HTTP 客户端工具，可以帮助开发者更加轻松地发送 HTTP 请求和处理响应数据，适用于各种场景，特别是对于需要频繁发送 HTTP 请求和处理响应数据的场景，Axios 的优势非常明显。

2.7 npm 包管理工具

npm 是 Node.js 的包管理工具，用于安装、发布、升级和管理 Node.js 模块和包。通过 npm，开发者可以方便地获取、分享和使用各种 Node.js 模块和包，大大提高了开发效率和代码重用性。^[14]

下面是 npm 包管理工具的一些特点和用法：npm 提供了一种便捷的方式来管理 Node.js 模块和包，开发者可以使用 npm 安装、卸载、更新和查找模块和包，也可以创建自己的模块和包，并发布到 npm 中心仓库。版本管理：npm 使用语义化版本号规范来管理模块和包的版本，开发者可以通过指定版本号范围来控制依赖关系和兼容性，也可以使用 npm 提供的命令来管理和升级版本号。依赖管理：npm 使用 package.json 文件来管理模块和包的依赖关系，开发者可以在 package.json 文件中指定依赖的模块和包，也可以使用 npm 提供的命令来安装、卸载和更新依赖。模块查找：npm 提供了一个庞大的模块和包仓库，开发者可以通过 npm 查找并安装需要的模块和包，也可以通过配置自定义仓库来管理自己的模块和包。^[15]全局安装：npm 支持全局安装模块和包，使得这些模块和包可以在任何地方使用，开发者可以通过 npm 提供的命令来安装和管理全局模块和包。

总的来说，npm 是一个非常重要、强大、易用的包管理工具，可以大大提高开发效率和代码重用性，也可以方便地管理和分享 Node.js 模块和包，是 Node.js 生态系统中不可或缺的一部分。

3 需求分析

3.1 系统面向的用户群体分析

（1）图书馆的管理者：

该系统主要是帮助图书馆管理者进行全面有效的管理入库、出库和书籍信息。

3.2 系统功能需求分析

3.2.1 图书馆的管理者

（1）管理者登录、注册模块

该图书管理系统的用户分为超级管理员和普通操作员。操作员可以维护各种图书和书籍信息,以及管理库存。超级管理员除了拥有操作员的权限外,还可以进行系统管理。用户登录系统时需要输入正确的用户名和密码。系统会根据用户名和密码确认用户身份,然后赋予相应的权限。如果用户名和密码输入错误,用户会停留在登录界面。超级管理员拥有最高权限,可以管理整个系统。普通操作员只能进行日常业务操作,如图书和书籍信息维护、库存管理等。管理员的登录用例图如图 3-1 所示:

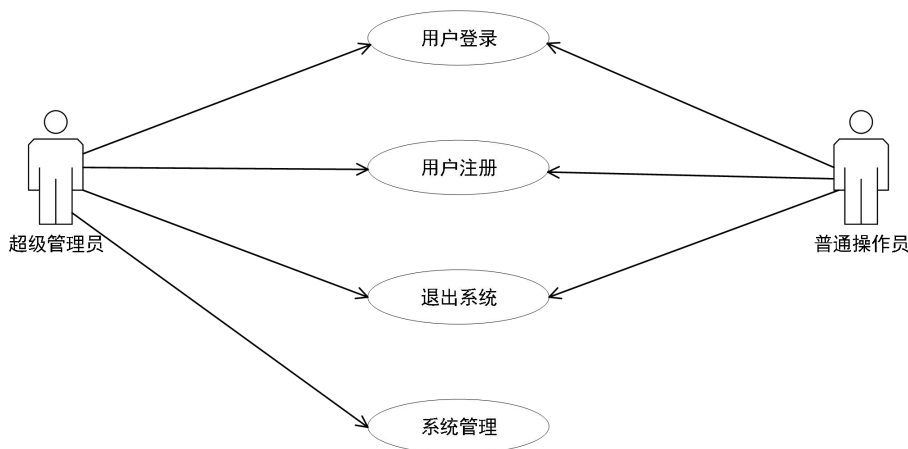


图 3-1 管理员的登录用例图

（2）管理者系统管理模块

管理员可以对书籍进行添加、删除、修改,对普通用户进行添加用户、删除用户、重置用户密码。对日志列表进行删除,对书籍分类进行添加、修改、删除。管理员的管理功能用例图如图 3-2 所示:

① 总览信息

实时了解书籍的数量,用户数量,日志条数,以及最近添加书籍,最近的操作日

志。

② 图书管理：

可以根据书籍名称进行搜索、添加书籍信息，包含书籍的书名、分类、作者、出版时间、折扣、价格、库存、操作还可以查看书籍详情，对书籍信息进行编辑书名、价格、作者、出版日期、分类也可以对书籍进行删除。^[7]详情页可以显示书籍的书名、价格、作者、分类、生产日期、出入库日志信息，点击入库可筛选出入库日志，点击出库可筛选出出库日志信息。下方显示日志的信息，包含数量和时间，最后是分页功能，点击可以进行切页。

③ 用户管理：

点击用户管理可以跳转到用户管理界面。最上方的搜索框可以输入账户名称进行搜索，搜索出用户信息，点击右方添加用户，可以显示一个弹窗添加用户的表单，可以输入用户、密码、选择角色是成员还是管理员。下方是用户列表，包含账户、角色、创建日期、操作，其中角色中可以进行修改，将用户的角色改为成员或管理员，操作一栏可以进行重置密码，或者进行删除用户。上传 Excel 添加：批量添加用户的方式，可以通过 Excel 批量上传用户信息。

④ 操作日志：

点击操作日志会进入到操作日志界面，操作日志包含日志列表，包括有用户名、动作、记录时间、操作等列，用户名记录着哪个用户进行的操作，动作则是进行操作的内容，记录时间即时操作时间，操作一栏有删除，可以对日志进行删除。最后有分页功能。

⑤ 重置密码列表：

点击可跳转至重置密码列表界面，该界面包含有关重置密码的信息，其中包括用户账户和操作选项。用户账户信息显示在账户部分，而操作选项包括“重置”和“忽略”按钮。当用户点击“重置”按钮时，该账户的密码将被重置。点击忽略则忽略该账户的重置密码申请。

⑥ 书籍分类管理：

点击可以调到书籍分类管理模块，管理员可以对书籍分类进行添加新的分类名称，下方是书籍分类列表，可以进行修改分类名称和删除分类。如图 3-2 书籍管理。

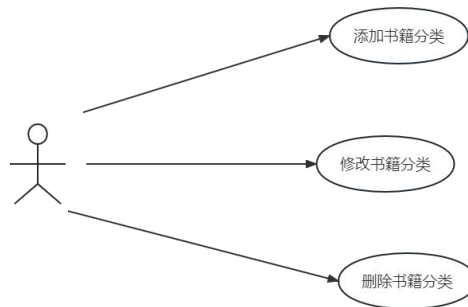


图 3-2 书籍管理

⑦ 邀请码列表：

管理员可以创建邀请码，且邀请码只能被一个用户使用，下方是邀请码的列表，包含邀请码、是否使用、操作，操作可以进行删除邀请码。

⑧ 个人设置：

个人设置里可以对密码进行修改，输入旧密码，新密码。

3.3 系统性能需求分析

对系统的性能需求分析影响到系统建成后用户的体验，即是否方便快捷的进行管理，各部门工作人员处理工作的效率是否提高，安全性是否有所保障。对该管理系统的性能需求分析主要体现在以下几个方面：

（1）系统可扩展性/可伸缩性

该系统需要有很好的可扩展性,即适应变化的能力。一旦用户需求发生改变,系统应能通过版本更新或功能扩展和调整来满足用户的新需求,而保持系统的主体结构不变。可扩展性是评价一个系统质量的重要指标。它反映了系统适应变化和新增需求的灵活性。对于该系统来说,具有良好的可扩展性意味着:一方面,系统有模块化和层次化的结构,功能和数据能够很容易地重新组织和调整,以支持用户需求的变化。系统不是一个既定的整体,而是由许多相互连接但相对独立的组件构成,这使其易于变更和扩展。另一方面,系统有清晰的接口,外部系统和新功能很容易接入,而不影响系统内部分。这使得系统有较强的兼容性和适应性,能够满足交互性和开放性的要求。

（2）系统健壮性

该系统需要有处理非标准输入的能力,以确保系统的正常运行。系统采用现代的体系结构、全面严密的安全措施以及标准化的应用平台。参考各种网站标准,确保系统的可靠性。系统应具备托管异常输入的能力。也就是说,对于用户输入的参数超出标准范围或不符合系统要求的情况,系统必须有相应的校验和处理机制,而不会直接

中断或崩溃。这需要系统具有较强的容错性和兼容性,能够处理格式错误、越界值等输入异常情况。系统架构采用现代化设计,具有模块化、松耦合、容错、伸缩等特征。这使得系统更加稳定可靠,不容易受到局部变更或异常情况的影响。安全机制全面严密,可以防范注入、DoS 等攻击,保障系统安全。

（3）简单易用、界面友好性

该系统的使用要简单和人性化。要求界面友好美观、易学易用,操作性强,能促进人机交互,给用户体验舒适、大方的感觉。同时适应不同操作平台,并能高效快速地处理数据。简单易用是评价一个系统最重要的指标之一。它体现在用户界面友好、交互效率高以及易于学习掌握等方面。该系统的界面要求美观大方,不会给用户太过复杂或乱象的视觉体验。同时界面简洁易学,用户很快可以上手使用,不需要太长时间的学习和记忆。高效的人机交互可以大大提高用户体验。系统要响应迅速,执行任务流畅。用户的每次输入都会得到即时反馈,不会有卡顿或无响应的情况出现。这要求系统具有高效的数据处理能力和稳定的性能。跨平台兼容是现代软件系统的基本要求之一。系统应支持主流桌面和移动操作系统,并根据不同终端优化交互方式和界面展示,提供最佳的用户体验。这使更多用户有机会接触和使用该产品。

4 系统设计

4.1 概要设计

（1）用户界面层(Vue.js) 这一层负责呈现和交互 UI 元素，也被称为前端层。Vue.js 可以将 UI 界面划分成不同的组件，从而实现代码的重用性和扩展性。^[9]通过 Vue Router 进行路由管理，向后端 API 发送请求来获取数据或执行其他操作。

（2）应用程序层(Koa.js) 这一层位于前端与后端之间，也被称为后端层。应用程序层主要是通过 Koa.js 作为 Web 应用程序框架来提供 API 服务，将前端发来的 API 请求进行处理、验证、授权等，并可以调用底层的存储服务来获取或更新数据。

（3）存储层 这一层包含了数据库或者其他的存储系统，负责保存和管理数据，为应用程序层提供数据支持。

系统架构中，这三层之间通过 RESTful API 协议进行交互。通过将系统功能分配到不同的层次，可以提高系统模块之间的松耦合性，增加维护和扩展性，并为应用程序添加新功能提供了良好的框架。

总之，图书管理应用程序的完整架构由前端层、应用程序层和存储层组成，旨在提供最佳的用户界面、及最高质量的软件架构，并且可以快速，安全地在 Koa.js 和 Vue.js 的技术平台上进行开发和部署。

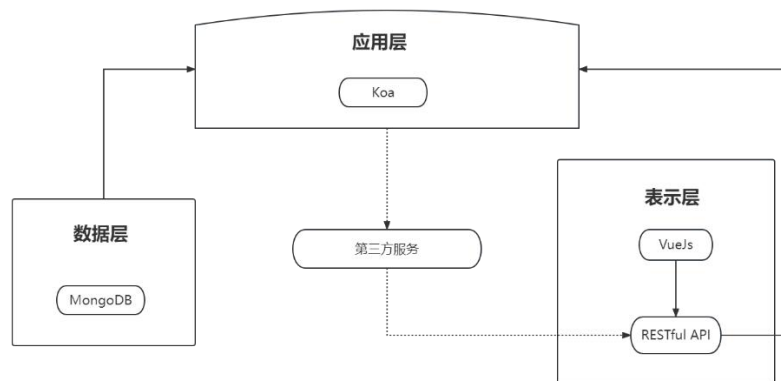


图 4-1 系统架构部署图

（1）功能模块设计

① 超级管理员登陆模块

超级管理员可以通过输入用户名和密码登陆系统，超级管理员账户是系统的最高权限账户。设置超级管理员的数据库表，包含用户名、密码等字段，通过 bcrypt 等加密方式保证账户安全。在登陆页面进行账户验证，提供忘记密码等功能。

② 普通管理员注册登陆模块

普通管理员需要通过超级管理员的邀请码注册账号，并可以通过输入用户名和密码登陆系统，普通管理员账号的权限由超级管理员进行授权。设置邀请码数据库表，包含邀请码、使用情况、生成时间等字段，超级管理员可以使用该表进行邀请码的生成和管理。设置普通管理员的数据库表，包含用户名、密码、权限等字段，通过 bcrypt 等加密方式保证账户安全。在注册页面进行邀请码验证，提供忘记密码等功能，普通管理员账号的权限由超级管理员进行授权。

③ 邀请码模块

超级管理员可以通过邀请码来授权普通管理员账号的创建和使用。

④ 总览模块

显示系统的总体情况信息，包括用户数量、书籍数量、管理员数量等等。设置用户、书籍、管理员的统计数据库表，在总览页面进行查询并展示。

⑤ 书籍管理模块

管理员可以添加、编辑和删除书籍信息，包括书名、作者、出版社、价格等字段。设置书籍的数据库表，包含书名、作者、出版社、价格等字段，通过数据库操作进行添加、编辑、删除等操作。在前端页面通过表格等形式展示书籍信息，提供搜索、排序等功能。

⑥ 用户管理模块

管理员可以查看用户列表并进行编辑、删除等操作，包括昵称、账户名、密码等字段。设置用户的数据库表，包含昵称、账户名、密码等字段，通过数据库操作进行编辑、删除等操作。在前端页面通过表格等形式展示用户信息，提供搜索、排序等功能。

⑦ 操作日志模块

记录管理员的操作日志，包括操作类型、操作者、操作时间等字段。设置操作日志的数据库表，包含操作类型、操作者、操作时间等字段，并通过数据库操作进行记录和查询。在前端页面通过表格等形式展示操作日志信息，提供搜索、排序等功能。

⑧ 个人设置模块

管理员可以进行个人信息的修改，包括昵称、密码等字段。在管理员登陆后，在前端页面提供个人信息修改的入口，通过数据库操作进行修改。在前端页面提供表单等形式进行信息的输入和修改确认。在实现过程中，需要注意数据库表的设计、页面设计、接口设计以及安全性设计。对于数据敏感的操作，如管理员登录、个人信息修

改等，需要进行数据加密保护。

4.2 数据库设计

(1) 数据库概念结构设计

本项目的 E-R 图如图 4-2 所示：

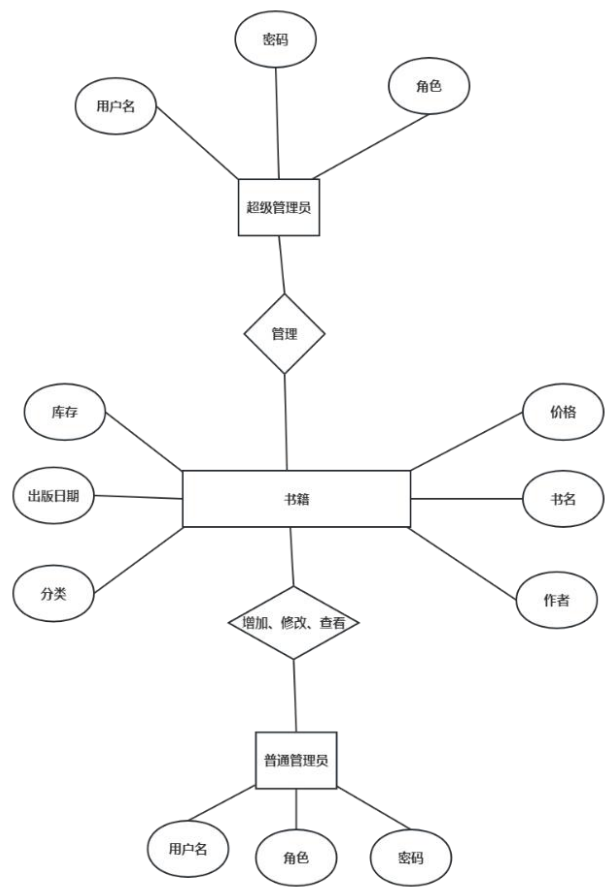


图 4-2 E-R 图

4.2.1 数据库关系表设计

通过 E-R 图及系统自身的业务功能需求分析，将该系统的数据文档划分为用户文档、日志文档、邀请码文档、出库入库日志文档、忘记密码申请文档、用户角色文档、书籍文档、书籍分类文档。^[10]本节主要介绍了这些文档的设计以及文档中每个字段的类型和长度的信息，具体文档的设计如下所示：

(1) 用户文档，该表用于普通管理员和超级管理员的登录、注册以及邀请码的管理，具体设计如表 4-1 所示：

表 4-1 用户表（users）

序号	列名	类型	含义	备注
----	----	----	----	----

1	_id	int	管理员编号	主键
2	account	varchar(20)	管理员名称	
3	password	varchar(20)	管理员密码	
4	character	varchar(20)	管理员权限	
5	meta	Object	元信息	

（2）日志文档，该表显示管理员操作的日志，包括增加用户，增加书籍，更改书籍信息等操作，具体设计如表 4-2 所示：

表 4-2 日志表(logs)

序号	列名	类型	含义	备注
1	_id	ObjectId	文档 id	
2	user	Object	用户	
3	request	Object	请求方式	
4	meta	Object	元信息	

（3）邀请码文档，该表管理员对邀请码的添加和删除、使用。具体设计如表 4-3 所示：

表 4-3 邀请码(invitecodes)

序号	列名	类型	含义	备注
1	_id	ObjectId	文档 id	主键
2	code	String	邀请码	
3	user	Object	使用邀请码的管 理员	

（4）出库入库文档，用于展示管理员对书籍的出库入库操作。具体设计如表 4-4 所示：

表 4-4 出库入库(inventorylogs)

序号	列名	类型	含义	备注
1	_id	ObjectId	文档 id	主键
2	type	String	出库/入库	
3	num	Int32	数量	

4	meta	Object	元信息
---	------	--------	-----

（5）忘记密码文档，用于管理员对忘记密码操作的用户进行相关操作，具体设计如表 4-5 所示：

表 4-5 忘记密码(forgetpasswords)

序号	列名	类型	含义	备注
1	_id	ObjectId	文档 id	主键
2	Account	String	用户名	
3	Status	Int32	状态	
4	Meta	Object	元信息	

（6）角色信息文档，显示管理员的角色，是普通用户管理员还是超级管理员，具体设计如表 4-6 所示：

表 4-6 角色信息(forgetpasswords)

序号	列名	类型	含义	备注
1	_id	ObjectId	文档 id	主键
2	Name	String	角色名称	
3	Title	String	角色中文名称	
4	Power	Object	权限	
5	Meta	Object	元信息	

（7）书籍文档，显示书籍的名称价格、作者、分类、数量等，具体设计如表 4-7 所示：

表 4-7 书籍(forgetpasswords)

序号	列名	类型	含义	备注
1	_id	ObjectId	文档 Id	主键
2	Name	String	书籍名	
3	Price	Int32	价格	
4	Author	String	作者	
5	Classify	String	分类	
6	Count	Int32	数量	
7	Meta	Object	元信息	

（8）书籍分类文档，显示书籍分类的名称，具体设计如表 4-8 所示

表 4-8 书籍分类(bookclassifies)

序号	列名	类型	含义	备注
1	_id	ObjectId	文档 id	主键
2	Title	String	书籍分类名称	
3	Meta	Object	元信息	

5 系统实现

5.1 系统实现环境

系统后端使用 Node.js 开发。Node.js 是一种基于 Chrome V8 引擎的 JavaScript 运行环境,它允许 JavaScript 代码在服务端运行。使用 Node.js 可以提高开发效率,特别适合开发实时应用、流媒体应用等需要高并发的场景。系统运行在 Windows 11 操作系统上。Windows 11 是 Microsoft 公司的最新操作系统,具有更友好的界面、更强的性能和更丰富的功能等优点。MongoDB 是一种文档导向数据库,使用 JSON-like 格式来存储数据。相比 SQL 数据库,MongoDB 的可扩展性和灵活性更强,存储结构更加自由,更适合 Web 2.0 时代的应用。该系统采用 MongoDB 存储各类数据信息。用户使用 Google Chrome 浏览器访问该系统。

5.2 管理员登录注册模块

（1）模块描述

超级管理员要登录该系统,需要提供系统中已有的用户名和密码。如果输入的用户名和密码不一致,超级管理员将无法进入系统,停留在登录页面。只有当用户名和密码均正确时,超级管理员才会被允许进入系统的功能操作界面。超级管理员作为系统的最高权限用户,负责关键配置和管理工作。由于超级管理员拥有较高的操作权限,系统采取双因素认证的方式严格控制超级管理员的登录。用户名和密码都必须正确输入,超级管理员才会被授权访问系统。这是确保系统安全的重要措施。对超级管理员这类高权限用户,系统通常会采取更严格的认证机制。仅凭密码或单一因素是不够的,因为一旦超级管理员账户遭到破解,后果可能严重。系统采用用户名加密码的双因素认证,增强了超级管理员账号的安全性。攻破者需要同时获得用户名和密码,难度更大,提高系统的抗破解能力。在权限控制上,超级管理员具有其他普通用户无法具备的操作权限,因此也面临更高的安全风险。双因素认证 login 机制可以有效遏制超级管理员账号被滥用的可能性,保证只有授权用户才能登录系统并执行管理操作。这是系统权限设计和安全防护的重要举措。新的管理员用户需要通过邀请码注册该系统。注册成功后,新的管理员账户即可用于登录系统。超级管理员是系统的高权限用户,负责关键配置和管理维护工作。由于超级管理员拥有较高的操作权限,系统采取用户名和密码的验证机制严格控制超级管理员的登录。只有当用户名和密码都正确输入时,超级管理员才会被允许登录系统。这是确保系统安全的重要措施之一。对于新的管理员用户,

系统采取邀请注册的方式进行权限控制。只有获得超级管理员提供的有效邀请码,普通用户才能注册管理员账户。邀请注册机制可以过滤恶意注册行为,避免系统出现未授权的管理员账户。注册成功的管理员用户即拥有一定的系统操作权限,可以登录系统执行相关管理工作。

（2）功能实现

登录描述：用户输入用户名和密码后，点击登录，页面将会将用户输入数据通过 ajax 发送到后端，通过匹配管理员登录表中的数据进行判断。登录实现如图 5-1 所示：



图书管理系统后台

登入 注册

账户

密码

忘记密码

登入

图 5-1 管理员登录图

注册描述：普通管理员进行注册时要填写邀请码进行注册，邀请码由超级管理员分发，每个邀请码只能有一位普通管理员注册。然后输入账户和密码，密码不得少于 8 位，由数字字母组成。点击注册即可注册成功新用户。注册实现如图 5-2 所示：



图 5-2 管理员注册图

5.3 管理员后台总览模块

(1) 模块描述

超级管理员是系统中拥有最高权限的用户，可以全面管理整个系统。总览页面是管理员进行图书管理的重要工具之一，可以为超级管理员提供系统的全局概览，展示系统中的书籍数量、用户数量、日志数量、书籍列表和最近操作记录列表等信息。通过总览页面，超级管理员可以了解系统中现有的图书数量和用户数量，进而判断图书资源的丰富度以及系统的用户规模。同时，超级管理员还可以查看系统的日志数量，了解用户行为和系统运行情况，从而进行更加精细的管理和优化。如图 5-3

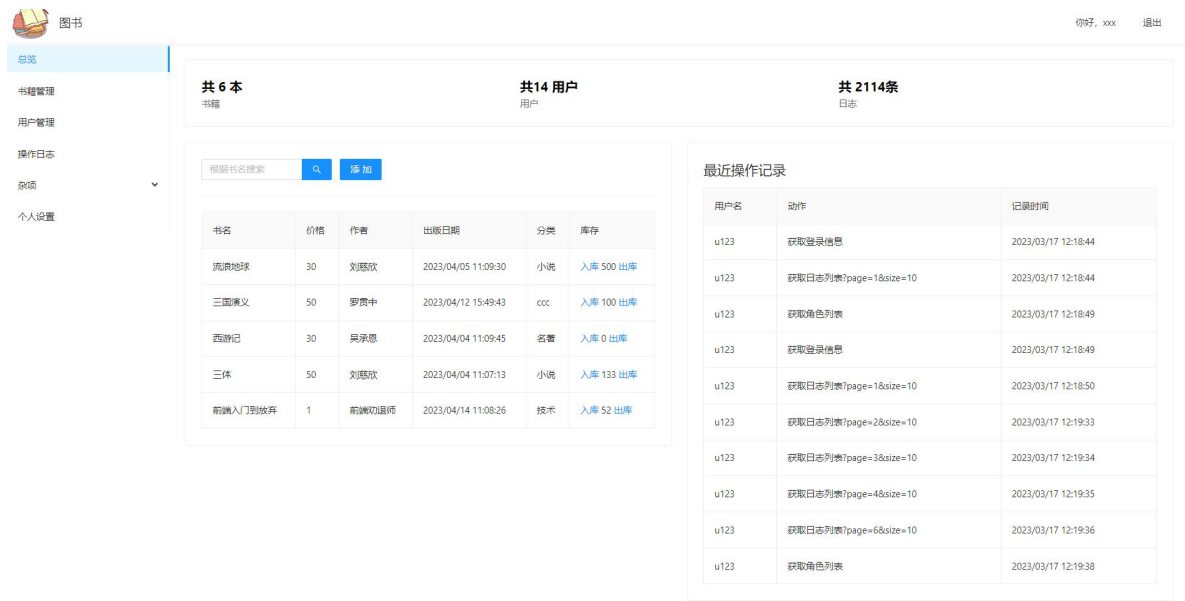


图 5-3 总览页面

(2) 功能实现

总览功能：通过 Koa 框架与 MongoDB 数据库进行交互，分别统计了书籍、用户总数和日志三个集合的数据。通过计算这些集合中各自的总数量，我们可以获得一个全面的数据总览，从而更好地了解和管理数据。最终，我们将这些统计数据通过后端返回到前端进行展示，以帮助用户更好地理解 and 利用这些数据。

书籍列表和日志列表：左边为书籍列表，右边为日志列表。这两个列表分别与书籍管理页面和日志管理页面相关联，作为独立的组件，可以在不同的页面中进行复用。通过这种组件化的方式，我们可以减少代码的重复性和冗余性，提高代码的可读性和可维护性，进而提高开发效率和代码质量。

5.4 管理员书籍管理模块

（1）模块描述

超级管理员可以通过访问特定页面来对图书进行各种操作，包括搜索、添加、删除、编辑等。其中，该页面所呈现的书籍列表是其中一个至关重要的功能。这个列表清单详细记录了每一本图书的书名、价格、作者、出版日期、分类、库存以及操作等关键信息，这些信息的准确性和及时性对于管理员的决策和操作具有极大的影响。此外，在面对庞大的图书库存时，为了提高操作效率，该页面还包括了分页功能，让管理员可以方便地浏览、定位和操作目标图书信息。综上所述，这个页面为超级管理员提供了一个高效、方便、可靠的图书信息管理平台，使得管理员能够更好地管理和维护图书信息，为用户提供更好的服务。如图 5-4

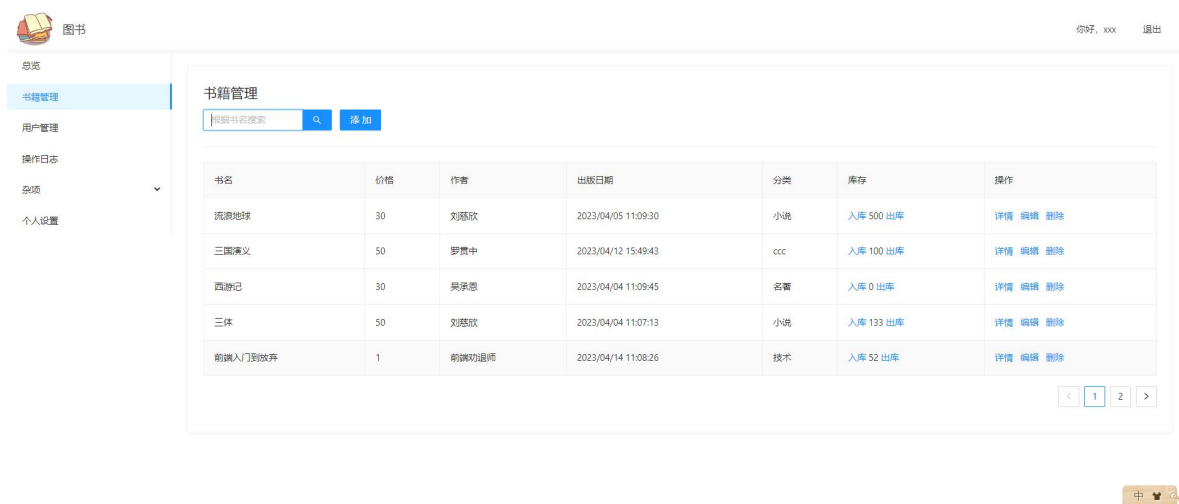


图 5-4 书籍管理

（2）功能实现

添加书籍：管理员可以通过点击添加按钮打开一个书籍表单界面，以便于在系统

中增加新的图书资源。这个表单界面的设计考虑到了管理员填写信息的方便性和操作性，以及数据的准确性和完整性，因此它包含了书名、价格、作者、出版日期、分类、库存等关键信息。管理员需要认真填写每个字段的信息，以保证新添加的图书可以被正确地记录和管理。在填写完毕后，管理员需要点击确定按钮，以便将表单数据通过前端 ajax 技术发送到后端服务器。后端服务器会将这些数据写入到 MongoDB 数据库对应的书籍集合中。这个过程需要确保数据的完整性、一致性和安全性，以保证图书信息管理系统的稳定性和可靠性。总之，这个书籍表单界面是图书信息管理系统的一个重要组成部分，它为管理员提供了一个高效、准确和方便的图书信息添加平台，同时也为用户提供了更加丰富和全面的图书资源。如图 5-5

添加书籍 X

书名:

价格:

作者:

出版日期:

分类:

库存:

Cancel OK

图 5-5 添加书籍

搜索书籍：在输入框里输入书籍名称，然后回车或者点击搜索按钮，就会进行搜索，如图 5-6，前端通过 get 方法发送请求给后端，后端接收到后解构关键字参数，然后进入书籍集合中按照关键字进行查询，找到该数据信息，然后返回给前端。

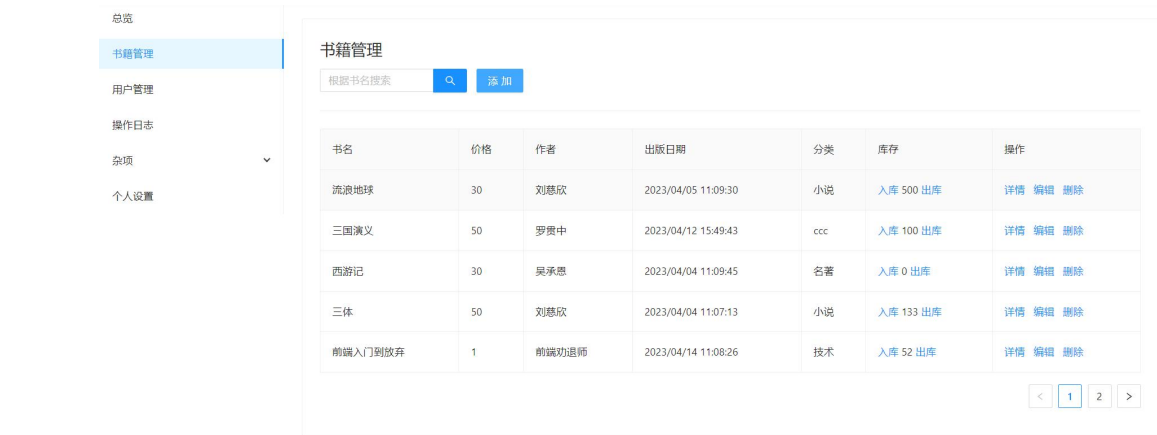


图 5-6 搜索书籍

5.5 管理员用户管理模块

（1）模块描述

其中，超级管理员可以使用搜索功能查找用户账户，添加新用户并设置相应的权限，删除不需要的用户账户，重置用户密码以帮助用户找回自己的账户密码，还可以修改普通管理员的角色信息，确保他们有权访问必要的功能。此外，超级管理员还可以使用 Excel 上传功能快速添加多个用户，以提高管理效率。无论是在添加新用户还是在修改角色信息方面，这个页面都提供了非常方便和高效的工具，帮助超级管理员轻松管理所有普通管理员。如图 5-7

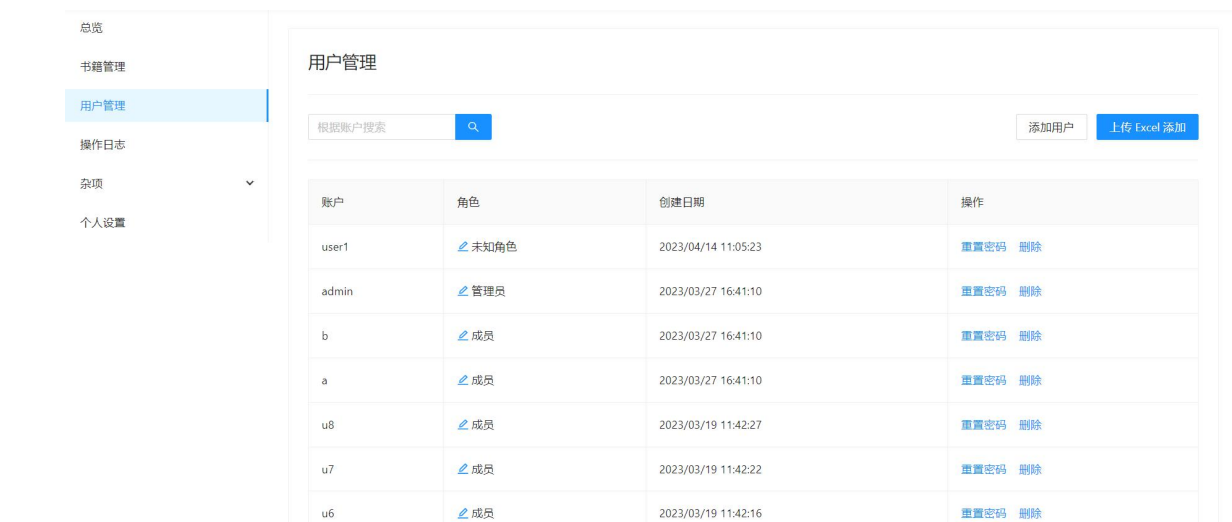


图 5-7 用户管理模块

（2）功能实现

用户搜索： 将搜索框内的文本传到后端，然后进行查询用户字段，若查找到则将数据重新发到前端，前端进行重新渲染。

添加用户： 点击后，弹出一个表单输入框，输入用户密码角色，然后点击确定，

前端将数据传给后端，后端保存到数据库中。如图 5-8



图 5-8 添加用户

修改角色：点击修改角色按钮，弹出一个下拉框，选择修改后的角色，然后点击确定，前端将修改后的数据发送到后端，后端进行保存。如图 5-9

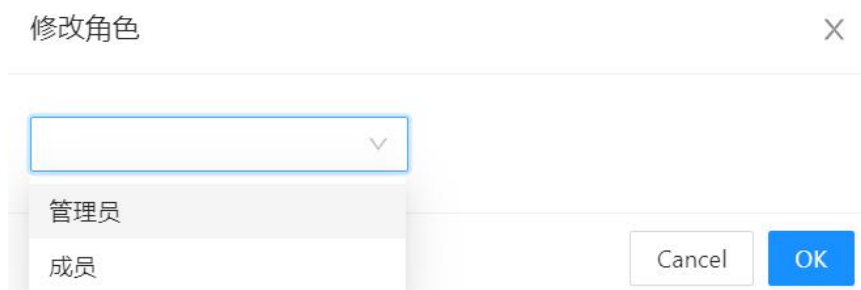


图 5-9 修改角色

重置密码：超级管理员点击重置密码。

删除用户：点击某个删除按钮，则该用户信息就从后端数据库中删除了。

5.6 操作日志模块

（1）模块描述

该页面专门为超级管理员展示和操作日志记录，旨在提供一种全面的管理方式。主要记录某个用户在特定时间内的操作行为，包括登录、注销、修改等等。这些操作记录可以方便管理员了解系统的使用情况，同时也为用户提供了更好的保障和服务。此外，超级管理员可以根据需要删除某些日志记录。需要注意的是，这种删除只是在前端页面进行了隐藏，实际上后端数据库依然会保存这些信息。这么做的目的是为了以防误删或其他错误操作对系统数据的影响，同时也为管理员提供了一种备份和恢复数据的途径。总之，该页面为超级管理员提供了一种全面的日志记录和管理方式，帮助他们更好地监控和管理系统使用情况。

（2）功能实现

将后端数据库里的日志数据，包括用户名，动作，记录时间，返回给前端，前端在页面上进行渲染。

5.7 书籍分类模块

（1）模块描述

该页面是一个用于书籍分类管理的重要工具，它提供了多种管理功能，以方便管理员对书籍分类进行管理。如图 5-10。

管理员可以通过在输入框内输入分类名称，然后点击添加按钮，来新增一个新的分类。一旦分类被添加到系统中，管理员可以在下方的书籍分类列表中查看所有已有的分类，并进行相应的管理操作。在书籍分类列表中，每个分类的名称和相关操作都会被列出来。管理员可以通过点击相应的操作按钮，对书籍分类进行修改或删除。修改功能允许管理员修改分类的名称或其他属性，以便更好地管理和组织书籍分类。删除功能则可以帮助管理员快速删除不需要的分类，以保持分类列表的整洁和有序。

总之，该页面是一个非常实用和重要的书籍分类管理工具，为管理员提供了方便和高效的分类管理方式，帮助他们更好地管理和组织图书资源。

（2）功能实现

在输入框输入内容，点击添加按钮就会将前端的内容发送给后端，然后查看后端是否存在，若存在则提示，不存在则添加进数据库。点击修改按钮可以对书籍分类名称进行修改，后端也跟随同步完成，删除也从后端数据库中删除。

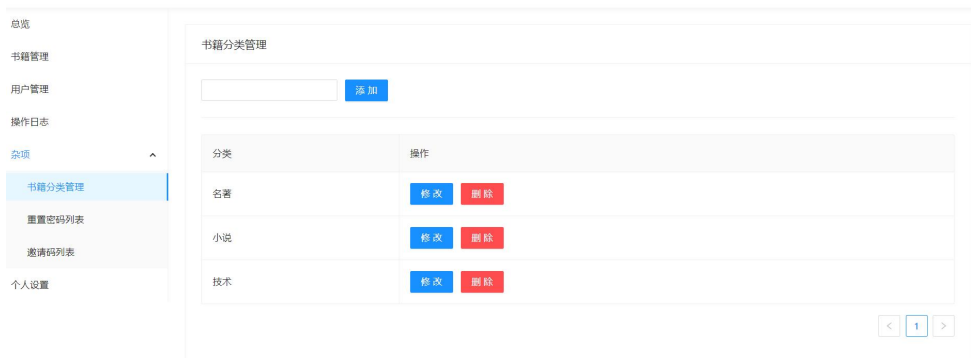


图 5-10 书籍分类名称

5.8 重置密码列表模块

（1）模块描述

这个页面只有超级管理员才能看到，它展示了申请密码的账户和相关操作。这些

操作包括重置密码和忽略申请。其他用户或管理员无法访问或操作此页面。因此，超级管理员在管理和维护密码安全方面扮演着至关重要的角色，他们必须保证只有合法的用户才能获得密码并防止非法用户获得未经授权的访问权限。通过严格的安全措施和管理流程，超级管理员可以确保系统的安全性和保密性，从而为组织和用户提供更高的保障和信任。如图 5-11 重置密码申请列表。

（2）功能实现

普通管理员点击申请重置密码按钮，后端数据库记录信息，然后传到前端，在超级管理员的重置密码申请列表中显示，超级管理员可以点击重置来初始化用户的密码或者忽略该条重置密码的申请信息。

重置密码申请列表	
账户	操作
u1	<button>重置</button> <button>忽略</button>
u2	<button>重置</button> <button>忽略</button>

< 1 >

图 5-11 重置密码申请列表

5.9 邀请码列表模块

（1）模块描述

该页面是邀请码列表模块，用于超级管理员进行生成新的邀请码给用户注册使用，每个邀请码只能用一次，可以在输入框内输入要生成的邀请码数量，然后点击添加按钮，就会生成若干个邀请码。下面是邀请码列表，包括邀请码本身、使用状态、操作，在操作中可以点击删除来删除此邀请码。如图 5-12

（2）功能实现

输入框内输入数字，然后传到后端，后端使用 uuid 来生成若干个邀请码存到数据库，然后发送到前端进行渲染，若邀请码已经使用过了则显示已使用状态，点击删除按钮即可把邀请码从后端数据库中删除。

邀请码管理

1

添加

邀请码	使用状态	操作
a620d967-90fc-4919-9c11-42011f7cd181	已使用	<div>删除</div>
c30ab9a7-9a77-4dd5-85a4-492dc2ee6f77	未使用	<div>删除</div>
75cd305e-27f9-4e76-8d87-6f4646909ef6	未使用	<div>删除</div>
d0a5b2f5-33be-4ae7-8d2f-245e34545e94	未使用	<div>删除</div>
65516492-e994-4ddc-bb58-2b8b5ea36fcb	未使用	<div>删除</div>
9386daf6-1bdf-4e2a-bcbc-b68858d42ac8	未使用	<div>删除</div>
b0ebe5b4-a00d-4e23-ab72-af7899a9e580	已使用	<div>删除</div>

图 5-12 邀请码列表

5.10 个人设置模块

（1）模块描述

该功能模块是用于用户修改密码的一个关键工具，它提供了一种简单和直接的方式，让用户能够快速、方便地修改他们的密码。

在使用该功能模块时，用户首先需要在原始密码框中输入他们的原始密码，以便系统验证其身份。一旦身份验证成功，用户就可以输入新密码和确认密码，以便系统进行密码更新。

在确认修改密码之前，系统会对新密码和确认密码进行验证，以确保两次输入的密码一致。如果新密码和确认密码匹配，则系统会将新密码存储到后端数据库中，以供用户下一次登录时使用。

在整个过程中，为了保证数据的安全性和可靠性，我们需要采取一系列的数据安全措施，例如加密传输、密码强度校验、防止 SQL 注入等，以确保用户的密码不被恶意攻击者盗取或篡改。只有在数据安全和可靠性的基础上，我们才能为用户提供更好的密码修改服务。如图 5-13



修改密码

原始密码:

新密码:

确认新密码:

图 5-13 修改密码

（2）功能实现

用户输入原始密码，新密码，然后从前端将数据发送到后端，后端对用户表进行查询，查到该用户，找到密码，进行修改。

6 系统测试

系统测试是项目开发过程中最为关键而不可省略的步骤。它的主要目的是确保系统在实际应用中能够稳定可靠地运行,并满足用户需求与标准要求。为达此目的,该系统选择采用黑盒测试(又称功能测试)方法进行测试。黑盒测试是一种根据系统功能需求规范设计测试用例的方法。在黑盒测试中,测试人员不需要考虑系统的内部实现细节,而是根据需求规范编制涵盖各种输入及操作的测试用例。系统测试的重要性不言而喻。实施系统测试需要设计全面而严密的测试用例,要考虑系统在各种条件下的功能与性能表现。黑盒测试正是一种有效的系统测试方法。它不依赖系统内部的技术细节,纯粹从最终用户的角度出发,构思各种操作流程和测试场景。这种基于外部表现的测试方法,可以发现系统接口、功能实现等方面的缺陷,检查系统的整体稳定性和易用性。相比之下,白盒测试直接依赖代码实现,TestSuite 等自动化方法可能存在遗漏或盲点,未必能全面覆盖各类输入与使用场景。而黑盒测试则更贴近真实用户,测试用例设计也更加全面,这使其成为验证系统质量的重要手段,尤其适用于系统测试这最终阶段的检验。综上,系统测试采用黑盒测试方法,有利于从最终用户视角检验系统的整体质量与可用性。黑盒测试不依赖系统内部实现,测试用例设计更加全面,可以较全面地揭示系统潜在的问题,这对保证交付产品的质量至关重要。系统测试是项目成功与否的最后判断标准,其重要性不言而喻。这些测试用例被分为有效等价类和无效等价类两大类别。

(1) 该系统总结的等价类表如下表 6-1 所示:

表 6-1 等价类表				
需求	有效等价类	编 号	无效等价类	编 号
管理员注册，用户添加：用户名为 2-16 个字符，且不重复（普通管理员注册为例）	2=<输入长度<=16 且不重复	1	输入长度>16	1
管理员注册：两次输入密码相同	输入密码相同	2	输入密码不同	3
添加/修改：联系电话符合格式	正确手机号格式	3	非数字，字母	4
时间：相关日期符合格式要求	正确日期格式	5	有特殊符号	5
入库：数量填写符合规范	正确价格数字	6	非数字，字母	6

(2) 具体的覆盖数据如下表 6-2 所示:

表 6-2 测试用例表

测试用例标识	输入	用例描述	期望输出	实际输出
1	zhangsan	该系统已存在客户：admin，zhangsan 在有效输入长度范围内，且不重复	正确	正确
2	123, 123	密码输入相同	正确	正确
3	18854565688	输入符合手机号码格式要求	正确	正确
4	6228573747828818188	输入符合银行卡号格式要求	正确	正确
5	2018-07-07	输入符合日期格式要求	正确	正确
6	55	符合价格填写要求	正确	正确
7	15	符合数量填写要求	正确	正确
8	表中数据完整填写	符合系统规范填写要求	正确	正确
9	Admmmmmmmmmmimnn	用户名输入长度为 17 位	提示错误	提示错误
10	Y	用户名输入长度为 1 位	提示错误	提示错误
11	admin	该系统已存在客户：admin，用户名重复	提示错误	提示错误
12	123, 111	两次密码输入不同	提示错误	提示错误
13	1883483g900	手机号码中含有非数字	提示错误	提示错误
14	188	输入手机号位数为 3 位，不满足 11 位	提示错误	提示错误
15	622857374782881818h	银行卡号中含有非数字	提示错误	提示错误
16	622	输入位数为 3 位，不符合卡号要求	提示错误	提示错误
17	2018-07-0\$	含有特殊符号	提示错误	提示错误
18	2018-07-07	日期前面有空格	提示错误	提示错误
19	2k	含有非数字	提示错误	提示错误
20	4%	含有特殊符号	提示错误	提示错误
21	2.5	数量输入为小数	提示错误	提示错误
22	5g	含有非数字	提示错误	提示错误
23	表单填写有空格	为完整填写表单内容	提示错误	提示错误

7 总结与展望

本章主要是对系统的设计与实现的过程进行总结。本系统还存在很多值得探索的地方，在本章最后也阐述了系统未来的发展方向。

7.1 总结

本毕设是基于 Vue+Node 的图书管理系统，这是一个全栈开发项目，主要涉及前端技术 Vue 和后端技术 Node。通过这个项目，不仅掌握了前端和后端开发技术，还了解了全栈开发的整个流程。

在整个项目开发过程中，首先了解了 Vue 框架的特点和使用方法，学习了 Vue 的基本语法和组件开发方式。随后，使用 Vue-cli 搭建了项目框架，并选择了 Ant-Design-Vue 作为 UI 框架，使得整个项目的开发变得更加高效和便捷。在后端开发方面，选择了 Node 作为后端技术，并使用了 Koa 框架，实现了后端 API 的开发和数据库的连接。通过对数据库的学习和使用，进一步掌握了数据库的基本操作和查询语句，以及如何在项目中运用数据库，如何通过数据库存储数据和读取数据。在项目开发的过程中，也遇到了很多问题，如前后端的数据交互问题、跨域问题、认证和授权等问题，但我都通过查阅资料和自己的思考解决了这些问题。在项目的最后，开发了图书管理系统的基本功能，包括用户登录、注册、图书查询、图书添加、修改和删除等功能。在这个过程中，不仅熟悉了整个项目的流程，还深刻体会到了如何设计和开发一个完整的应用程序，并且通过这个项目，也进一步提高了自己的编程能力和实践能力。

7.2 展望

基于 Vue+Node 的图书管理系统是一个很好的全栈开发项目，其应用范围广泛，涉及到许多行业和领域。未来，图书管理系统将会更加智能化，更加便捷、高效，具有更好的用户体验和用户交互。

随着技术的不断发展，前端和后端技术也将不断更新和迭代，更多的新技术将应用到图书管理系统中，例如人工智能、大数据、区块链等技术。这些技术的应用将会进一步提升图书管理系统的智能化和安全性。未来的图书管理系统还将更加注重用户体验和用户需求，更加人性化和个性化。在用户界面方面，将会加入更多的交互和动效设计，使得用户操作更加顺畅和自然。在数据分析方面，图书管理系统将会更加注重数据的收集和分析，通过数据分析来了解用户需求和用户偏好，进而为用户提供更

好的服务和体验。另外，未来的图书管理系统还将更加注重安全性和隐私保护。随着数据泄露事件的不断发生，数据安全已经成为了图书管理系统开发中必须要考虑的问题。未来的图书管理系统将会采取更加严格的安全措施，保证用户数据的安全和隐私保护。

总之，未来的图书管理系统将会更加智能化、便捷、高效、人性化、个性化、安全和隐私保护。通过不断更新和迭代，图书管理系统将会为用户提供更好的服务和体验，成为用户生活中不可或缺的一部分。

致 谢

在我的毕业论文即将结束之际，我想向所有支持我完成这个项目的人们表示最深刻的感谢和敬意。

首先，我要感谢我的指导老师王大伟老师，他对我的毕业论文进行了耐心、细致、严谨的指导，帮助我排除了一个个困难和问题。他的专业知识、严谨的态度和深入浅出的教学方法，让我在完成这个毕业论文的过程中受益匪浅，收获了很多宝贵的经验和知识。

其次，我要感谢我的家人和朋友。感谢他们在我完成毕业论文的过程中给予我的精神支持、生活照顾和鼓励。他们是我完成毕业论文的坚强后盾，是我在迷茫和疲惫时给予我的坚定支持和鼓舞，他们的支持和鼓励是我前进的动力。

此外，我还要感谢我的同学们，感谢他们与我分享自己的经验和成果，为我提供了宝贵的资料和参考。我们共同完成毕业论文的过程是一段宝贵的经历，让我了解到团队协作的重要性，也让我体验到了自己的成长和进步。

最后，我要感谢所有为我提供帮助和支持的人们。在完成毕业论文的过程中，我遇到了很多困难和挑战，但是也因此结交了许多朋友和同事。感谢他们的耐心解答、热情帮助和真诚支持，他们的帮助让我能够成功地完成这个毕业论文。

在未来的日子里，我会继续保持积极向上的态度和精神，为实现自己的人生目标而不懈努力。再次感谢所有支持我的人们，祝愿你们幸福、健康、快乐！

参 考 文 献

- [1] 胡芸. 基于 React 和 Node.js 的中台开发框架设计与实现[D]. 华中科技大学, 2019. DOI:10.27157/d.cnki.ghzku.2019.005754.
- [2] 吴彦亮. 基于 WIFI 技术的黑龙江大学图书管理系统设计与实现[D]. 北京邮电大学, 2012.
- [3] 刘佳音. 涉密人员及载体的安全保密管理信息系统的设计与实现[D]. 吉林大学, 2014.
- [4] 满勤. 基于 J2EE 的 B2C 电子商务系统的设计与实现[D]. 电子科技大学, 2012.
- [5] 王皓. 培训机构综合管理系统的设计与实现[D]. 西北民族大学, 2019. DOI:10.27408/d.cnki.gxmzc.2019.000357.
- [6]]张小帅. 车险智能定价系统的设计与实现[D]. 北京交通大学, 2020. DOI:10.26944/d.cnki.gbfju.2020.001046.
- [7] 赵美勇, 宋思睿. 基于 PHP 的图书管理系统设计[J]. 计算机产品与流通, 2019(09):79.
- [8] 王莉萍, 戴晓峰. 基于大数据的智能图书管理系统的设计与实现[J]. 信息与电脑(理论版), 2023, 35(01):162-164.
- [9] 陈明雪, 赵勇, 周小丽等. 基于 Node.js 和 Bootstrap 的图书管理系统设计与研究[J]. 现代信息科技, 2021, 5(24):12-15. DOI:10.19850/j.cnki.2096-4706.2021.24.004.
- [10] 李云. 基于 Vue 框架开发的学生管理系统[J]. 数字通信世界, 2022, No. 207(03):50-53.
- [11] 毕好昌, 徐在文. 基于 Require.js+koa2+Typescript+Routing-controllers 框架设计与实现[J]. 集宁师范学院学报, 2022, 44(05):82-86.
- [12] 唐榜. 基于 Node.js 的 Web 服务端框架研究与实现[D]. 西南科技大学, 2021. DOI:10.27415/d.cnki.gxngc.2021.000946.
- [13] 徐浪. 基于 Node.js 的 Web 应用框架研究与实现[D]. 安徽工业大学, 2019. DOI:10.27790/d.cnki.gahgy.2019.000001.
- [14] 张晓颖. 试析基于 Node.js 的前后端分离框架的实现[J]. 计算机产品与流通, 2018(10):24.
- [15] 王萍芳. 基于 VUE 的智能采摘机器人前端开发框架研究[J]. 农机化研究, 2023, 45(05):229-232. DOI:10.13427/j.cnki.njyi.2023.05.016.