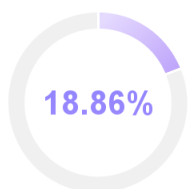


NO. 395730a4859c7a3f | 2025-05-09 03:13:31

- 题目：高校图书管理系统的设计与实现
- 作者：王钰博
- 检测所属单位：重庆工商大学派斯学院

📄 论文字符数：16364 📄 论文页数：38 📊 表格数量：9 🖼️ 图片数量：26

检测结果



18.86%

全文疑似AIGC生成

81.14%

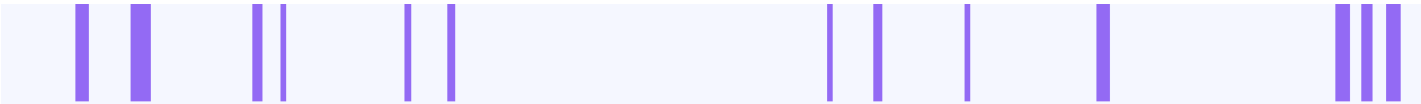
全文人写概率

结果分布

序号	章节	AI生成文字/章节总字数	AI生成章节占比	人工占比
1	摘要	287/327	87.0%	13.0%
2	1 引言	0/727	0.0%	100.0%
3	2 相关技术介绍	204/1478	13.0%	87.0%
4	3 需求分析	96/2498	3.0%	97.0%
5	4 系统设计	278/2159	12.0%	88.0%
6	5 系统实现	397/3458	11.0%	89.0%
7	6 系统测试	559/1063	52.0%	48.0%
8	7 总结与展望	477/1083	44.0%	56.0%

*注:格式规范的情况下可准确识别章节, 若论文中无章节, 可能会识别有误。

片段分布



文字标注

■ 自写片段 ■ 疑似AI生成

高校图书管理系统的设计与实现

重慶工商大學

毕业设计说明书

题目	高校图书管理系统的		
	设计与实现		
学院	软件工程		
专业年级	计算机科学与技术2021级		
学生姓名	王钰博	学号	2021105127
指导教师	马亚军	职称	副教授
指导教师	彭泽玉	职称	助教

重庆工商大学派斯学院教务处 制

高校图书管理系统的设计与实现

重庆工商大学派斯学院 计算机科学与技术 2021级 4班 王钰博

指导教师 马亚军 彭泽玉

中文摘要：近年来，随着信息技术的快速发展，高校图书管理系统的数字化需求日益增长。传统的手工管理方式已无法满足现代图书馆对高效、准确服务的需求。本文设计并实现了一个高校图书管理系统。系统采用MVVM框架来实现前后端分离，数据存储采用的MySQL，前端采用WPF、XAML、Prism 框架，后端则采用Net框架和C#语言来实现完整的图书管理系统。系统分为读者和管理员两种角色，实现了图书信息管理、图书分类管理、读者管理、借阅管理、权限管理、逾期处理等功能。系统通过自动化管理功能，提高了图书管理的效率和准确性，改善了用户的借阅体验，并实现了图书资源的合理配置和利用。通过实验验证，系统在功能、性能和安全性方面表现出色，为图书馆管理提供了高效、便捷的解决方案。

关键词：图书管理；WPF；MVVM；Prism；SqlSugar

Abstract: In recent years, with the rapid development of information technology, the digital demand for university library management systems has been increasing. The traditional manual management methods are no longer able to meet the needs of modern libraries for efficient and accurate services. This article designs and implements a university library management system. The system adopts the MVVM framework to achieve front-end and back-end separation, with MySQL used for data storage. The front-end uses WPF, XAL, and Prism frameworks, while the back-end uses the Net framework and C # language to implement a complete library management system. The system is divided into two roles: reader and administrator, and implements functions such as book information management, book classification management, reader management, borrowing management, permission management, and overdue processing. The system has improved the efficiency and accuracy of book management through automated management functions, enhanced the borrowing experience for users, and achieved the rational allocation and utilization of book resources. Through experimental verification, the system has shown excellent performance in terms of functionality, performance, and security, providing an efficient and convenient solution for library management.

Key Words: Library Management WPF; MVVM; Prism; SqlSugar

目 录

1 引言1

1.1课题研究的背景1

1.2课题研究意义1

2 相关技术介绍2

2.1 WPF技术2
2.2 MVVM架构模式2
2.3 Prism 框架2
2.4 SqlSugar ORM 框架2
2.5 DevExpress UI 控件库3
2.6 SHA256 加密技术3
3 需求分析4
3.1 可行性分析4
3.2 功能需求分析4
3.3 非功能需求分析7
4 系统设计9
4.1 系统功能模块设计9
4.2系统类图10
4.3 数据库设计10
5 系统实现15
5.1 登录模块15
5.2 图书管理模块17
5.3 借阅管理模块19
5.4 用户管理模块22
5.6技术总结27
6 系统测试29
6.1系统的测试方法29
6.2测试结果30
总结与展望31
7.1 工作总结31
7.2 展望32
致谢33
参考文献34

1 引言

本章节主要阐述了研究课题的背景与意义，它解析当前时代传统图书管理已经无法满足当前大数据的时代，揭示了数字化图书管理的必然性。

1.1课题研究的背景

信息技术发展速度加快，图书管理的信息化需求不断上升，现代图书馆要应对海量图书资源管理任务，传统手

工管理方式缺点渐显，手工管理效率不高，花费很多人力与时间，还易有记录错误、数据丢失问题，难契合用户对快速准确服务的要求，随着数字化技术普及，图书馆用户对借阅体验和管理服务期望更高，传统管理模式无法适应这一变革。

在此情况下，数字化图书管理系统出现，成为图书管理领域关键发展方向，数字化系统借助信息技术，能实现图书信息自动化管理、优化借阅流程并高效处理数据，和传统方式相比，它提高了管理效率，为用户带来更便捷服务体验，也为图书管理决策提供可靠数据支撑，研究开发高效数字化图书管理系统有关键现实意义。

1.2 课题研究意义

本课题将要设计且实现一个功能齐全、性能优良的图书管理系统，解决传统图书管理模式里效率偏低以及错误率居高不下的问题，拉动图书馆管理朝着智能化、数字化方向拓展，利用本系统的开发与施用，预计可达成以下几方面的意义：一是提高图书管理的效率以及精准度，系统凭借自动化管理的实用功能，去掉人工操作的繁杂流程，减小人为犯错出现的几率，图书信息录入、修改及查询这些操作变得更便捷高效，为管理员省下大把时间，二是让用户的借阅体验变好，系统推出简化的借阅流程，具备借书、还书等功能模块，用户可凭借友好界面简单完成操作，这种便利性与透明度可提升用户的满意度，三是实现对图书资源的合理配置与运用，依靠对借阅数据和图书信息开展统计分析，管理员可更明白地掌握图书使用情形，提升资源配置的合理性，提高图书的流通速度与利用程度。

2 相关技术介绍

在本系统中，采用了一系列先进的技术和框架，以确保系统的功能性、性能和安全性。以下是对这些关键技术的详细介绍及其在系统开发中的应用。

2.1 WPF技术

WPF即Windows Presentation Foundation，是微软所推出的一款用于构建Windows桌面应用程序的框架，它凭借强大的UI设计能力而闻名，可支持丰富的图形、动画以及多媒体功能，在本系统当中，WPF被运用来构建用户界面，提供了直观且现代化的交互体验。它的核心特性之一便是XAML语言，借助XAML，界面设计和逻辑代码实现了分离，这利于系统的维护与扩展，开发者可利用XAML去定义复杂的界面布局与样式，并且凭借WPF的数据绑定机制达成UI与后台数据的动态同步，以此提升开发效率以及用户体验。

2.2 MVVM架构模式

MVVM即Model-View-ViewModel是一种软件设计模式，此模式可将用户界面同业务逻辑分离开来，提升代码的可维护性以及可测试性，在本系统当中，MVVM模式得到了广泛运用，其结构主要分成三层。Model层负责数据管理以及业务逻辑，对系统的核心功能给予封装。View层负责UI展示，仅仅处理界面显示以及用户交互。ViewModel层作为Model和View之间的桥梁，处理用户输入以及数据绑定，保证界面和数据实现实时同步，这种清晰的架构设计让系统结构更具模块化，方便团队协作以及后期维护，同时也为单元测试给予了便利。

2.3 Prism 框架

Prism属于WPF开发里颇为关键的工具，它给出了一整套完整的MVVM实现办法，像模块化开发、依赖注入以及命令绑定等功能都涉及其中，在本系统内，Prism框架用来管理应用程序的模块与组件，以此保证系统达成高内聚低耦合，依靠Prism的模块化特性，系统划分成了多个独立的功能模块，像用户管理、图书管理等，每个模块都可单独进

行开发与测试。而且Prism还有导航和事件聚合功能，提高了模块间的通信能力，提高了系统的灵活性。

2.4 SqlSugar ORM 框架

SqlSugar是一款有轻量级特点的ORM也就是对象关系映射工具，它可支持多种不同类型的数据库，还提供了简洁明了的API用以执行CRUD操作，在当前这个系统当中，SqlSugar被应用于数据库交互工作，极大地简化了数据访问层的开发工作。它所拥有的高性能以及易用性，让开发者可借助fluent API迅速构建复杂的查询，同时还支持事务管理以及批量操作，以此来保证数据操作的高效以及安全。

2.5 DevExpress UI 控件库

DevExpress是一组功能强大的UI组件，包含多种高级控件，像数据网格、图表以及导航栏等，在本系统里，DevExpress控件用于提高用户界面的功能性与视觉效果，举例来说，数据网格控件用来展示图书列表和借阅记录，有排序、过滤以及分页等功能，图表控件用于生成统计报表，直观呈现借阅数据与图书使用状况，导航栏控件提供简洁的菜单结构，便于用户快速访问系统功能。这些控件的运用提高了系统专业性，也改善了用户体验。

2.6 SHA256 加密技术

SHA256 属于安全散列算法范畴，其作用在于保障用户密码的安全程度，在这个系统里，用户密码于存储之前会借助 SHA256 实施加密操作，因为SHA256算法有不可逆的特性，哪怕数据库出现泄露情况，攻击者看到的也是乱码并且难以翻译成明文密码。

本系统借助集成 WPF、MVVM、Prism、SqlSugar、DevExpress 以及 SHA256 等多项技术，达成了高效的开发流程以及优质的用户体验，这些技术协同运用，提升了系统的功能性、性能与安全性，为用户供给了稳定、可靠且直观的服务。

3 需求分析

需求分析是系统开发过程中的关键步骤，旨在明确系统要解决的问题、用户的需求以及系统的功能和非功能要求。以下是对图书管理系统需求分析的完整阐述。

3.1 可行性分析

在开发图书管理系统之前，进行可行性分析以确保项目能够成功实施是必要的。可行性分析从以下四个方面展开：

3.1.1 技术可行性

系统采用成熟的 .NET 技术栈（如 WPF、MVVM、Prism 和 SqlSugar），这些技术在桌面应用开发中已被广泛验证。开发团队具备相关经验，系统设计注重模块化和可扩展性，便于未来维护和升级。

3.1.2 经济可行性

传统图书过于管理依赖人工操作，并且图书数量过多不方便统一管理，成本高而且效率很慢，往往需要花费很长的时间去整理图书数据。数字化系统初期需投入开发和部署成本，但长期来看能显著降低人工管理成本、提升效率，带来经济效益。

3.1.3 操作可行性

系统界面设计简洁友好，操作流程直观，用户无需复杂培训即可上手。同时提供操作指南和帮助文档，进一步提升使用便捷性。

3.1.4 法律可行性

系统遵守数据保护和隐私法规，用户数据加密存储，设置权限管理机制，防止未经授权的访问，确保合法合规。

综上所述，此图书管理系统依托了非常成熟的.NET技术栈，并且结合了模块化设计，具有可靠的实现基础，并且能够在后期进行更新与维护。因此，该图书管理系统的可行的。

3.2 功能需求分析

对于图书馆里系统，如果直接购买一个图书管理系统，相对的经费开销就比较大，并且后期也需要不断的对系统进行维护。这样下来开销就比较打。所以因地制宜，根据当前的状况自己开发一套适合自身使用的图书管理系统是个不错的选择。

本文的功能需求分析首先明确图书管理信息系统中业务角色定位，再对系统中的三个功能模块地组成部分进行解析，每一模块都有功能需求。

3.2.1 角色分析

角色分析就是指系统里的读者和管理员分别可能是哪些用户，例如读者可以是学生，老师，和部分社会人员。管理员一般都是图书馆相应的工作人员。

读者类别的用户可以直接查阅当前馆里所有的图书状态和记录，可以通过搜索关键词、书名、作者等在系统中查找所需图书是否在馆、是否可借等，还可以实时查看个人的借阅信息等。

管理类别的用户就可以有更多的权限，例如修改图书信息，添加图书，修改用户状态，删除用户。从而更方便的对图书信息进行管理。管理员的工作职责是让图书馆能够稳定有序的为广大师生服务。

3.2.2 业务功能

图书管理系统中的用户分为两大类，一类是读者一类是管理员。

(1) 读者功能需求

注册登录：用户注册允许用户输入用户名、密码、邮箱等信息。注册时，系统会检查用户名是否已存在、注册的信息是否符合要求。并且可以通过此账号系统。

充值余额：用户借阅图书过后，如果未按时归还则需要缴纳逾期费用，所以用户需要余额进行充值才能够归还图书。

修改个人信息：方便用户随时对自己的信息进行修改，以免发生信息填写错误无法修改的问题。

借还图书：用户主要的功能就是对图书的借阅和归还功能，这样为用户提供了便捷的借还操作，并且确保了系统安全性和身份的唯一性。

查询图书：图书的数据往往是大量的，查询功能方便用户对于图书的查找。

如图3-1所示,读者可以登录、注册、对个人的信息进行修改和对账户进行余额充值。管理员权限更高，不仅可以对自己的信息进行修改还能修改用户的信息，对用户进行添加、删除，但不能注册，因为员账号已经在内部添加上。





图3-1用户管理用例图

(2) 管理员功能需求

登录：管理员账户是内部存在的，不能够注册。因此只能够完成登陆功能。

图书、用户信息修改：管理员的权限级别高，能够对图书、用户的信息进行增删改查，通过这一功能极大的方便了对信息的全面管理。

修改个人信息：管理员也是可以对自身的信息进行修改，主要目的也是以免发生信息填写错误无法修改的问题。

系统值设置：对最大借阅期限，逾期费用等系统设定值进行修改，方便管理员随时根据当前用户需求来随时进行修改。

如图3-2所示读者和管理员是本系统的使用者，读者可以借还图书，并且可以按照图书名称，作者名等关键字来搜索图书，并且在归还图书的时候支付逾期费用。管理员的功能就是对图书的信息进行修改，添加图书，并且查看用户的借阅记录。

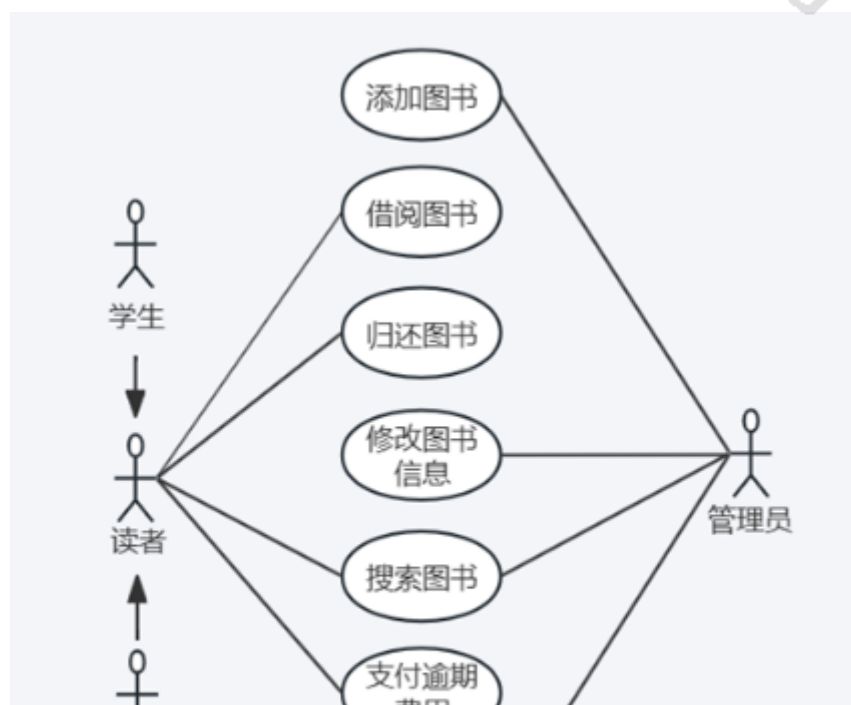




图3-2图书管理用例图

如图3-3所示，该功能为管理员使用的功能。管理员可；以对系统的设置进行修改，例如用户的最大借阅数量，逾期的费用多少钱一天。方便管理员随时调整。

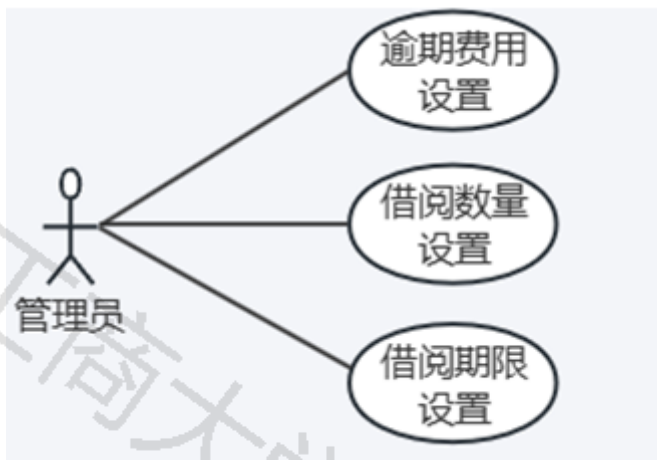


图3-3系统设置用例图

3.3 非功能需求分析

在非功能需求方面，本图书管理系统主要涵盖安全性、性能以及可靠性等维度，以下是详细分析：

（1）安全性

账户安全：对于图书管理系统还需要对用户的安全性进行保障，以免发生账号被盗，

用户都是有余额的，如果管理不当会带来不小损失。所以本系统使用了SHA256 加密技术，这样密码就算泄露了也不易被破解，数据库中存放的都不是直接的密码，而是经过加密并且不易逆向解开。

系统防护：采用ORM层封装数据库交互逻辑，明显增强了抵御SQL注入的能力，对象-关系映射，对象关系映射技术隔离了SQL动态片段与静态代码，变量经预编译处理后，其生成的语句与参数分别发送至数据库执行。

数据保护：用户的权限分类也能够保护数据安全，普通用户查询不到其他用户的数据信息，为用户提供了隐私安全保护。

（2）高性能

图书管理系统的数据库量是不容小觑的，其中存放了大量的图书数据，所以对数据库的操作也有要求，如果在每一次需要用到数据都向数据库发送请求，那系统相应会非常缓慢。本系统实现将需要的数据提前取出，需要用到的时候可以直接使用，这样就防止了多次对数据库进行操作从而影响程序响应。在模拟高并发场景测试下，90%的数据库查询操作响应时间不超过200毫秒，数据写入操作平均耗时不超过50毫秒，有效保障了系统的响应效率。

（3）可靠性

系统在进行修改数据时是最容易出现错误的时候，例如当用户进行修改数据时如果退出当前修改界面，很有可

能会将数据库进行更改，导致以前的数据被丢失。为了解决这个问题，此项目增加了备份功能，将原有数据进行备份，如果用户取消修改或者其他操作，都会提示用户是否保存更改，如果用户不用更改则将备份的数据覆盖当前的数据。

4 系统设计

在完成了对图书管理系统的需求分析之后，本章节将对系统的整体架构、模块划分、技术选择和数据结构等方面进行详细的规划与构建。系统设计的目标是实现系统的各项功能和非功能需求，确保系统的高效性、可维护性和可扩展性。通过合理的系统设计，我们将为后续的开发工作奠定坚实的基础。

4.1 系统功能模块设计

系统功能模块设计是系统的核心，将整个系统划分为多个独立的功能模块，每个模块负责特定的功能。以下是图书管理系统的功能模块设计：

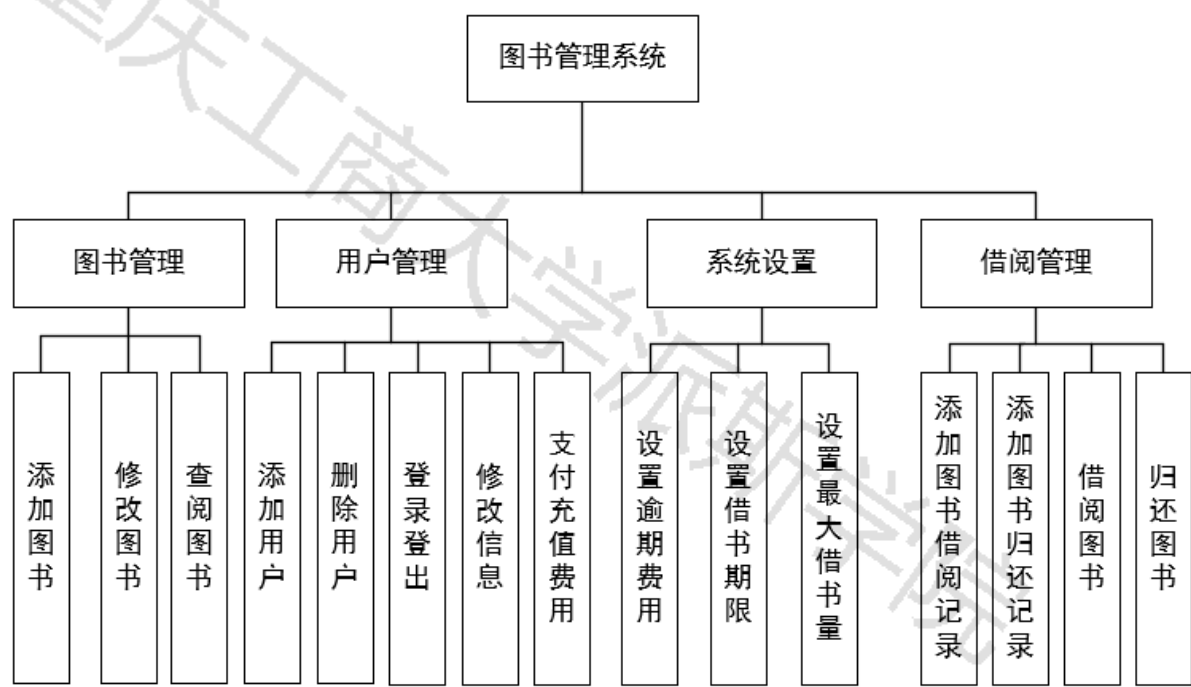


图4-1图书管理系统基本模块图

(1) 用户管理模块

普通用户：可以对自身的信息进行修改，有登陆登出功能，能对逾期的图书支付逾期费用。

管理员用户：可以对个人以及用户的信息进行修改，登录登出功能。

(2) 图书管理模块

管理员具有对图书信息的 CRUD 操作管理员可以添加、修改、删除，查询图书信息，普通读者只有查询，租借和归还功能。

(3) 借阅管理模块

借阅流程：用户借阅图书时，系统记录借阅信息并更新图书库存。

归还流程：用户归还图书后，系统更新借阅记录和图书库存。

（4）系统设置模块

系统参数配置：管理员可设置借阅规则、逾期费用、界面主题等参数。

4.2 系统类图

系统类图描述了系统的类、属性、方法以及类之间的关系，是系统设计的重要组成部分。以下是图书管理系统的类图设计：

（1）实体类设计

DbUser：存储用户信息

DbBook：存储图书信息

DbBorrowRecord：存储借阅记录

DbSystemSetting：存储系统设置

（2）模型类设计

UserModel：封装用户信息，用于数据传输。

BookModel：封装图书信息，用于数据传输。

BorrowRecordModel：封装借阅记录信息，用于数据传输。

（3）视图模型类设计

LoginViewModel：处理登录界面的逻辑。

MainWindowViewModel：处理主界面的逻辑。

AdminViewModel：处理管理员界面的逻辑。

（4）服务类设计

UserService：提供用户管理服务，如注册、登录、权限管理等。

BookService：提供图书管理服务，如添加、修改、删除图书等。

BorrowRecordService：提供借阅管理服务，如借书、还书、续借等。

SettingsService：提供系统设置服务，如借阅规则和逾期费用设置。

4.3 数据库设计

数据库设计是图书管理系统的核心，因为数据库存放了所有数据，系统的所有操作基本都是基于对数据库。然而对于数据库的设计直接影响到了系统的性能，一个好的数据库设计可以极大的提升系统的性能，每个系统当它的数据量过于庞大时，查询数据就会占据极大一部分时间，优秀的数据库设计就能够减少这一部分所消耗的时间来提升系统性能。

此系统采用的关系型数据库设计范式，结合了图书管理的业务特点，构建了一个以用户-图书-借阅记录为何行的数据模型。

4.3.1 系统数据模型E-R图

如图4-2所示为用户实体图，包括用户ID、用户名、密码、邮箱、账户余额、用户状态、姓名、联系方式、用户类别。





图4-2用户实体图

如图4-3所示为用户实体图，包括图书ID、ISBN、库存、图书名、作者、出版社、描述、出版日期、状态、价格、类型。



图4-3图书实体图

如图4-4所示为借阅记录实体图，包括借阅ID、用户ID、图书ID、借阅日期、逾期天数、归还日期、状态、逾期费用。



图4-4借阅记录实体图

如图4-5所示为系统设置实体图，包括设置名称、设置描述、设置值。

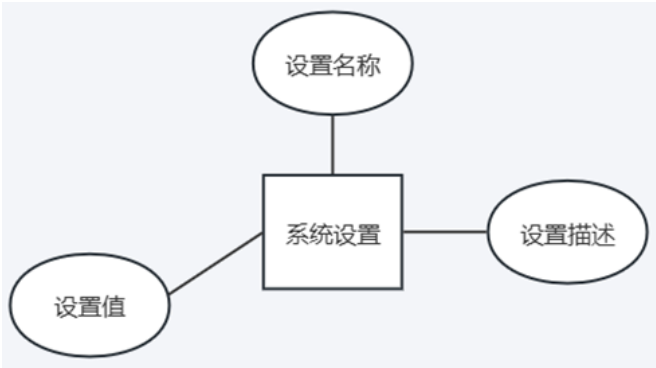


图4-5系统设置实体图

以下是该系统的系统基本E-R图。

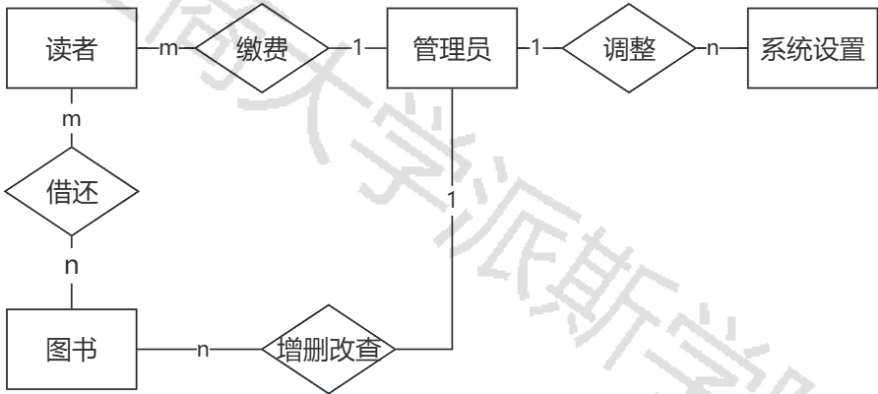


图4-6系统基本E-R图

根据以上的系统设计，此图书管理系统基本上实现了模块化、结构化的构建。功能模块设计可以明确系统的功能划分使得代码清晰明确，便于后期维护系统，类图设计定义了系统的类结构，数据库设计为数据存储提供了坚实基础，良好的数据库设计可以极大的提升系统响应时间。

4.3.2 数据库表设计

该数据库有图书表、借阅记录表、用户表和系统设置表。

图书表包含了bookID的字段，此字段是主键，唯一的，借阅记录表根据bookID来关联图书表的，图书表主要保存图书的信息，供用户查阅。

表4-1 图书表

列名	数据类型	字段含义	约束条件
----	------	------	------

BookID	Int (11)	图书ID	主键
--------	----------	------	----

ISBN	Varchar (20)	图书ISBN号
Book Name	Varchar (100)	图书名
Author	Varchar (50)	作者
Publisher	Varchar (100)	出版社
Publish Date	Date	出版日期
Price	Decimal (10.2)	价格
Category	Varchar (50)	类别
Stock	Int (11)	库存
Status	enum	状态
Description	text	描述

用户表包含了userID的字段，这是user表的主键，借阅记录表存放的信息也保存了userID来作为主键，可以连接用户表，此表主要保存了用户的基本信息，用户进行登录时会从此表获取信息来选择登录权限是管理员或者读者。

表4-2 用户表

列名	数据类型	字段含义	约束条件
UserID	Int (11)	用户ID	主键
Username	Varchar (50)	用户名	
Password	Varchar (100)	密码	

UserType	enum	用户类型
Name	Varchar(50)	姓名
Contact	Varchar(20)	联系方式
Email	Varchar(100)	邮箱
CreateTime	datetime	创建时间
Status	enum	状态
Balance	Decimal(10,2)	余额

借阅记录表存放了两个外键userID和bookID以此可以关联两张表。此表主要存放的是借阅记录的信息。

表4-3 借阅记录表

列名	数据类型	字段含义	约束条件
BorrowID	Int(11)	借阅记录ID	主键
UserID	Int(11)	用户ID	外键
BookID	Int(11)	图书ID	外键
BorrowDate	datetime	借阅日期	
DueDate	datetime	应归还日期	
ReturnDate	datetime	实际归还日期	
Status	enum	状态	

系统设置表是不与前面的表相关联的，它是存放“最大借阅期限”“逾期费用”：最大借书量“这些辅助功能的表。

表4-4 借阅记录表

列名	数据类型	字段含义	约束条件
SettingID	Int (11)	设置ID	主键
SettingName	Varchar (50)	设置名称	
SettingValue	Varchar (100)	设置值	
Description	text	描述（备注）	

5 系统实现

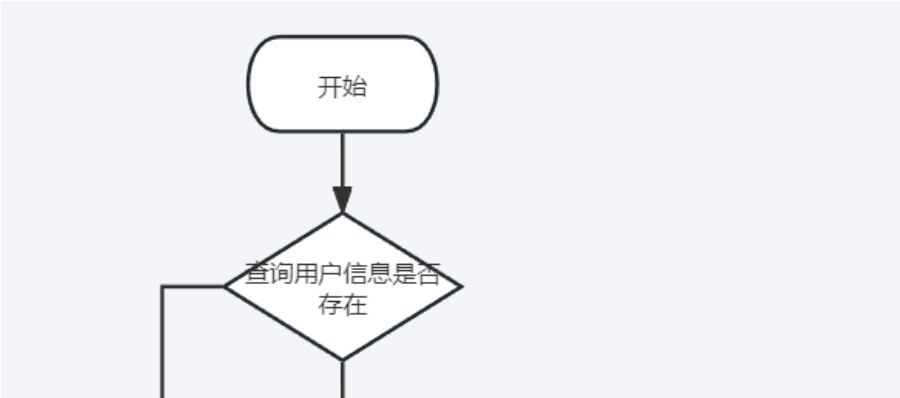
以下是图书管理系统的“系统实现”部分的详细描述，涵盖了系统中各个模块的具体实现方式，包括技术选择、关键代码逻辑和界面设计等几个方面。本部分将详细介绍登录模块、图书管理模块、借阅管理模块、用户管理模块和系统设置模块的实现过程。

5.1 登录模块

(1) 用户认证流程

用户注册允许用户输入用户名、密码、邮箱等信息。注册时，系统首先检查用户名是否已存在。如果用户名未被占用，系统创建新用户记录，将密码及相关信息存储到数据库中。

当用户点击登录按钮时，系统触发LoginViewModel 中的登录命令。LoginViewModel使用SqlSugarORM从数据库中查询用户信息，并将查询结果与用户输入进行比对。如果认证成功，系统根据用户角色（普通用户或管理员）导航至对应的主界面。登陆流程图如图5-1所示。



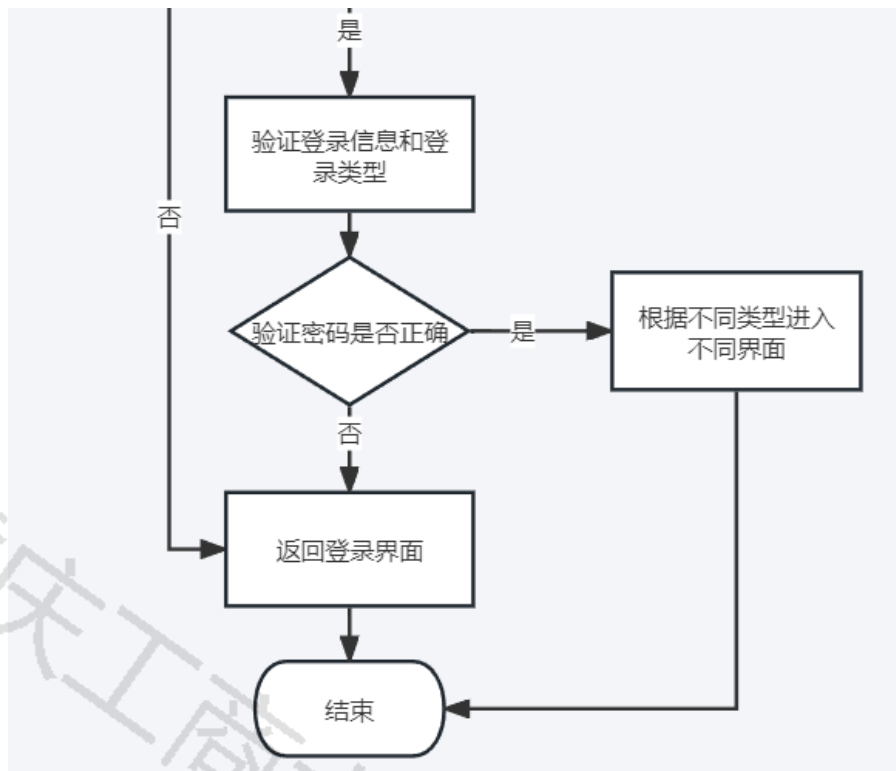


图5-1 登录流程图

(2) 密码加密实现

为维护安全性，采用SHA256哈希算法对用户注册密码进行加密保存，系统处理逻辑如下：系统采集用户所输入的密码。

采用SHA256算法对用户密码执行哈希运算，系统把哈希处理结果存储至数据库，系统实施统一。

采用SHA256算法对用户提供的密码执行哈希运算，与数据库中预先存储的哈希值对照。

具体代码实现如下：

```

private string HashPassword(string password)
{
    using (SHA256 sha256 = SHA256.Create())
    {
        byte[] bytes = sha256.ComputeHash(Encoding.UTF8.GetBytes(password));
        StringBuilder builder = new StringBuilder();
        for (int i = 0; i < bytes.Length; i++)
        {
            builder.Append(bytes[i].ToString("x2"));
        }
        return builder.ToString();
    }
}

```

}

(3) 用户登录界面实现

登录界面采用 WPF (Windows Presentation Foundation) 技术，通过 XAML 语言设计。界面包含用户名输入框、密码输入框、登录按钮和用户类型选择，设计简洁直观。界面通过数据绑定与 ViewModel 层连接，确保用户输入能够实时响应，提升交互体验。用户输入用户名、密码选择登陆类型，如果验证正确则进入相对应的主界面。登陆界面如图5-2所示。

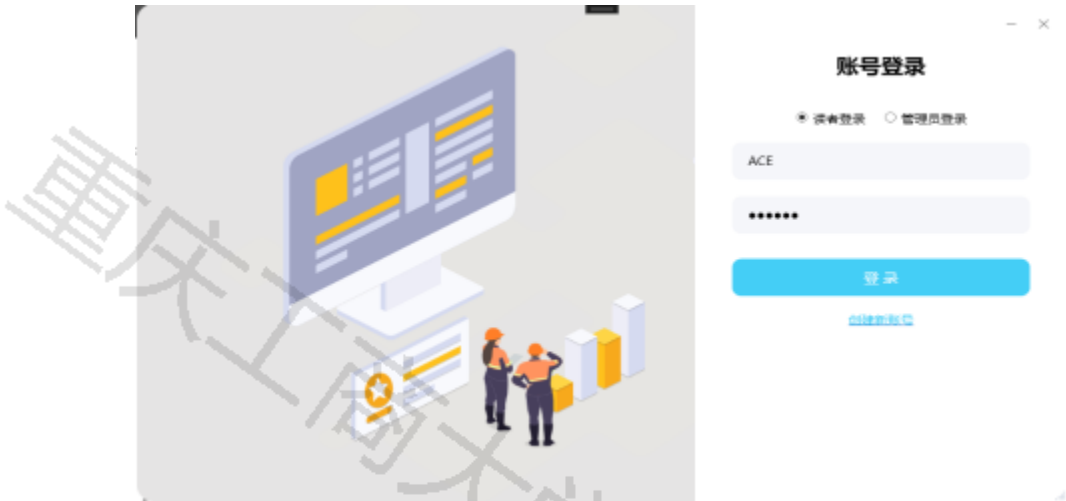


图5-2登陆界面

5.2 图书管理模块

(1) 图书信息管理界面

图书管理界面使用DevExpress提供的DataGrid控件展示图书列表，支持排序、过滤和分页功能。管理员可以通过界面上的工具栏按钮执行图书的添加、修改和删除操作，界面设计注重操作便捷性。图书管理界面如图5-3所示。

图5-3图书管理界面

(2) 图书添加、修改、删除功能

添加图书：管理员点击“添加图书”按钮，左边的修改图书信息对话框切换成添加图书信息对话框，允许输入图书名称、作者、ISBN 等信息（如图5-4）。点击“确定添加”后，BookService将数据插入数据库。

添加图书信息

图书ISBN

图书名称

请输入图书名称

作者

请输入作者

出版社

请输入出版社

出版日期

0001-01-01

价格

¥0.00

类别

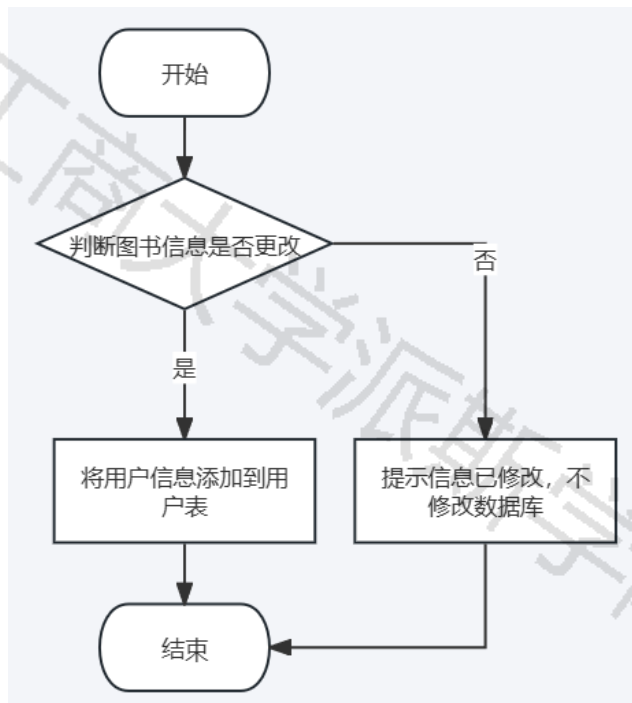
请输入类别

评价	<div></div>
状态	<div>请选择状态</div>
介绍	<div>请输入图书介绍</div>

确定添加

返回修改

修改图书：管理员选中某本图书后点击“修改”按钮，弹出编辑对话框，可更新图书信息。保存后，BookService 更新数据库中的对应记录。修改图书流程图如图5-5所示。



删除图书：管理员选中图书后点击“删除”按钮，系统弹出确认框，确认后BookService删除数据库中的图书记录。

(3) 图书查询和筛选功能

用户和管理员可以通过搜索框输入关键词，系统导入了NPinyin包，将书名、作者名转换成首字母大写和全拼实现模糊查询，结果实时显示在界面上。图5-6展示根据图书名首字母大写进行的查询。

hz

小王

所有图书列表

	ISBN	图书名	作者	出版社	出版日期	价格	类别	库存	状态	介绍	
1	978753...	活着	余华	北京十...	2025/3/7	45.00	文学	32	可借	《活着》讲述了在大...	

图5-6图书查询功能

5.3 借阅管理模块

(1) 借书功能实现

读者通过图书列表选择自身所需借阅文献，触发“借阅此书”操作，系统会优先核查图书库存状况及用户借阅资格。

当库存状态可借且用户无不良记录，系统借助BorrowRecordService新增借阅日志，且自动更新馆藏书籍数量。借阅记录流程图如5-7所示，当借阅数量超过系统设置值时的失败提示如图5-8所示。

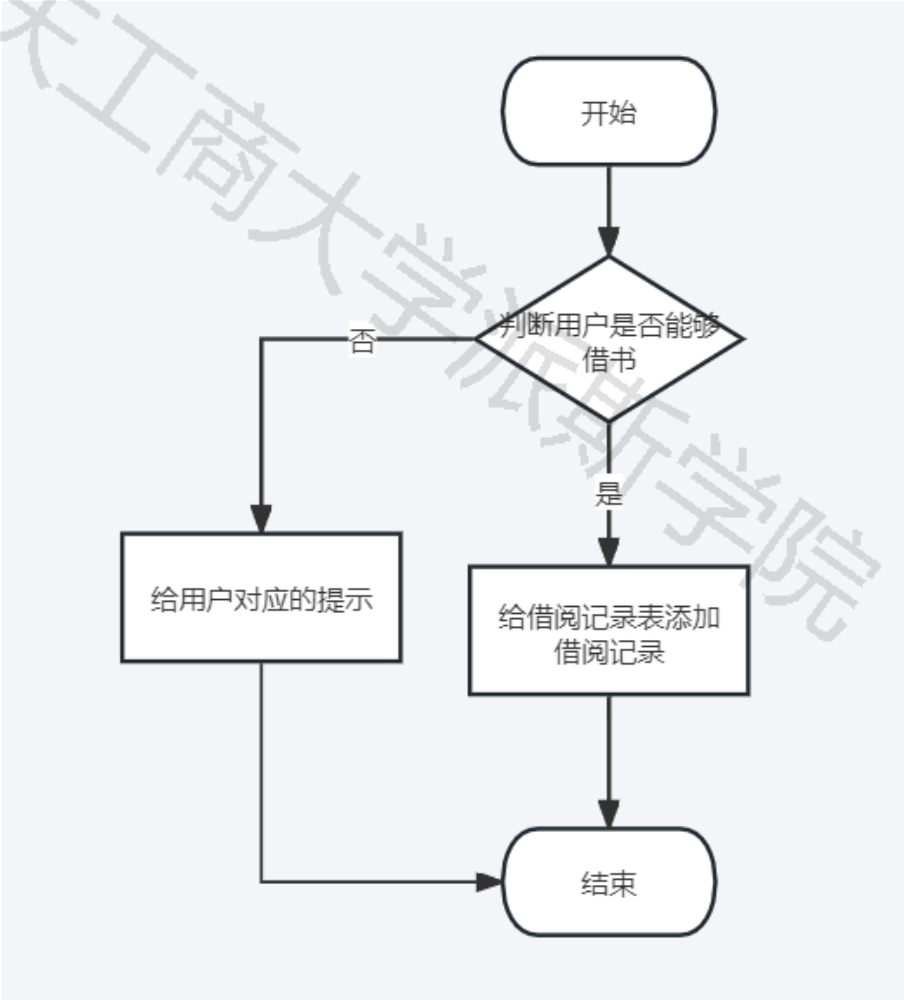


图5-7图书借阅流程图



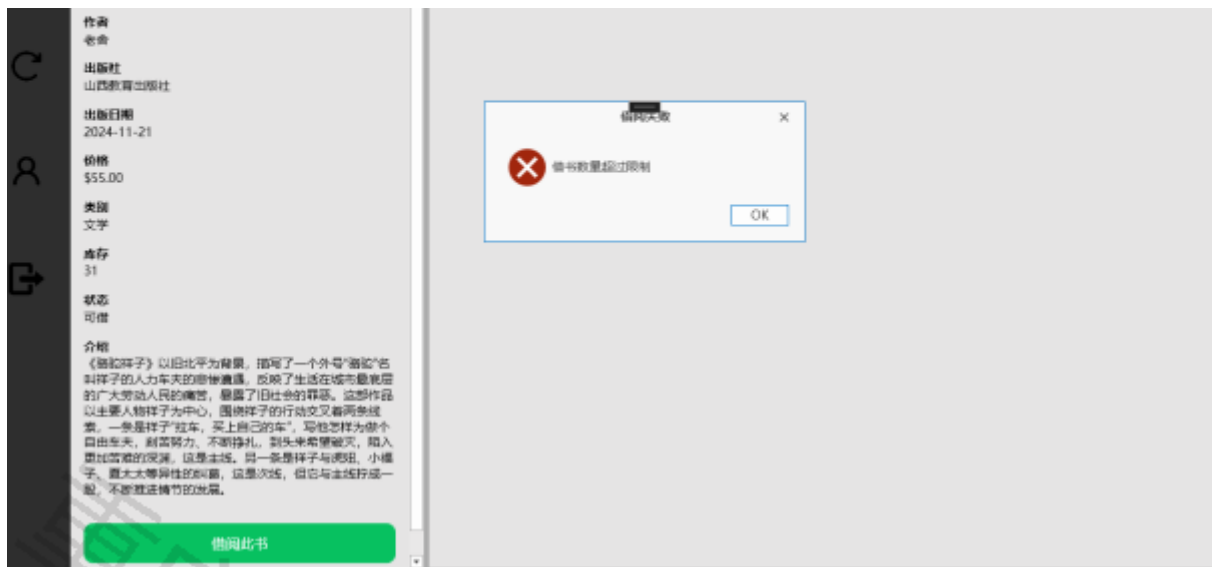
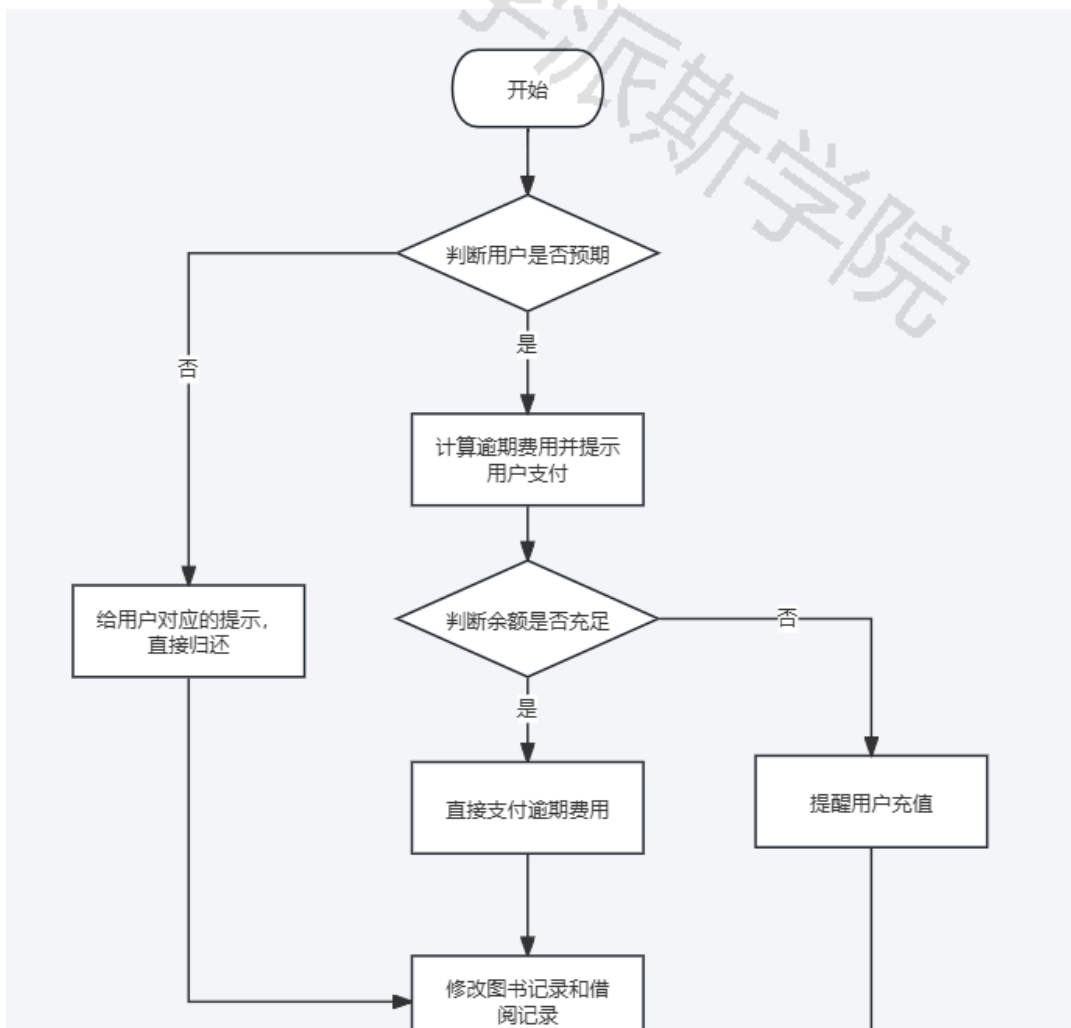


图5-8 借阅图书失败界面

(2) 还书功能实现

用户在个人借阅记录中选择已借图书，触发“还书”操作。系统借助BorrowRecordService新增借阅日志，且自动更新馆藏书籍数量。还书功能流程图如5-9所示，当余额不足并且归还时间逾期时，会提示用户需要充值余额。还书失败界面如图5-10所示。



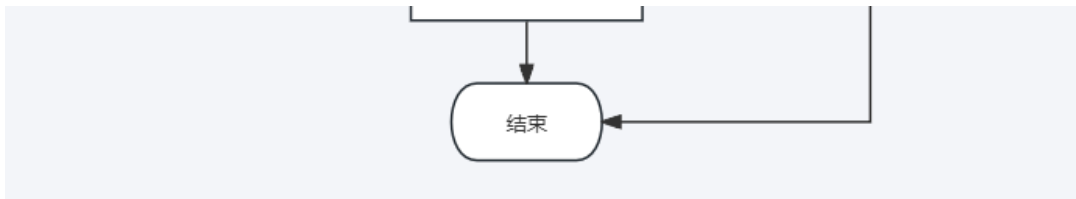


图5-9还书功能流程图

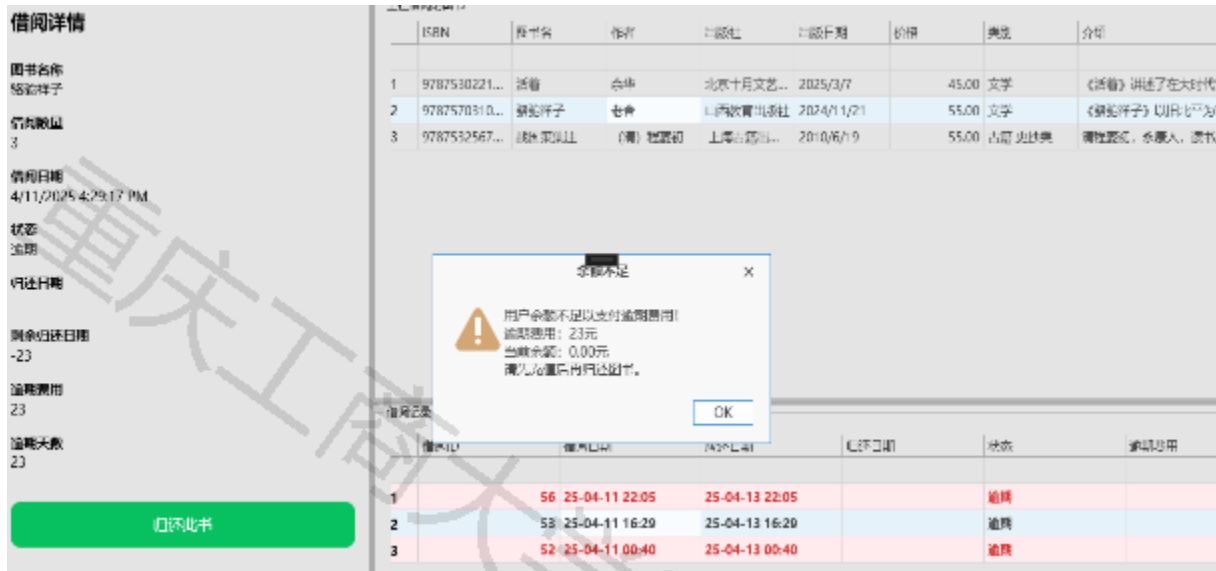


图5-10还书失败提示图

(3) 借阅记录查询功能

用户可查看个人的借阅历史记录和借阅统计数据，管理员则可查看系统中所有借阅记录和借阅统计数据。记录通过 DevExpress 的 DataGrid 控件展示，支持按日期、借阅状态等条件进行筛选。用户个人中心界面如图5-11。

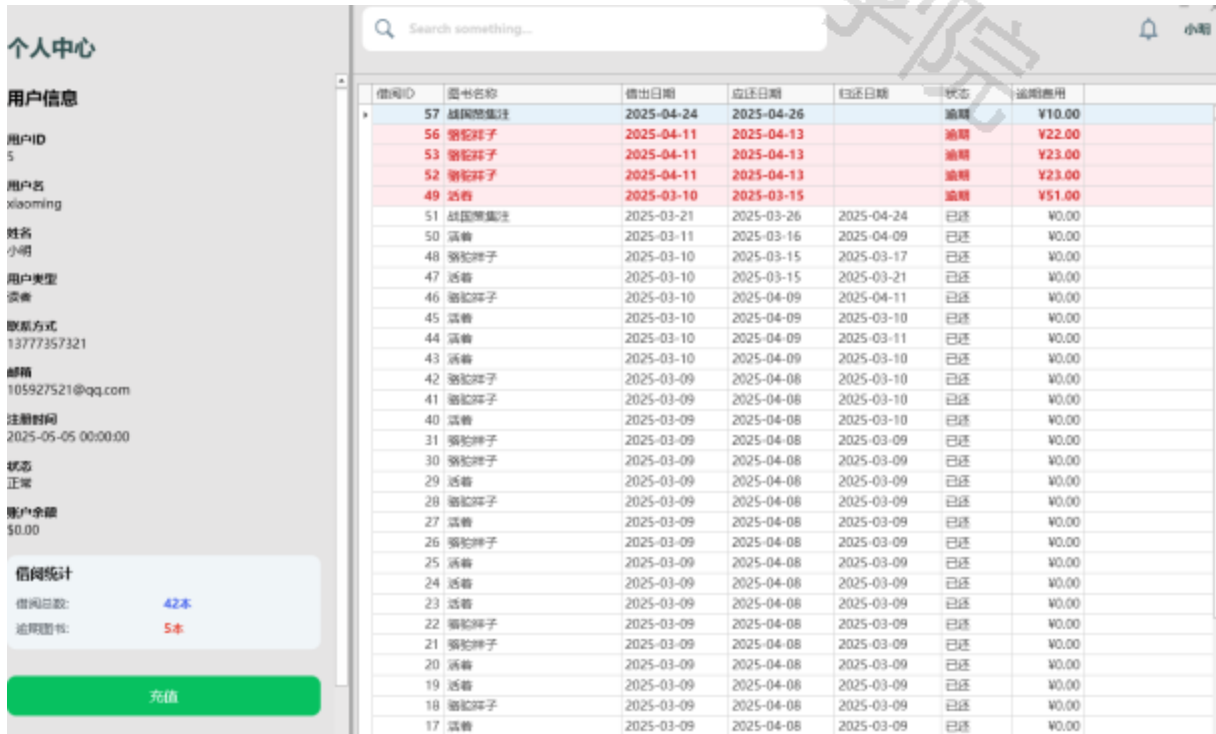


图5-11个人中心界面

5.4 用户管理模块

(1) 用户信息管理界面

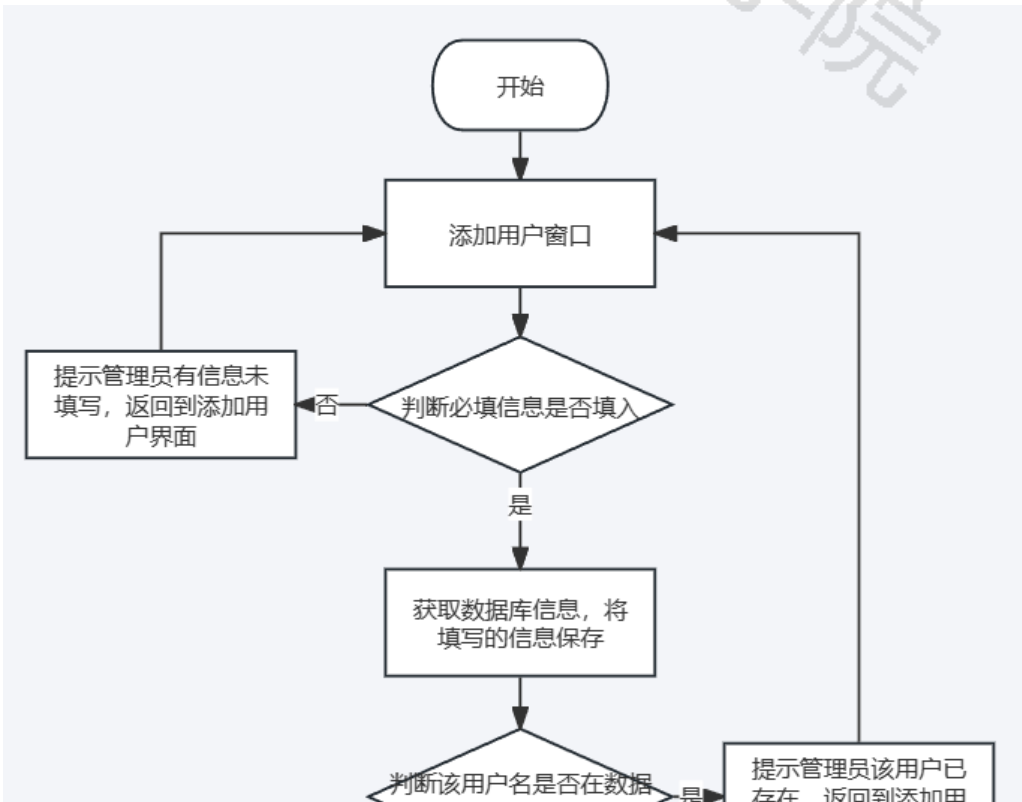


图5-12用户管理界面图

用户管理界面仅对管理员开放，采用 DataGrid 控件展示用户列表，支持搜索功能，便于管理员高效管理用户信息。

(2) 用户添加、修改、删除功能

添加用户：管理员触发“添加用户”操作，系统会弹出添加用户窗口，管理员设置新增用户信息并设置用户角色，触发“保存”操作后，系统借助UserService新增用户日志。加用户流程图如图5-13所示。



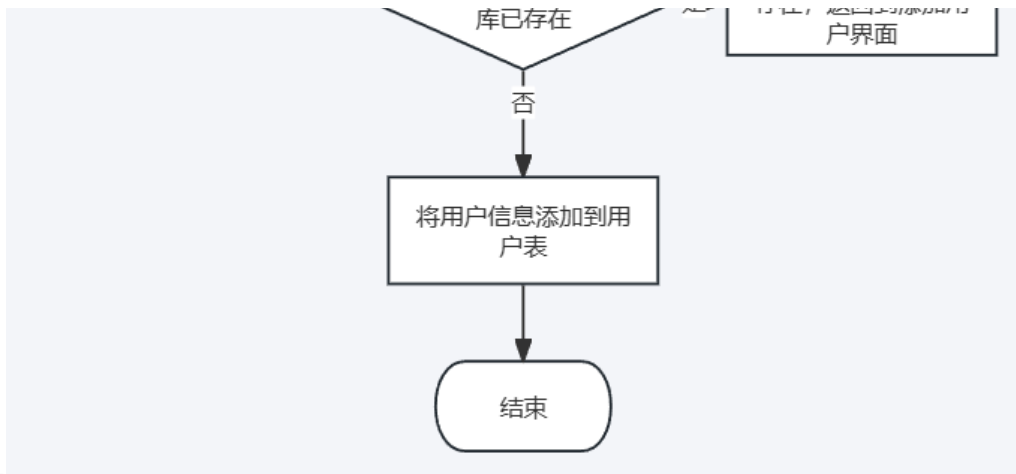


图5-13添加用户流程图

修改用户：管理员选中某用户后触发“修改”操作，可更新用户信息或角色，保存后系统借助UserService修改用户信息。修改流程图如图5-14所示。

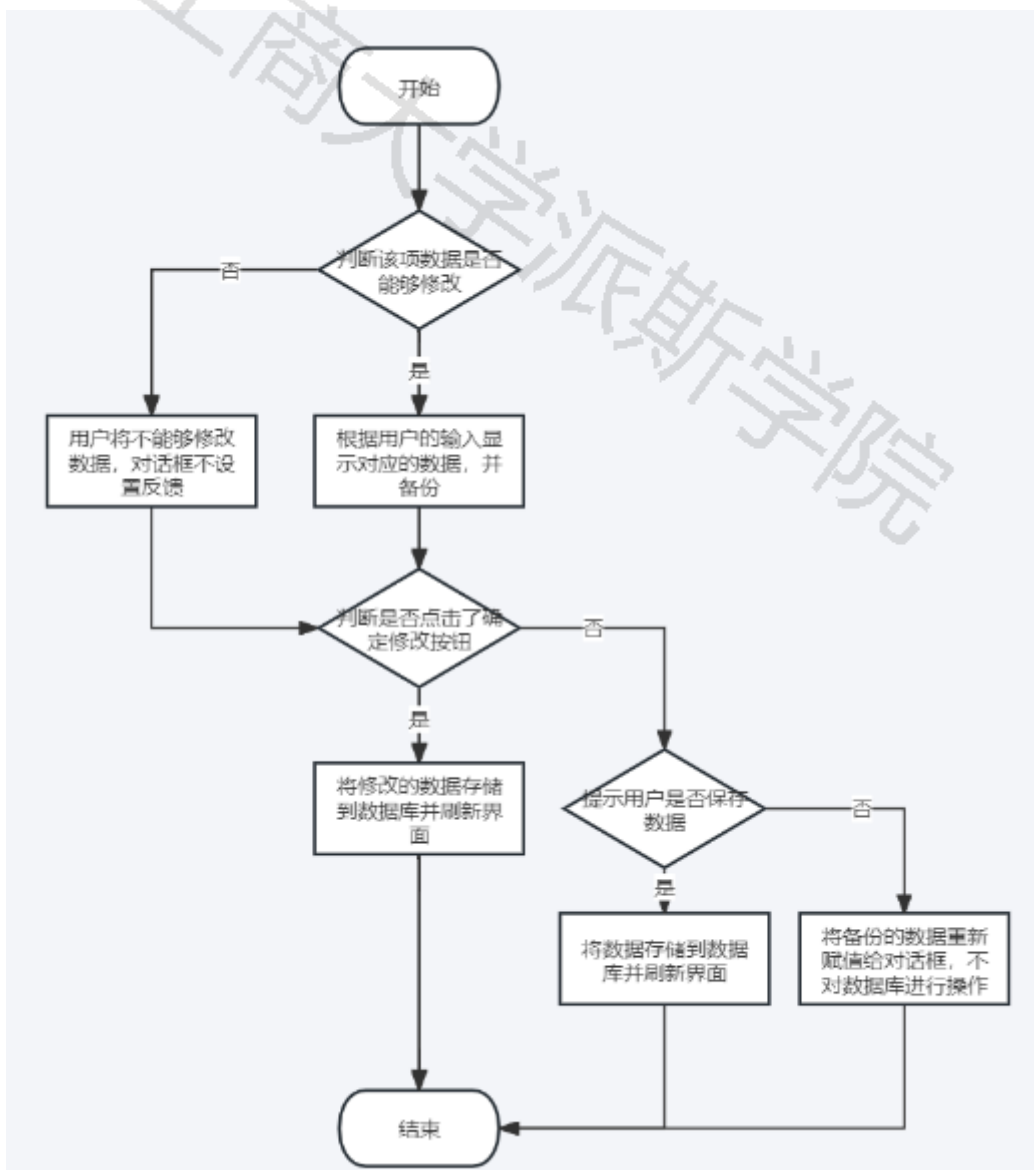


图5-14修改用户信息流程图

删除用户：管理员选中用户后触发“删除”操作，系统借助UserService删除用户信息。

(3) 用户状态管理功能

管理员可启用或禁用用户账户。禁用账户后，用户将无法登录系统，此功能通过更新用户状态字段实现。用户管理图如图5-15所示，通过下拉框设置两个可选项，前端代码如下：

```
<dx:ComboBoxEdit Grid.Row="3" Grid.Column="1"
    CornerRadius="6"
    Effect="{StaticResource ModernShadowEffect}"
    Style="{StaticResource ModernComboBoxStyle}"
    EditValue="{Binding SelectedUser.Status}"
    Margin="0,5"
    NullText="请选择用户状态">
    <dx:ComboBoxEditItem Content="正常"/>
    <dx:ComboBoxEditItem Content="禁用"/>
</dx:ComboBoxEdit>
```

通过EditValue="{Binding SelectedUser.Status}"来绑定SelectedUser的Status属性，点击确定修改后修改对应的数据库。

用户管理界面

用户信息

用户姓名

小张

用户名

xiaozhang

用户类型

管理员

用户状态

正常

正常

禁用

联系方式

邮箱

12495729@qq.com

注册时间

2025-03-07

确定修改

添加用户

图5-15修改用户信息界面图

5.5 系统设置模块

系统设置图如图5-16所示，流程图如图5-17所示。

借阅系统设置

借阅参数设置

最大借书数量:

5

借阅期限(天):

30

逾期费用(元/天):

1

图5-16系统设置图

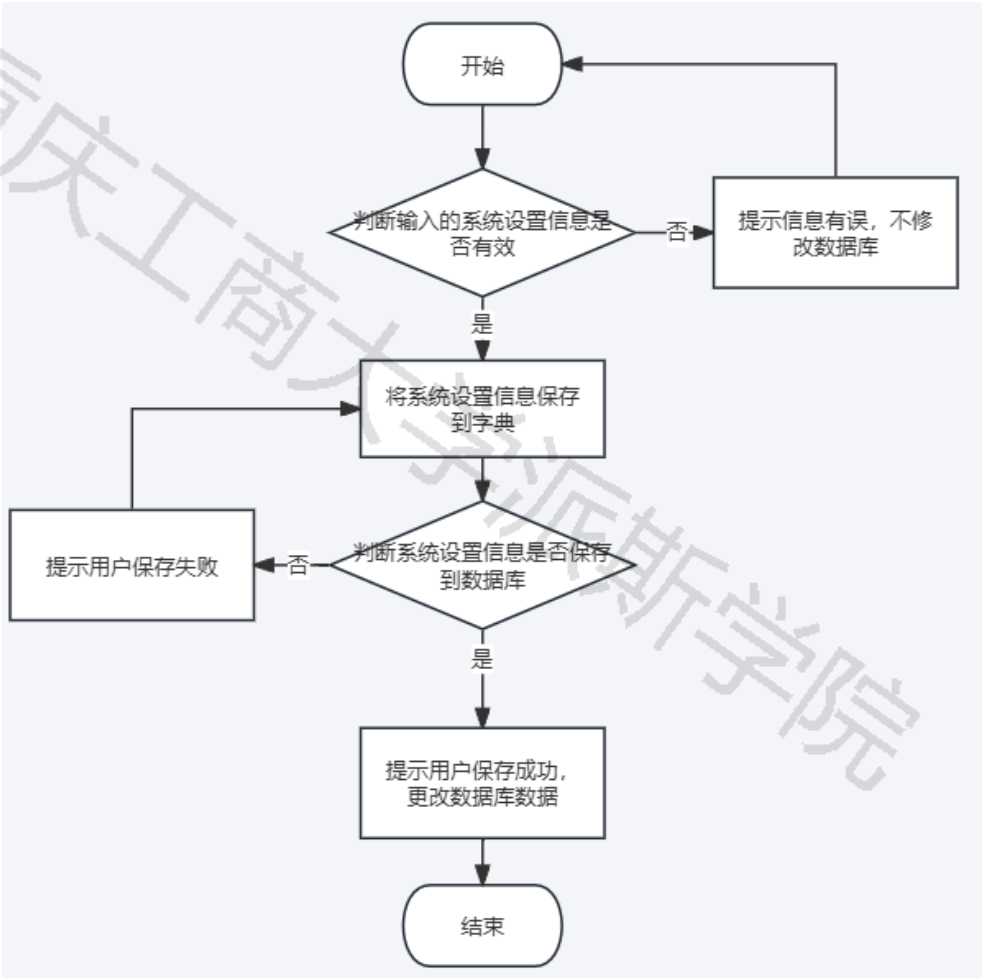


图5-17系统设置流程图

(1) 借阅规则设置功能

管理员可在系统设置界面调整借阅期限，最大借阅数量规则。这些设置保存在数据库的 `SystemSettings` 表中，系统根据这些规则判断借阅操作是否有效，例如设置的最大借阅数量为5本，该用户已经借阅了5本图书就不能够继续借阅图书了。

(2) 逾期费用设置功能

管理员可设置逾期罚款标准。用户还书时，系统自动根据借阅记录计算逾期天数和罚款金额。例如当用户归还图书时，系统会根据该用户借阅的时间和当天归还的日期来计算逾期费用，逾期费用=天数×逾期费用设置值，当用

户余额不足时提醒用户需要缴纳逾期费用。

5.6技术总结

通过以上实现，图书管理系统将需求转化为具体功能，采用了以下技术：

开发语言：作为.NET的配套语言，C#由微软在2000年6月完成开发，采用.NET框架作为支撑平台，发端于C语言流派，其设计汲取了C++和Java的强项，剥离了语言中的臃肿部分，比如去掉了宏定义与多重继承支持，其集成开发环境Visual Studio支持界面设计的即时可视化，便捷性突出。

界面开发：WPF、XAML、DevExpress。WPF功能强大，支持复杂的UI和图形处理，并且支持大部分第三方插件，本文使用的DevExpress就是其中一个非常强大的插件，并且WPF.NET Framework，与XAML语言结合，易于设计界面。

架构设计：采用MVVM模式与Prism框架，借助Prism框架可实现单元测试组件的重复利用，实现运行期间组件的即时探测与融合，把应用程序功能封装为可跨模块调用的复用单元，实现应用层功能模块的复用，该技术方案支持动态扩展，支持通过功能迭代与继承实现平滑升级，将业务逻辑与界面呈现解绑，让开发者能集中精力处理核心业务逻辑。

数据操作：SqlSugar对象映射系统，借助ORM技术可自动化生成数据库操作的代码段，降低了数据访问层的实现复杂度，减少重复劳动，客户机-服务机结构：即采用客户端与服务端的结构，客户端/服务器模式可实现较强的界面表现力与操作灵活性，可高效实施复合认证流程，采用单层交互架构，交互延迟低。

C/S结构：即客户端和服务端结构，C/S结构的界面和操作极为丰富，并且安全性很高，能够非常容易的实现多重认证，还只有一层交互，响应速度快。

这些技术的结合确保了系统的功能性、高性能和良好的用户体验。并且使得开发和后期维护都非常的便捷。

6 系统测试

常用的测试方法有两种，一种是白盒测试，一种是黑盒测试。

白盒测试亦称结构测试或逻辑驱动测试，其核心机制在于通过分析系统内部逻辑结构与运行流程，验证程序内部各组件是否严格遵循既定规范执行，重点检测程序路径的覆盖性与控制流的正确性，但不针对系统外部功能特性进行验证。该方法主要应用于软件开发周期的单元测试阶段，通常在开发人员完成主体编码与基础测试后实施。

相对的黑盒测试则被称为功能测试，其技术特征表现为忽略系统内部实现细节，仅通过输入输出数据的比对来验证功能模块的符合性，该方法不涉及源代码层面的检测，完全基于功能规格说明进行验证，因而能够确保测试过程不会干扰系统的外部行为表现。

6.1系统的测试方法

采用黑盒测试作为核心测试手段，检验结果证实，用户需求的所有功能模块均已成功部署，实施多批次整体功能检验，检出并修正了若干存在错误或与预期需求不符的功能模块，实施初步改进后，所有功能模块最终测试全部合格。

论文对下述两个功能点完成检测。

(1) 用户登录模块测试目的：分析用户登录流程的可靠性，实现用户身份的凭证验证，用户信息的真实性是本

轮测试的核心前提，详细测试用例参见表6-1。

表6-1 用户登录功能测试用例

用例编号	001	用例名称	用户登录功能测试
测试目标	测试登录功能能否正常运行		
主 要 测 试 步 骤	过程描述	预期测试结果	
	输入用户名	显示输入框正常	
	输入密码	密码以密文形式显示	
	点击登录按钮	验证成功，进入系统主界面	

（2）图书信息管理模块测试目的：分析图书信息管理功能的有效性，验证图书添加、修改、删除等功能是否能够正常完成。详细测试用例参见表6-2。

表6-2 图书添加功能测试用例

用例编号	001	用例名称	图书添加功能测试
测试目标	测试图书添加功能能否正常运行		
主 要 测 试 步 骤	过程描述	预期测试结果	
	点击添加图书按钮	显示图书信息输入界面	
	输入图书信息	信息输入框显示正常	
	点击保存按钮	数据保存成功，添加图书信息	

通过以上测试用例的执行，系统的各项主要功能均能正常运行，满足了用户的基本需求。测试过程中发现的问

题已经得到及时修正，系统整体运行稳定可靠。后续还可以根据实际使用情况，进一步完善测试用例，确保系统的长期稳定运行。

6.2 测试结果

通过以上测试，图书管理系统在功能、性能、安全性和用户体验方面表现出色。单元测试确保了各模块的独立功能正确，集成测试验证了模块间的协同工作，系统测试则全面评估了系统的整体质量。测试过程中发现的问题已及时修复，确保系统稳定可靠。

系统测试完成后，图书管理系统已准备好投入使用，满足了设计要求和用户需求，为图书馆管理提供了高效、便捷的解决方案。

7 总结与展望

本论文从实际的图书馆需求出发，研发出一个有助于图书管理、实用的系统，能够满足基本的图书管理，日常使用。后续还可以进一步对系统进行开发维护。

7.1 工作总结

本项目成功设计并实现了一套功能完善、性能优越的图书管理系统，旨在解决传统图书管理方式中效率低下、错误率高等问题，同时提升用户体验。以下是对开发工作的详细总结：

（1）系统开发成果

系统实现了以下核心功能，满足了图书馆管理的基本需求：用户管理，图书管理，借阅管理，系统设置。

通过采用实用的技术WPF、MVVM、Prism 和 SqlSugar，系统具备良好的可维护性、可扩展性和高性能。

（2）关键技术应用

系统开发中使用了多项关键技术，确保了功能的实现和系统的稳定性：

WPF：提供现代化的用户界面，支持丰富的交互体验。

MVVM架构：实现界面与逻辑分离，提升代码可维护性和可测试性。

Prism框架：支持模块化开发，方便后期维护以及更新。

SqlSugar ORM：简化数据库操作，防止SQL注入攻击，提高数据访问效率。

DevExpress UI 控件库：提供高级控件，优化用户体验。

SHA256 加密：保障用户密码的安全性。

（3）主要功能实现

用户管理：包含用户注册、登录及权限配置模块。

图书管理：完成图书数据的增删改查及检索功能实现。

借阅管理：支持图书的借出、回收流程。

查询功能：支持图书检索与借阅历史查询服务。

系统设置：系统支持由管理员主导的借阅规则与逾期费用配置。

（4）遇到的问题及解决方案

在开发过程中遇到了一些挑战，并通过以下方式解决：

问题：在数据测试中数据库无法进行修改数据。

原因是因为传入数据库的值与数据库的值不符合，使用ORM对数据库进行操作需要使用AutoMapper映射，这样才能正确的一一对应数据，将数据传入数据库。

7.2 展望

尽管系统已实现预期功能，但为适应未来需求和提升用户体验，仍有改进和扩展的空间。以下是未来的发展方向：

（1）功能扩展

图书推荐：引入推荐算法，根据用户借阅历史推荐相关图书。

读者评价：允许用户对图书评分和评论，增强互动性。

电子图书管理：支持电子图书阅读，适应数字化趋势。

移动端应用开发：开发移动应用，方便用户随时随地访问系统。

（3）用户体验改进

界面设计：优化布局和视觉效果，提升美观度。

操作流程：简化步骤，减少用户操作负担。

多语言支持：提供多语言界面，满足不同用户需求。

通过本次开发，图书管理系统实现了预期目标，还为未来扩展和优化奠定了基础。但是目前的功能还是有所欠缺不够完善，还有许多可以更新扩展的地方。随着技术进步和用户需求变化，系统将持续改进，为图书馆管理提供更高效、便捷的服务。

致谢

时光飞逝，转眼间我的本科学习生涯即将结束。在这里，我要向在毕业设计过程中给予我帮助和指导的老师、同学以及家人表示最诚挚的感谢。

首先，我要特别感谢我的指导老师，在整个毕业设计期间，她始终以严谨的治学态度和渊博的专业知识指导我完成这项工作。她不仅在选题方向、技术难点等方面给予了我悉心指导，更教会了我如何以科研的态度处理问题，这些都使我受益匪浅。

其次我要感谢我的家人，他们一直在背后给予我支持和帮助，在我求学期间给予的理解和支持，让我能够专心致志地完成学业。

参考文献

- [1] 闫童. 基于BIM的高校图书馆运维管理应用研究[D]. 华北理工大学, 2024.
- [2] 李超然. 面向智能书架的RFID图书信息化管理系统开发[D]. 黑龙江大学, 2024.
- [3] 李鹏. 扬州大学多校区图书馆管理研究[D]. 扬州大学, 2024.
- [4] 张滢月. 高校图书管理系统设计与实现[D]. 西安工程大学, 2019.
- [5] 李婷婷, 张金玉, 马凤乾. 基于ORM的数据管理序列化实现方法[C]//天津市电子工业协会. 天津市电子工业协会2024年年会论文集. 天津七一二通信广播股份有限公司; , 2024:245-249. DOI:10.26914/c.cnkihy.2024.009945.
- [6] 吴香艳. 基于C#和.NET的班级学生信息管理系统设计与实现[J]. 无线互联科技, 2023, 20(22):47-50.
- [7] 王毅铭. 单向HASH函数下的密钥矩阵加密方法分析. 广西大学国际学院, 2024.
- [8] 赵云正. 基于RSA/SHA-256加解密协处理器的研究与设计[D]. 电子科技大学, 2022.

- [9]高鑫. 基于语法分析的SQL注入攻击检测的研究[D]. 大连理工大学, 2024.
- [10]许家玥. 基于多模安全共识的SQL入侵检测方法研究[D]. 中国电子科技集团公司电子科学研究院, 2024.
- [11]罗艺铭. 基于生成对抗网络的SQL注入攻击检测方法研究[D]. 河南工业大学, 2024.
- [12]徐杰铭. 顾及图书错位查询功能的数字图书馆系统的设计与实现[D]. 合肥工业大学, 2022.
- [13]魏秀丽. 基于协同过滤的高校图书馆个性化图书推荐系统[D]. 首都经济贸易大学, 2021.
- [14]刘磊. 基于中文图书自动分类的图书管理系统的研究与实现[D]. 江西师范大学, 2021.

须知：

- 报告编号系送检论文检测报告在本系统中的唯一编号
- 本报告为维普论文检测系统算法自动生成，仅对您所选择比对资源范围内检验结果负责，仅供参考。



微信公众号

唯一官网：<https://vpcs.fanyu.com> | 客服邮箱：vpcs@fanyu.com | 客服热线：400-607-5550 | 客服QQ：4006075550