文件编号：IDO - ProjectPlan - 0420

受控状态：■受控 □非受控

保密级别：□公司级 □部门级 ■项目级 □普通级



“IDO”智能图书管理系统

“IDO”Intelligent Book Management System



项目系统设计报告

Version 2.2



**Written by BM4P**

**2023.04.20**

**All Rights Reserved**

# 文件信息

|  |  |  |  |
| --- | --- | --- | --- |
| 文件状态： | [ ] 草稿文件  [√] 正式文件  [ ] 更改正式文件 | 当前版本： | V2.2 |
| 作 者： | 鄢豪、陈思城 |
| 审 核： | 李超钒 |
| 完成日期： | 2023-4-20 |
| 文档编号： | IDO - ProjectPlan - 0420 | 文档标题： | IDO智能图书管理系统项目系统设计报告 |
| 文档类别： | 项目系统设计报告 | 提交人员： | 李超钒 |
| 文 件 名： | 项目系统设计报告 | | |
| 文件摘要： | 详细介绍系统如何设计、系统的技术路线 | | |
| 项目名称： | I DO 智能图书管理系统 | | |
| 当前阶段： | 项目设计阶段 | | |
| 版权所有： | BM4P小组 | | |

# 修改历史

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 日期 | 版本 | 作者 | 修改内容 | 评审者 | 更改请求者 |
| 2023-03-10 | V1.0 | 鄢豪 | 定义文件模板、基本内容 | 李超钒 | 陈思城 |
| 2023-03-20 | V1.1 | 杨松林 | 补充接口设计 | 陈思城 | 杨松林 |
| 2023-04-01 | V2.0 | 李超钒 | 修改并填充概要设计和详细设计主要内容 | 鄢豪 | 李超钒 |
| 2023-04-10 | V2.1 | 陈思城 | 补充技术细节 | 左鹏杰 | 鄢豪 |
| 2023-04-20 | V2.2 | 鄢豪 | 补充运行界面 | 李超钒 | 鄢豪 |

目录

[文件信息 2](#_Toc138126626)

[修改历史 2](#_Toc138126627)

[1.引言 6](#_Toc138126628)

[1.1目的 6](#_Toc138126629)

[1.2 适用范围 6](#_Toc138126630)

[1.3项目定位 6](#_Toc138126631)

[1.3.1应用场景 6](#_Toc138126632)

[1.3.2目标人群 6](#_Toc138126633)

[1.4项目目标 6](#_Toc138126634)

[1.4.1读者模块 7](#_Toc138126635)

[1.4.2管理员模块 8](#_Toc138126636)

[1.5项目价值 9](#_Toc138126637)

[1.6创新点 10](#_Toc138126638)

[1.6.1功能性创新 10](#_Toc138126639)

[1.6.2非功能性创新 10](#_Toc138126640)

[1.7解决思路 11](#_Toc138126641)

[2.概要设计 11](#_Toc138126642)

[2.1处理流程 11](#_Toc138126643)

[2.2总体结构和模块设计 13](#_Toc138126644)

[2.2.1读者模块 14](#_Toc138126645)

[2.2.2管理员模块 15](#_Toc138126646)

[2.3功能分配 16](#_Toc138126647)

[2.4接口设计 17](#_Toc138126648)

[2.4.1外部接口 17](#_Toc138126649)

[2.4.2内部接口 18](#_Toc138126650)

[2.4.3 管理员Web端界面设计 19](#_Toc138126651)

[2.4.4读者web端界面设计 26](#_Toc138126652)

[2.5数据结构设计 32](#_Toc138126653)

[2.5.1逻辑结构设计 32](#_Toc138126654)

[2.5.2物理结构设计 33](#_Toc138126655)

[2.5.3数据定义 34](#_Toc138126656)

[2.6运行设计 37](#_Toc138126657)

[2.6.1运行模块关系 37](#_Toc138126658)

[2.6.2运行控制 37](#_Toc138126659)

[2.6.3运行时间 37](#_Toc138126660)

[2.7错误/异常处理设计 37](#_Toc138126661)

[2.8数据恢复方案 38](#_Toc138126662)

[2.8.1 全崩溃恢复机制 38](#_Toc138126663)

[2.8.2 服务器崩溃恢复机制 39](#_Toc138126664)

[2.8.3 磁盘阵列崩溃恢复机制 39](#_Toc138126665)

[2.8.4系统软件全崩溃恢复机制 39](#_Toc138126666)

[2.8.5 操作系统崩溃恢复机制 39](#_Toc138126667)

[2.8.6 数据库系统崩溃恢复机制 40](#_Toc138126668)

[2.9数据备份方案 40](#_Toc138126669)

[2.10系统配置策略 41](#_Toc138126670)

[2.10.1客户端 41](#_Toc138126671)

[2.10.2服务器端目前 41](#_Toc138126672)

[2.11系统部署方案 41](#_Toc138126673)

[2.12规范 41](#_Toc138126674)

[3.详细设计 41](#_Toc138126675)

[3.1 管理员端 42](#_Toc138126676)

[3.1.1信息安全模块 42](#_Toc138126677)

[3.1.2输入 42](#_Toc138126678)

[3.1.3输出 42](#_Toc138126679)

[3.1.4程序逻辑 42](#_Toc138126680)

[3.1.6接口 45](#_Toc138126681)

[3.1.7存储分配 45](#_Toc138126682)

[3.1.8限制条件 45](#_Toc138126683)

[3.2图书信息管理模块 46](#_Toc138126684)

[3.2.1功能描述 46](#_Toc138126685)

[3.2.2性能描述 46](#_Toc138126686)

[3.2.3输入 46](#_Toc138126687)

[3.2.4输出 46](#_Toc138126688)

[3.2.5程序逻辑 46](#_Toc138126689)

[3.2.6接口 48](#_Toc138126690)

[3.2.7存储分配 49](#_Toc138126691)

[3.3咨询回复模块 49](#_Toc138126692)

[3.3.1功能描述 49](#_Toc138126693)

[3.3.2性能描述 49](#_Toc138126694)

[3.3.3输入 49](#_Toc138126695)

[3.3.4输出 49](#_Toc138126696)

[3.3.5程序逻辑 50](#_Toc138126697)

[3.3.6接口 50](#_Toc138126698)

[3.3.7存储分配 50](#_Toc138126699)

[3.4逾期处罚模块 50](#_Toc138126700)

[3.4.1功能描述 50](#_Toc138126701)

[3.4.2性能描述 51](#_Toc138126702)

[3.4.3输入 51](#_Toc138126703)

[3.4.4输出 51](#_Toc138126704)

[3.4.5程序逻辑 51](#_Toc138126705)

[3.4.6接口 51](#_Toc138126706)

[3.4.7存储分配 52](#_Toc138126707)

[3.5读者端 52](#_Toc138126708)

[3.5.1信息安全模块 52](#_Toc138126709)

[3.5.2性能描述 52](#_Toc138126710)

[3.5.3输入 52](#_Toc138126711)

[3.5.4输出 53](#_Toc138126712)

[3.5.5程序逻辑 53](#_Toc138126713)

[3.5.6接口 55](#_Toc138126714)

[3.5.7存储分配 55](#_Toc138126715)

[3.5.8限制条件 55](#_Toc138126716)

[3.6图书借阅管理模块 56](#_Toc138126717)

[3.6.1功能描述 56](#_Toc138126718)

[3.6.2性能描述 56](#_Toc138126719)

[3.6.3输入 56](#_Toc138126720)

[3.6.4输出 56](#_Toc138126721)

[3.6.5程序逻辑 56](#_Toc138126722)

[3.6.6接口 58](#_Toc138126723)

[3.6.7存储分配 58](#_Toc138126724)

[3.7聊天咨询模块 58](#_Toc138126725)

[3.7.1功能描述 58](#_Toc138126726)

[3.7.2性能描述 59](#_Toc138126727)

[3.7.3输入 59](#_Toc138126728)

[3.7.4输出 59](#_Toc138126729)

[3.7.5程序逻辑 59](#_Toc138126730)

[3.7.6接口 59](#_Toc138126731)

[3.7.7存储分配 60](#_Toc138126732)

[3.8便捷服务模块 60](#_Toc138126733)

[3.8.1功能描述 60](#_Toc138126734)

[3.8.2性能描述 60](#_Toc138126735)

[3.8.3输入 60](#_Toc138126736)

[3.8.4输出 60](#_Toc138126737)

[3.8.5程序逻辑 60](#_Toc138126738)

[3.8.6接口 61](#_Toc138126739)

[3.8.7存储分配 61](#_Toc138126740)

# 1.引言

## 1.1目的

《IDO智能图书管理系统系统设计报告》是本项目开发、使用、维护的必备资料，它能够帮助团队进行有效的沟通，其描述了系统的整体结构、各个组件的功能和相互关系、数据流程、算法选择等，帮助开发者保证了开发的高效性和准确性，同时在软件的使用过程中有指导，帮助，解惑的作用，尤其在维护工作中，文档是不可或缺的资料。本文用于指导团队开发软件项目的过程，通过规范软件项目的开发过程达到提高软件质量，降低维护成本的目的。

## 1.2 适用范围

开发的系统的名称为：IDO（爱读） 智能图书管理系统。

本《系统设计报告》描述了系统的目标和适用范围，同时对于功能进行了详细的解释，展现了技术实现路线及方案、价值体现和解决思路，本文档可用于BM4P开发人员之间进行交流沟通，进一步了解系统的意义和实现流程，以便适用更低的成本位用户带来更好的技术和服务，保证完善了系统的信息化、完整化以及模化，同时，本文的旨在为BM4P团队提供进一步二点明确需求，以便更好的提供算法和技术支持以及完善功能

## 1.3项目定位

### 1.3.1应用场景

本产品的主要服务对象针对签订合作的大学和大学生，同时也可用于省市签订的图书馆当中，主要是为大学生和其他用户提供更便捷、更强大的图书管理系统，方便日常图书的借阅、归还等一系列操作。本产品希望尽可能多的在各地得到应用，形成自己的产品生态，为各类大学提供智能化、简便化、完整化的图书管理服务。

### 1.3.2目标人群

系统管理员：系统管理员负责维护系统正常运行、针对图书借阅、管理、归还、录入、公告处理、读者咨询、图书报表统计、图书推荐、图书互借、账户管理、违规处罚等

大学学生：智能借阅图书、针对于图书借阅、归还、推荐、互借、逾期处罚等多项功能

## 1.4项目目标

经过对各学校以及各个图书馆的常客进行需求调研，并且实地对图书馆进行了考察，将目前市场上现有的各类产品进行对比分析，直击目前市场的痛点，我们将未来所设计的功能划分为三大模块，分别是读者模块、管理员模块和系统模块，其中针对于图书的扫码归还、图书推荐等方面作为我们进行项目开发的主要目标，具体方面如下所示：

### 1.4.1读者模块

**账号管理**：

该模块主要是用户对于自己个人信息的管理、用户可以通过邮箱或者手机号进行注册、收到验证码信息之后，验证成功方可注册成功，登录I DO智能图书管理系统后，用户可以对自己的密码进行修改、同样也需要进行邮箱或者手机号的验证，验证成功，方可修改，同时用户还可以对自己的昵称进行修改，并且可以对账户中的余额进行充值等多项功能

**图书借阅：**

该模块主要是用户对图书馆内的图书进行借阅，用户点击书籍可以查看书籍的详细信息，包含书籍的作者、出版社、描述等信息的解释，用户进行借阅时，会将借阅信息传给管理员，管理员核验用户资格后，发送借阅二维码给用户，用户凭借此二维码进行书籍的借阅、如果用户借阅数量超出限度，则会收到信息告知借阅数量超额。

**图书归还：**

该模块主要是用户对于自己已经借阅的图书进行归还，用户在自己的系统当中查看已借阅书籍、即可观察到自己借阅的图书，用户点击归还图书会自动生成二维码，用户将书籍二维码通过管理员扫码成功后，核验书籍后归还成功。

**图书推荐展示模块：**

系统会通过机器学习的算法将用户喜好的图书进行推荐、于此同时系统也会通过滑动窗口为用户展示今日新书以及目前畅销热门书籍供应用户进行选择借阅。

**图书互借模块：**

该模块用户可以将自己的书籍进行上传到图书馆当中，通过图书馆将自己的书籍外借给他人，若他人借阅用户上传的图书，用户的账户中将收取到一定的费用，用户需要借阅他人上传书籍时，同样也现需要对于图书借阅功能的使用，最后进行扣款，扣款成功后，图书互借成功。

**违规处罚模块：**

该模块系统会筛选出书籍借阅逾期的读者和接近截止其的读者，，并将逾期信息发送给读者，读者在自己的信息栏当中可以接收到书籍借阅逾期的提示，提示用户需要进行图书归还，用户对于自己借阅逾期的图书需要向图书馆支付每日0.1元的处罚赔款、并且管理员需要核验图书是否完整，完整则归还成功，不完整则需要用户进行照价赔偿。

**在线阅读模块：**

该模允许用户对目前系统所有录入的图书进行在线阅读、通过初期的阅读，用户可以判断自己对该书籍是否喜爱，以此来进行借阅，进入图书详情界面，点击在线阅读即可进行在线阅读。

**人工智能聊天推荐模块：**

该模块系统使用当今流行的OPENAL——ChatGPT，用户可以针对本图书馆的信息以及个人借阅喜好、图书推荐等多方面进行功能进行询问，避免了新的用户前往图书馆时，时时需要前往管理员处进行询问信息的繁琐过程，增加了本图书管理系统的智能性。

### 1.4.2管理员模块

**用户信息管理模块：**

该模块管理员可以对目前所有注册的用户信息进行统一的管理，可以帮助用户进行修改账户、密码、手机号以及邮箱等多方面的功能，还可以对用户的在线情况和用户的账户进行管理，同时管理员拥有权限对用户进行注销操作，管理员同时可以对自己的信息进行录入、修改、查询等一系列操作。

**图书信息维护模块：**

该模块管理员可以对整个图书馆的图书信息进行维护管理，可以查看目前图书馆当中录入的所有图书，以及图书所在书架位置，方便管理员进行管理，同时系统会实时进行报表的处理，按月对图书借阅信息进行统计和处理分析，供图书管理员查询，并依据此来进行资源配置，同时图书管理员可以对图书进行录入、修改、删除等一系列图书相关操作。

**借还管理模块：**

该模块管理员可以对用户进行借书和还书的请求进行处理，通过判断用户是否可以进行借阅图书，可以，则发送二维码，不可以，则发送失败原因，针对用户归还图书，则是进行图书逾期和完整性判断，通过二维码扫描，成功之后归还成功。

**逾期处理模块：**

该模块管理员可以对逾期以及接近逾期的图书进行逾期警告，针对以及逾期的用户，通过判断其逾期时长对其进行信誉评价以及相关罚款处理、对于信誉评价较低的用户将降低其可借图书的数量

**图书互借模块：**

该模块管理员可以将用户对于申请上传的图书进行审核，审核成功之后将其录入图书馆当中，当其他用户需要借阅此图书时，向管理员发起申请，管理员接收请求，同意借书后，将收款码发送给该用户进行支付。

**系统模块：**

**读者咨询：**

该模块系统为两边用户提供通信服务，用户可以针对自己所需求的信息向管理员进行提问，同时管理员也可以将用户所需要的信息发送给用户，实现采用内网穿透的方式，实现异网的实时通信。

**信息公告：**

该模块系统为两边用户提供信息公告栏模块，管理员可以将自己对于新款图书上架等信息放入公告栏当中，供用户进行查看阅读，同时也满足用户对于公告栏进行评论留言，可以将自己对于图书的诉求留言在信息公告栏中，给管理员进行查看。

## 1.5项目价值

本项目价值主要体现在以下几方面：

对图书管理人员而言，图书馆拥有了一个全新的、更加完整的、便捷的、与众不同的智能化的图书管理系统。通过将该系统进行推广可以方便各个大学的图书馆的管理以及省市或者私人图书馆的管理，同时可以为各个图书馆带来更加客观的人流量，增加经济效益，该系统顺应了当今智能化的时代的潮流，结合机器学习、chatgpt等多方智能化方法及产物，有效的提高了图书管理的效率，同时也是对于从前老旧图书馆的技术进行一次变革创新，符合了当局潮流的发展宗旨，可有效的为图书馆带来人流量和新的热度浪潮。

对于读者而言，本系统提供了不同于以往老旧的图书管理系统等繁琐复杂的功能，将借阅图书使用当今潮流的二维码进行借阅，方便了借阅证以及借阅信息的保存，保证了信息的安全性，同时开通更多的新式图书馆功能，例如读者咨询、图书互借、信息公告、智能IDO聊天等多方面的适应当今时代的功能，同时使用机器学习的算法为读者提供了更符合其喜好的图书推荐、使用爬虫的方法将图书阅读信息线上化，实现了于当今网络阅读的潮流，为读者用户带来更加便捷、完善的服务

综上所述：“I DO 智能图书管理系统”实现了一个模式顺应潮流、功能创新、技术革新、交互性增强、安全性提高的智能化的图书管理系统，为图书管理多方面内容提供了智能化的管理，提高了读者的借阅体验，改善了管理员的管理繁杂，顺应智能化、机器学习等多方面的时代潮流，为大众打造出更加完善、便捷的智能化图书管理系统

## 1.6创新点

### 1.6.1功能性创新

IDO智能图书管理系统在功能上创新引入了在线阅读、IDO智能聊天、读者互借、读者咨询、信息公告栏、图书推荐等功能。

表1-1：功能性创新

|  |  |
| --- | --- |
| 功能名称 | 功能说明 |
| 在线阅读 | 可在系统当中对于图书馆已有图书进行线上阅读 |
| I DO智能聊天 | 为用户提供了类似于ChatGPT的人工智能聊天的功能，可进行图书推荐、书籍位置查询等信息的询问。 |
| 读者互借 | 用户可以通进行图书上传，将自己的书籍借阅给其他读者进行阅读，拓宽了图书信息的来源路径 |
| 读者咨询 | 为用户提供与管理员进行实时聊天通信交流 |
| 信息公告栏 | 为读者了解当前图书馆现状和提出诉求提供了功能，为管理员满足读者诉求提供了功能 |
| 图书推荐 | 帮助读者进行机器学习更加准确的图书推荐以求找到自己心仪的图书 |

### 1.6.2非功能性创新

表1-2：非功能创新

|  |  |
| --- | --- |
| 技术突破 | 功能说明 |
| 使用二维码进行借还书 | 采用微信提供的API实现二维码借还书保证了信息的安全性和便捷性 |
| 模型轻量化 | 采用Flask轻量级框架实现整个代码 |
| 提高数据备份速率 | 系统当中提供了三套数据备份的方案，双击冗余、用户备份以及云备份，可以根据不同的需求自主的选择备份方案。 |
| 减少恢复处理时间 | 采用异地备份的方案，以及数据库端的实时备份。 |
| 提供实时聊天的速率 | 采用内网穿透进行，保证了信息的传输速率 |
| 提高数据访问速度 | 系统采用当今大规模数据使用的数据库Redis进行数据备份，提高了数据的访问速度 |

## 1.7解决思路

本项目基于web端开发，Flask框架、MysQL数据库、NUXT框架、ChatGptAPI。

·用轻量级别的web开发框架，简化了代码的实现、确保了代码的流程和整洁、模块清晰等。

·使用成熟的web开发工具，保证了应用软件具有友好的用户交互界面和完善的功能以及更为良好的体验

·采用大数据集MySQL+Redis数据库，将数据存储在云端，保证了数据的安全性和高效性

·平台的扩展性和灵活性，能够嵌入新的功能模块，降低应用的复杂度。

·建立稳定的项目整体设计团队，充分发挥团队成员的优势，对项目设计过程进行严格控制，保证过程的规范性和可靠性。

**技术解决思路如下：**

根据功能架构的分析，本项目采用基于flask框架和NUXT框架的Python开发。

# 2．概要设计

## 2.1处理流程

IDO智能图书管理系统是用于读者和管理员对图书馆中书籍信息进行操作使用，具备图书互借、违规处理、信息专栏、在线阅读、智能聊天、信息维护、图书借阅、图书归还、图书报表和图书推荐等功能用户和通过计算机终端（手机、平板、电脑等）进入系统，经过系统查验后成功登入系统，不同用户可以对系统选择不同的操作，如读者可以进行推荐书籍查看、智能聊天机器人等功能，管理员可以进行图书报表的查看和图书录入等功能。



图2-1系统架构图

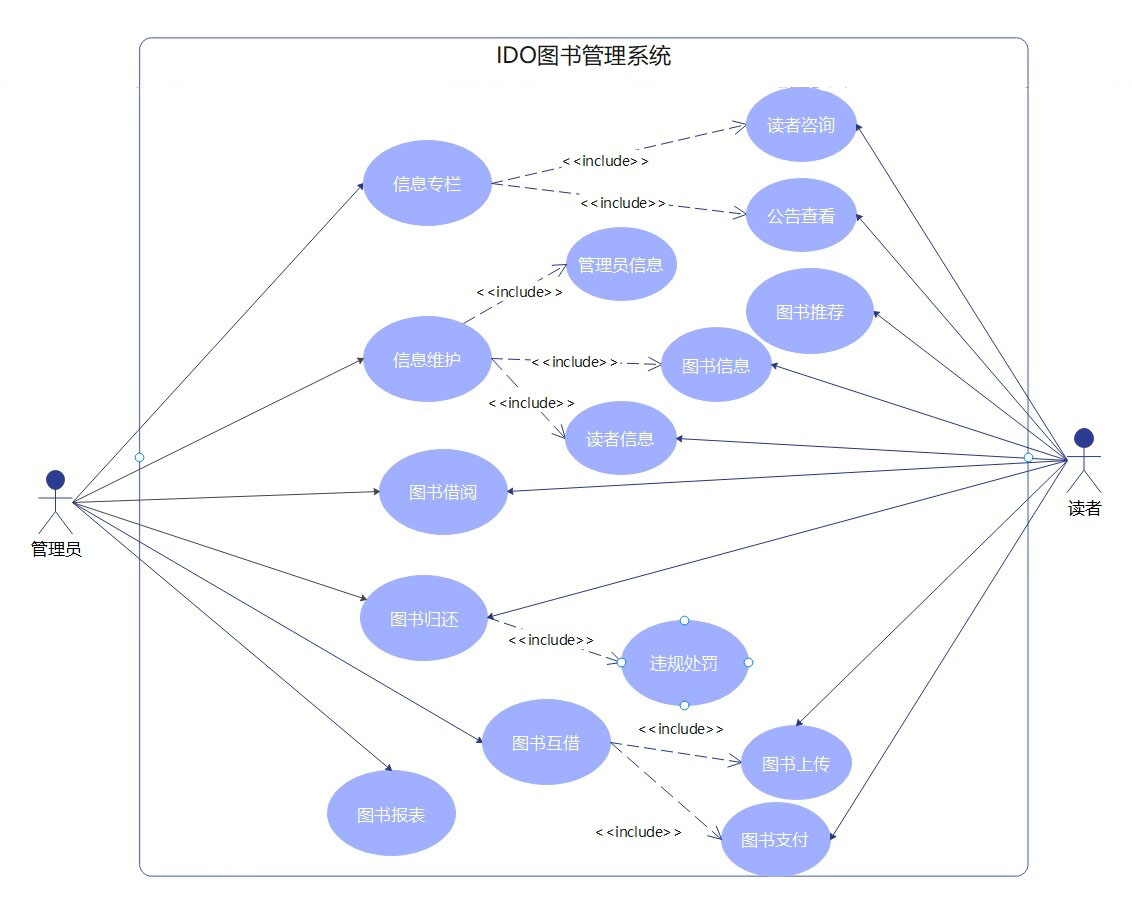


图2-2 用例图

## 2.2总体结构和模块设计

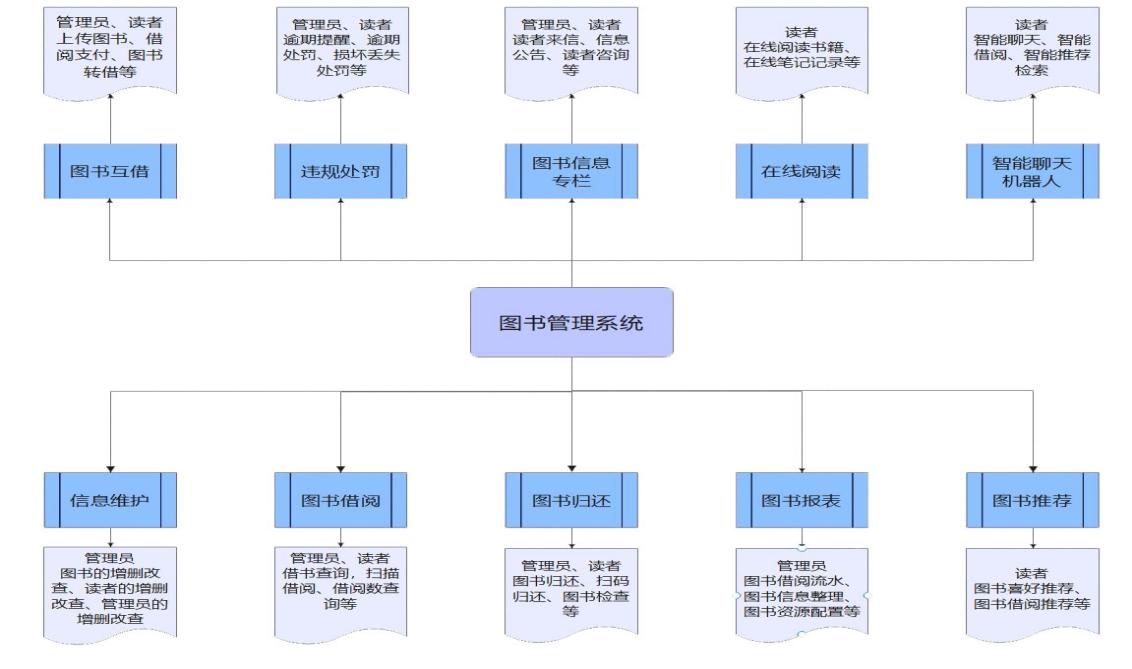


图2-2功能结构图

**模块化设计：**

I DO 智能图书管理分为两大模块：

### 2.2.1读者模块

**账号管理**：

该模块主要是用户对于自己个人信息的管理、用户可以通过邮箱或者手机号进行注册、收到验证码信息之后，验证成功方可注册成功，登录I DO智能图书管理系统后，用户可以对自己的密码进行修改、同样也需要进行邮箱或者手机号的验证，验证成功，方可修改，同时用户还可以对自己的昵称进行修改，并且可以对账户中的余额进行充值等多项功能

**图书借阅：**

该模块主要是用户对图书馆内的图书进行借阅，用户点击书籍可以查看书籍的详细信息，包含书籍的作者、出版社、描述等信息的解释，用户进行借阅时，会将借阅信息传给管理员，管理员核验用户资格后，发送借阅二维码给用户，用户凭借此二维码进行书籍的借阅、如果用户借阅数量超出限度，则会收到信息告知借阅数量超额。

**图书归还：**

该模块主要是用户对于自己已经借阅的图书进行归还，用户在自己的系统当中查看已借阅书籍、即可观察到自己借阅的图书，用户点击归还图书会自动生成二维码，用户将书籍二维码通过管理员扫码成功后，核验书籍后归还成功。

**图书推荐展示模块：**

系统会通过机器学习的算法将用户喜好的图书进行推荐、于此同时系统也会通过滑动窗口为用户展示今日新书以及目前畅销热门书籍供应用户进行选择借阅。

**图书互借模块：**

该模块用户可以将自己的书籍进行上传到图书馆当中，通过图书馆将自己的书籍外借给他人，若他人借阅用户上传的图书，用户的账户中将收取到一定的费用，用户需要借阅他人上传书籍时，同样也现需要对于图书借阅功能的使用，最后进行扣款，扣款成功后，图书互借成功。

**违规处罚模块：**

该模块系统会筛选出书籍借阅逾期的读者和接近截止其的读者，，并将逾期信息发送给读者，读者在自己的信息栏当中可以接收到书籍借阅逾期的提示，提示用户需要进行图书归还，用户对于自己借阅逾期的图书需要向图书馆支付每日0.1元的处罚赔款、并且管理员需要核验图书是否完整，完整则归还成功，不完整则需要用户进行照价赔偿。

**在线阅读模块：**

该模允许用户对目前系统所有录入的图书进行在线阅读、通过初期的阅读，用户可以判断自己对该书籍是否喜爱，以此来进行借阅，进入图书详情界面，点击在线阅读即可进行在线阅读。

**人工智能聊天推荐模块：**

该模块系统使用当今流行的OPENAL——ChatGpt，用户可以针对本图书馆的信息以及个人借阅喜好、图书推荐等多方面进行功能进行询问，避免了新的用户前往图书馆时，时时需要前往管理员处进行询问信息的繁琐过程，增加了本图书管理系统的智能性。

### 2.2.2管理员模块

**用户信息管理模块：**

该模块管理员可以对目前所有注册的用户信息进行统一的管理，可以帮助用户进行修改账户、密码、手机号以及邮箱等多方面的功能，还可以对用户的在线情况和用户的账户进行管理，同时管理员拥有权限对用户进行注销操作，管理员同时可以对自己的信息进行录入、修改、查询等一系列操作。

**图书信息维护模块：**

该模块管理员可以对整个图书馆的图书信息进行维护管理，可以查看目前图书馆当中录入的所有图书，以及图书所在书架位置，方便管理员进行管理，同时系统会实时进行报表的处理，按月对图书借阅信息进行统计和处理分析，供图书管理员查询，并依据此来进行资源配置，同时图书管理员可以对图书进行录入、修改、删除等一系列图书相关操作。

**借还管理模块：**

该模块管理员可以对用户进行借书和还书的请求进行处理，通过判断用户是否可以进行借阅图书，可以，则发送二维码，不可以，则发送失败原因，针对用户归还图书，则是进行图书逾期和完整性判断，通过二维码扫描，成功之后归还成功。

**逾期处理模块：**

该模块管理员可以对逾期以及接近逾期的图书进行逾期警告，针对以及逾期的用户，通过判断其逾期时长对其进行信誉评价以及相关罚款处理、对于信誉评价较低的用户将降低其可借图书的数量

**图书互借模块：**

该模块管理员可以将用户对于申请上传的图书进行审核，审核成功之后将其录入图书馆当中，当其他用户需要借阅此图书时，向管理员发起申请，管理员接收请求，同意借书后，将收款码发送给该用户进行支付。

## 2.3功能分配

IDO智能图书管理系统在传统图书馆的基础上，创新了信息专栏、在线阅读、读者咨询、智能聊天机器人、图书推荐等功能，主体上还是划分为管理员端和用户端，用户端又可以划分为图书借阅模块、聊天咨询模块、信息安全模块、便捷服务模块。而管理员端又可以划分为图书信息管理模块、咨询回复模块、逾期处罚模块、信息安全模块。

系统功能模块划分、具体功能、功能描述、用户群以及优先级如表2-1、2-2所示：

**用户端：**

表2-1 用户功能模块描述

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **用户群** | **优先级** |
| **信息安全模块** | **注册** | **用户自主选择昵称和密码惊进行注册，同时需要选择进行手机号或者邮箱验证** | **全体** | **高** |
| **登录** | **登录IDO图书管理的系统** | **全体** | **高** |
| **信息修改** | **用户修改自己的个人信息** | **全体** | **高** |
| **账户充值** | **用户对自己的账户余额进行充值** | **全体** | **高** |
| **图书借阅模块** | **图书借阅** | **用户通过二维码进行图书借阅** | **全体** | **高** |
| **图书归还模块** | **用户通过而二维码扫描进行图书归还** | **全体** | **高** |
| **逾期提示处罚** | **用户收到逾期提示信息，及时对图书进行归还，逾期进行一定金额罚款** | **全体** | **高** |
| **图书互借** | **用户上传图书给其他用户进行借阅并且收取一定的费用** | **全体** | **高** |
| **聊天咨询** | **读者咨询** | **用户和管理员进行实时的聊天通信，进行信息询问** | **全体** | **高** |
| **信息公告栏** | **用户通过公告栏对图书馆发布的信息进行了解和评价以提出个人诉求** | **全体** | **高** |
| **便捷服务** | **智能聊天机器人** | **用户可以与智能机器人进行聊天，了解推荐图书、今日图书以及其他图书馆相关信息** | **全体** | **中** |
| **图书推荐** | **系统通过机器学习，通过用户平日的借阅量搜索量等来进行图书推荐** | **全体** | **高** |

**管理员端：**

表2-2 管理员功能模块描述

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **用户群** | **优先级** |
| **信息安全模块** | **注册** | **管理员自主选择昵称和密码惊进行注册，同时需要选择进行手机号或者邮箱验证** | **全体** | **高** |
| **登录** | **登录IDO图书管理的系统** | **全体** | **高** |
| **信息修改** | **管理员修改自己的个人信息** | **全体** | **高** |
| **信息管理** | **管理员对全部图书馆的用户信息进行统一的监管和修改等操作** | **全体** | **高** |
| **图书信息管理模块** | **图书借阅** | **管理员核验用户资格，向用户发送二维码进行图书借阅** | **全体** | **高** |
| **图书归还** | **管理员查看书籍是否完好，通过对二维码扫描图书归还** | **全体** | **高** |
| **图书信息录入、删除** | **管理员对于图书馆的书籍信息进行增删改查** | **全体** | **高** |
| **图书互借** | **管理员审核用户的上传图书，并对借用此图书的用户提供，图书提供者的收款码** | **全体** | **高** |
| **图书报表** | **管理员对于借阅记录进行报表查看，统计分析便于后期对书籍资源的管理** | **全体** | **高** |
| **咨询回复** | **读者回复** | **用户和管理员进行实时的聊天通信，进行信息询问** | **全体** | **高** |
| **信息公告栏** | **管理员向整个图书馆的用户发布关于图书馆的最新信息，同时得到用户新的诉求** | **全体** | **高** |
| **逾期处罚** | **逾期信息提示** | **管理员向接近逾期和已逾期的用户发送提示还书的短信** | **全体** | **高** |
| **处罚** | **管理员针对逾期的用户进行每日0.1元的处罚，对于丢失或者损坏图书的按照原价赔偿** | **全体** | **高** |

## 2.4接口设计

### 2.4.1外部接口

百度API接口：进行地图定位

ChatGpt接口：智能聊天机器人

支付宝接口：逾期罚款支付、互借支付

### 2.4.2内部接口

系统在设计时充分解耦，为各模块提供了函数和类级别的接口。

深度学习模块对外暴露了若干函数，通过传递一系列的参数提供深度学习服务的快速调用；

数据库模块对外暴露ORM对象，通过实例化对象和公用函数库进行数据库操作；

程序主要内部接口：

**1）二维码借还书需求**

输入：二维码图片

数据格式：{二进制数组}

数量：1

输出：二进制数组转换城jpg格式的二维码图片

**2）读者通信需求**

输入：读者信息

数据格式：{字符串}

数量：1

输出：内网穿透通过服务器发送给另外一端

**3）逾期短信提示需求**

输入：逾期提示

输出：mob发送短信到相应的手机号

**4）图书报表需求**

输入：图书以往的借阅记录

数据格式：{借阅种类、借阅数量}

数量：2

输出：形成柱状图、折线图、饼状图

**5）信息公告需求**

输入：公告信息

数据来源：云数据库中读取

数据格式：{成员id，公告信息}

数量：2

输出：展示公告信息到整个系统

**6）用户详情数据需求**

输入：成员id

数据来源：Redis数据库信息

数据格式：{id，用户名，性别，电话 ，密码}

数量：1

输出：图书馆中读者的信息

**7）书本详情数据需求**

输入：书本id

数据来源：Redis数据库信息

数据格式：{id,ISBN,书名、数量}

数量：1

输出：图书馆中书籍的全部信息

**8）借阅详情数据需求**

输入：用户id

数据来源：Redis数据库信息

数据格式：{用户id，密码}

数量：1

输出：图书馆中借阅信息的展示

### 2.4.3 管理员Web端界面设计

**1）登录界面**

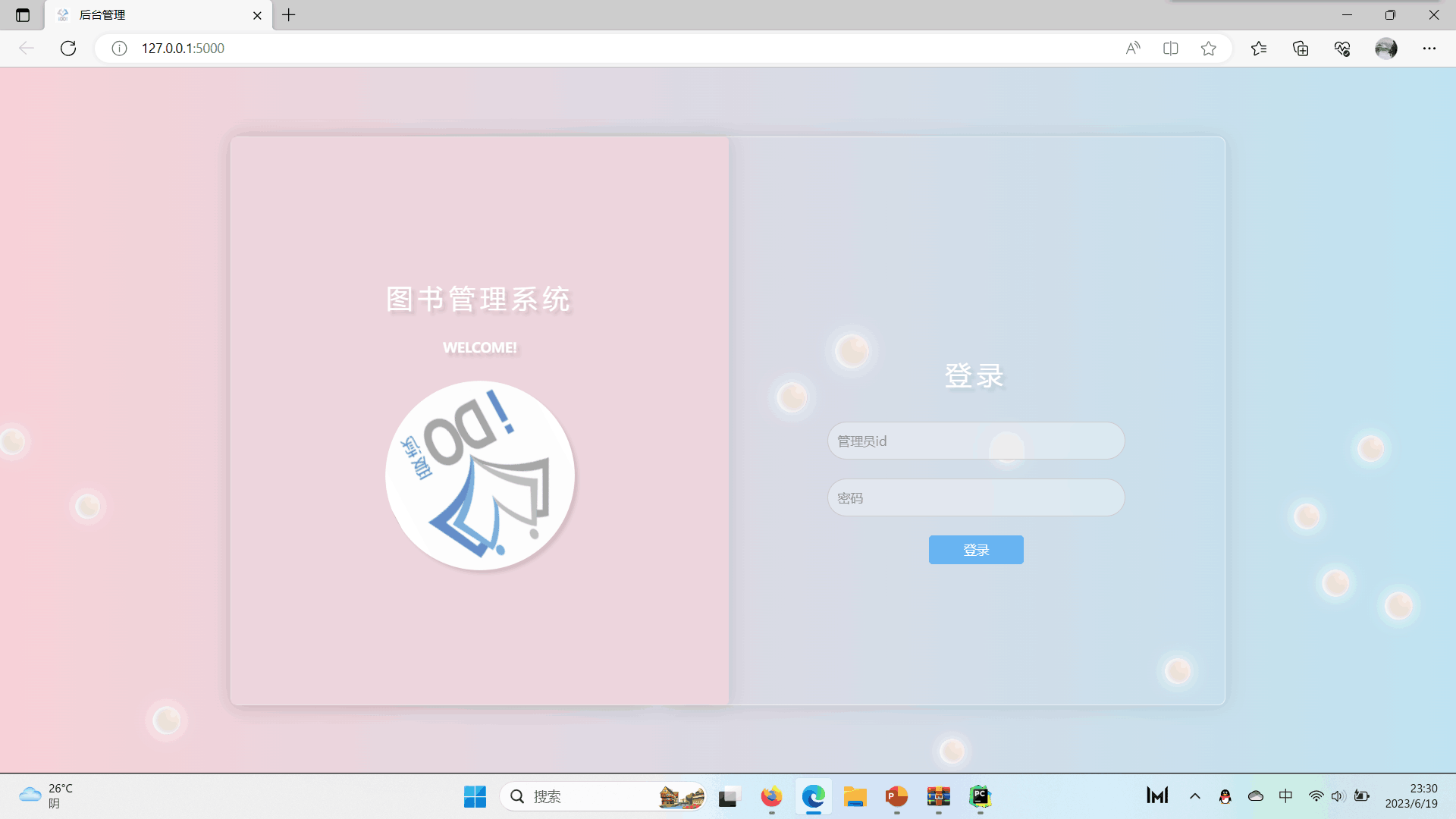


图2.3登录界面

**2）首页**

****

图2.4首页欢迎界面

**3）读者管理信息查看界面**

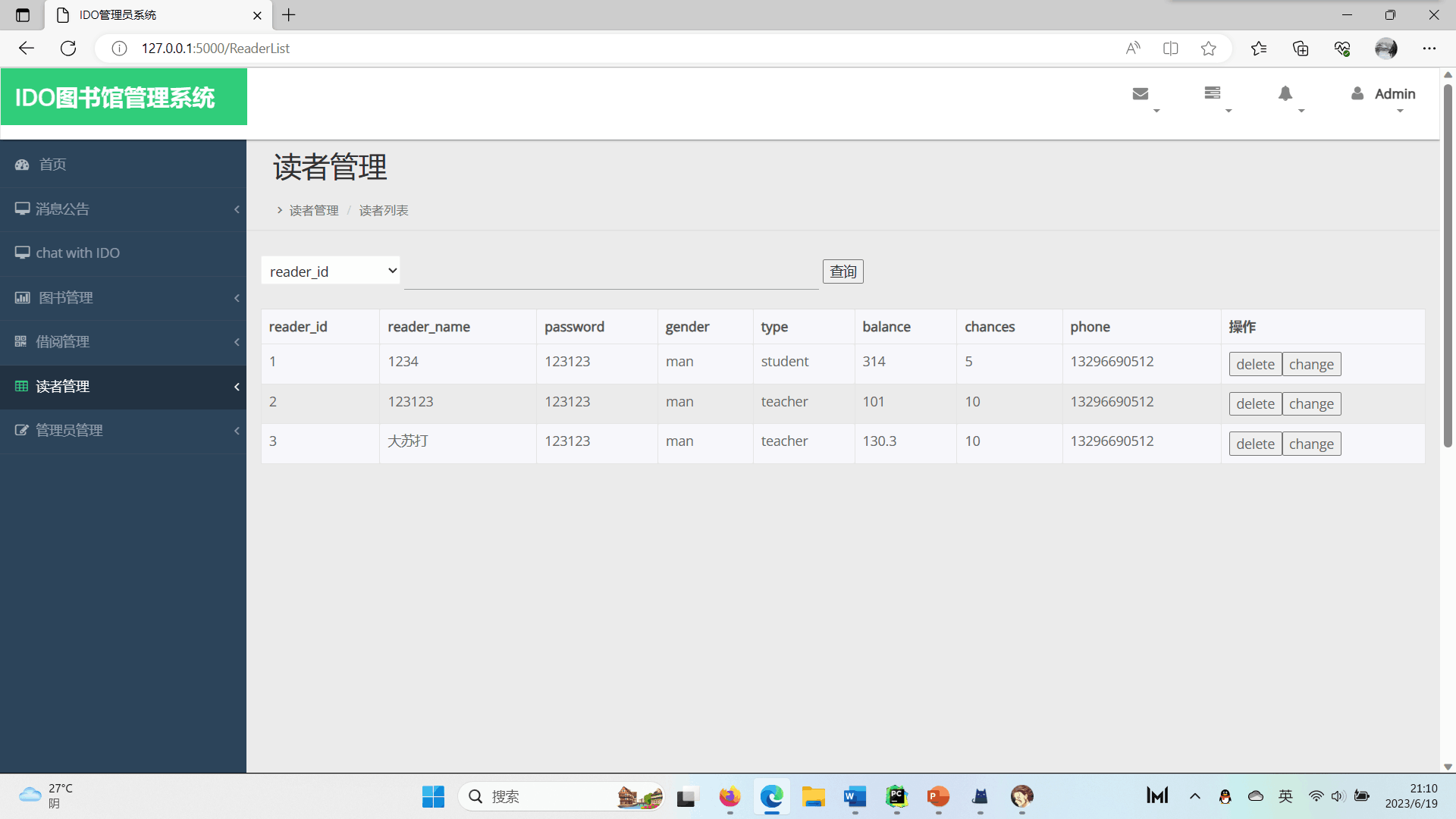
****

图2.5读者管理信息查看界面

**4）读者信息修改界面**

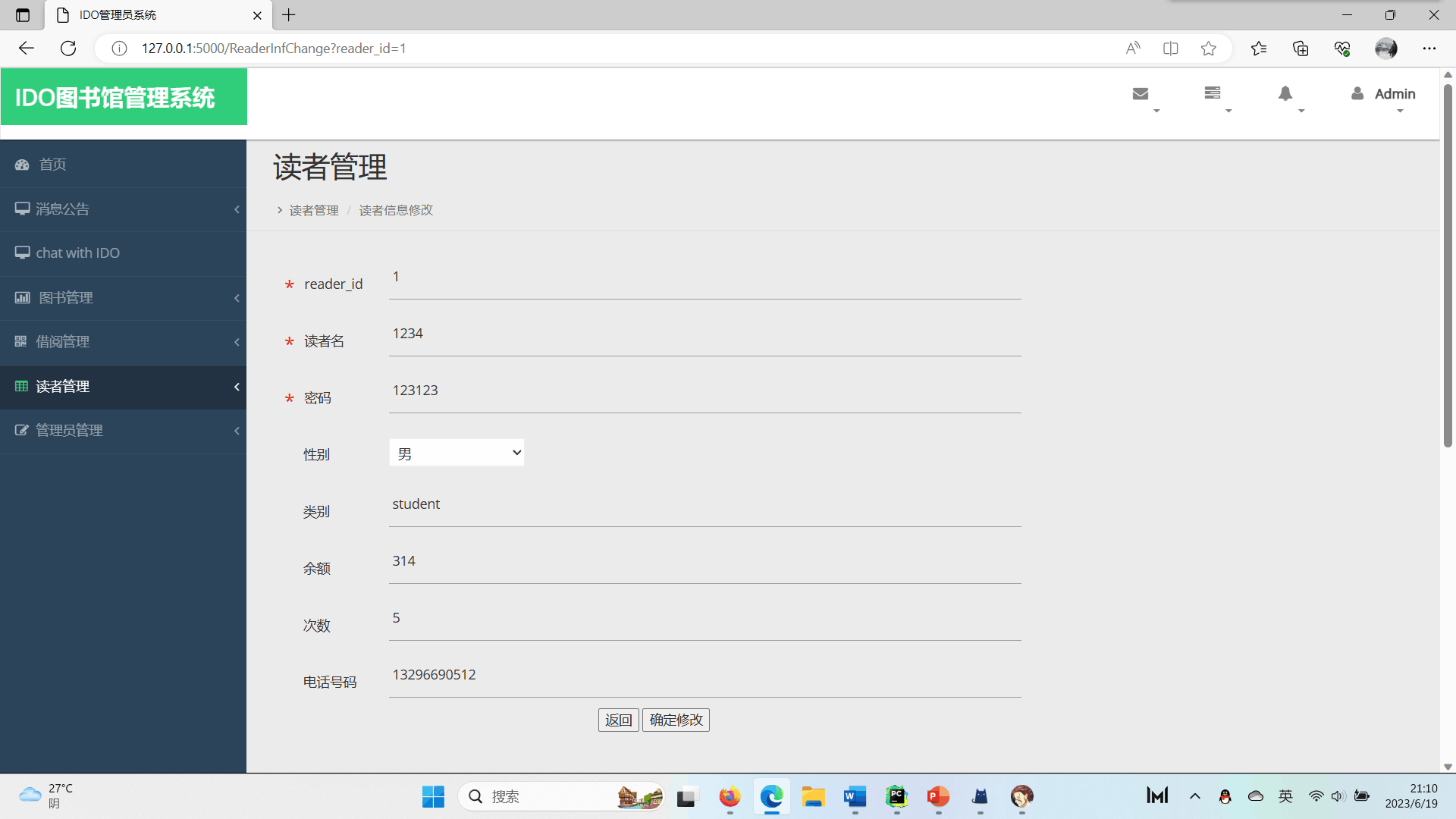
****

图2.6读者信息修改界面

**5）管理员信息查看界面**

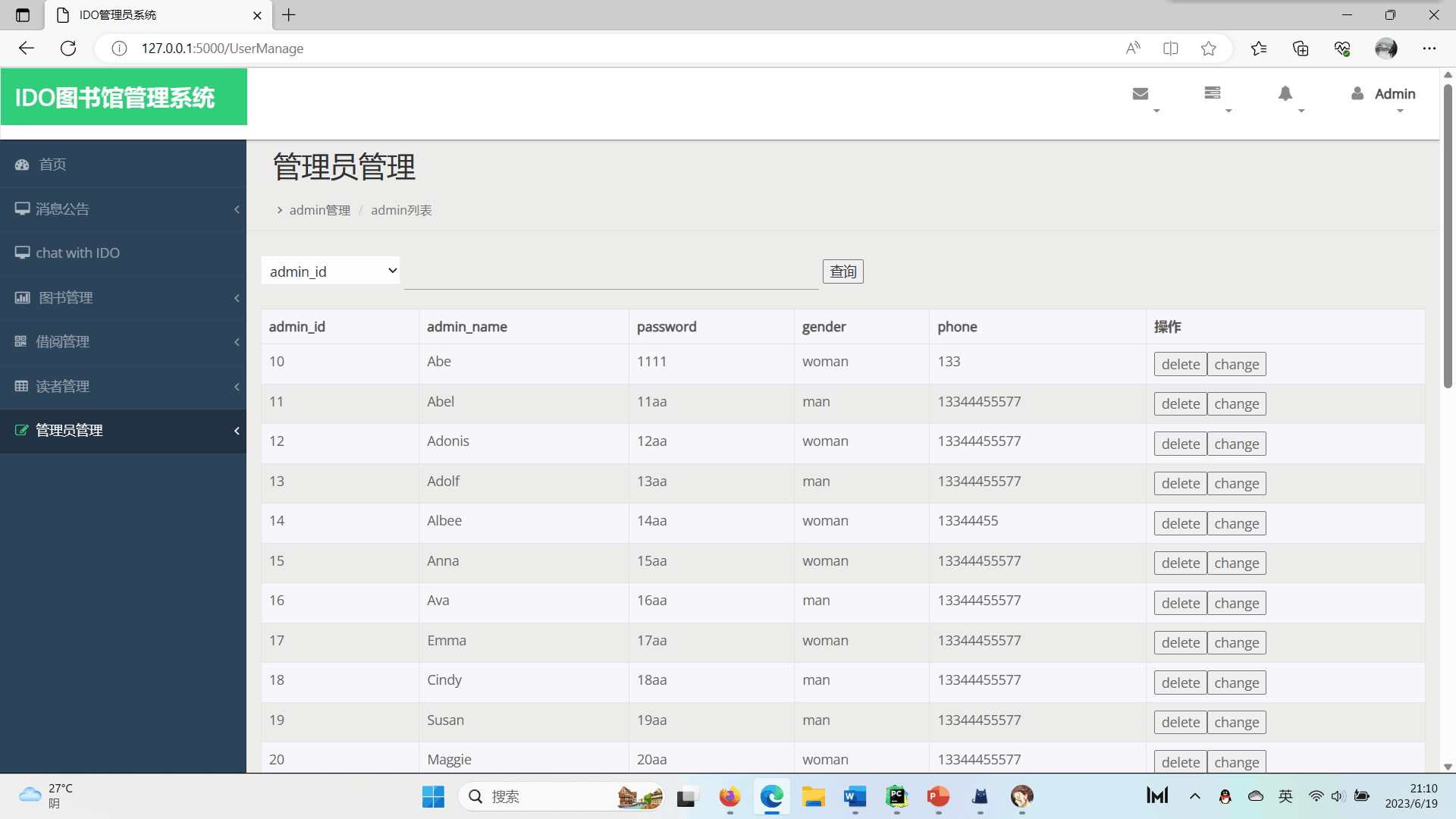
****

图2.7管理员信息查看界面

**6）管理员信息修改界面**

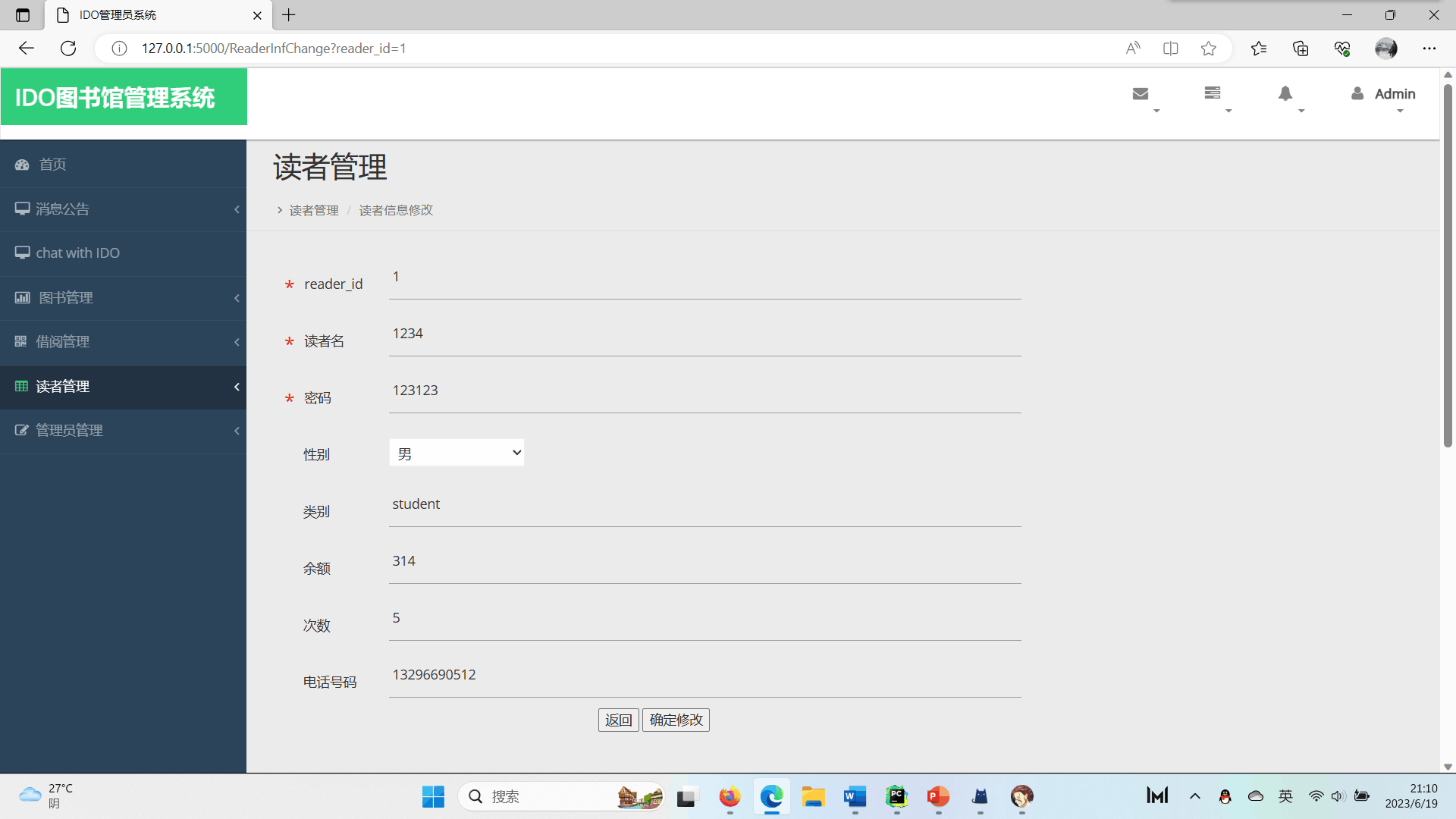
****

图2.8管理信息修改界面

**7）图书信息查看界面**

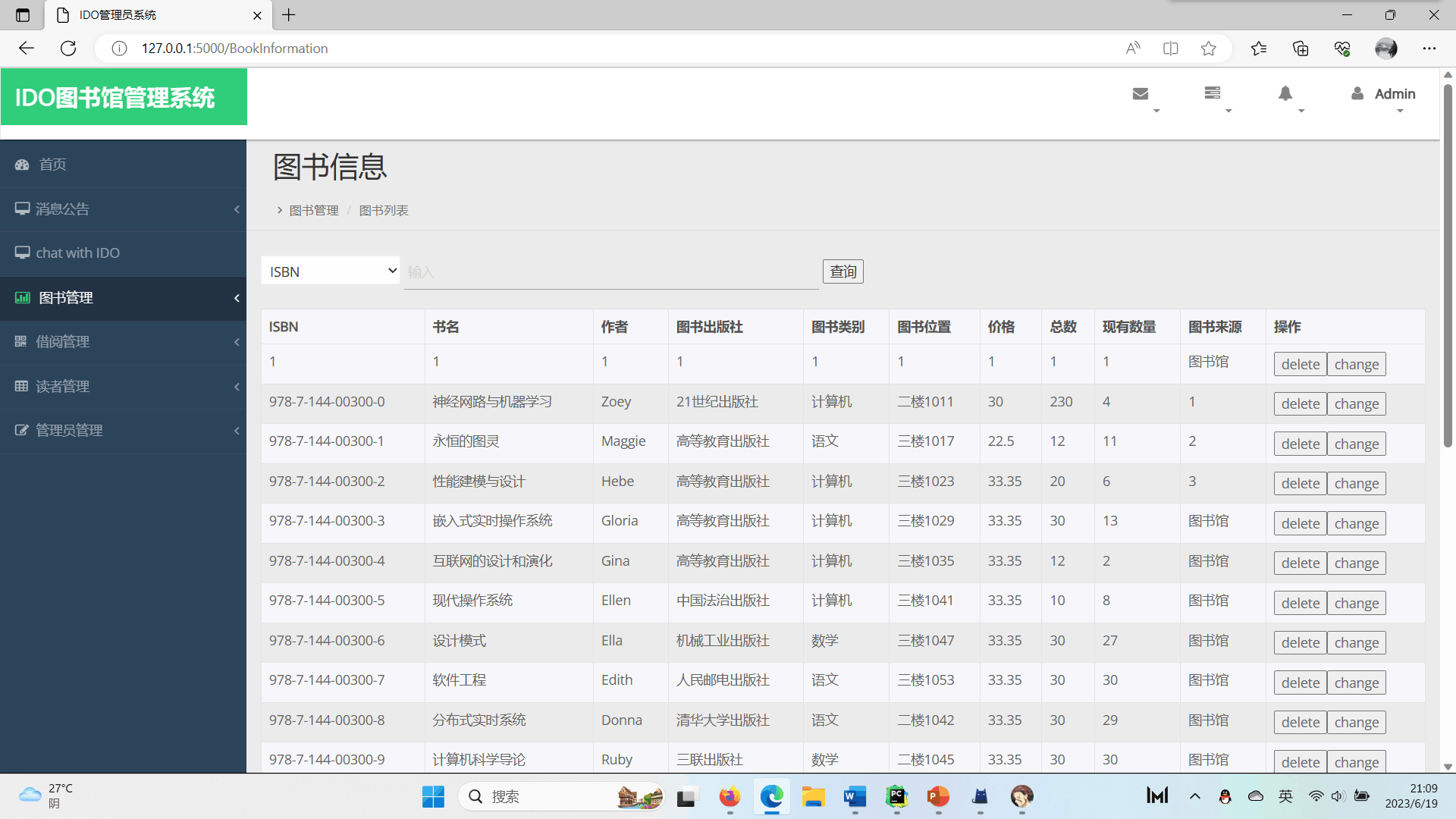
****

图2.9图书信息查看界面

**8）图书信息录入界面**

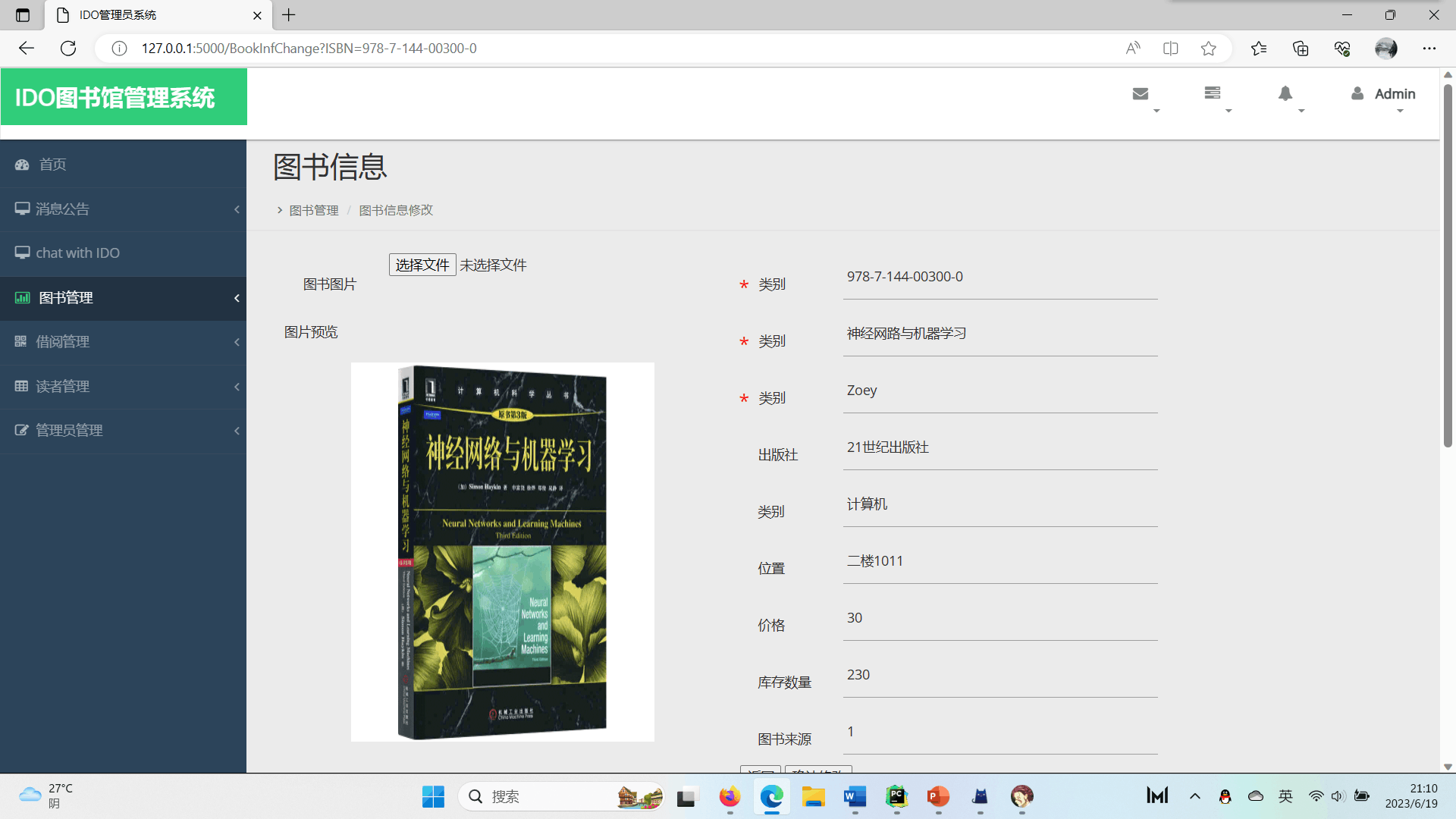
****

图2.10图书信息录入界面

**9）借阅管理界面**

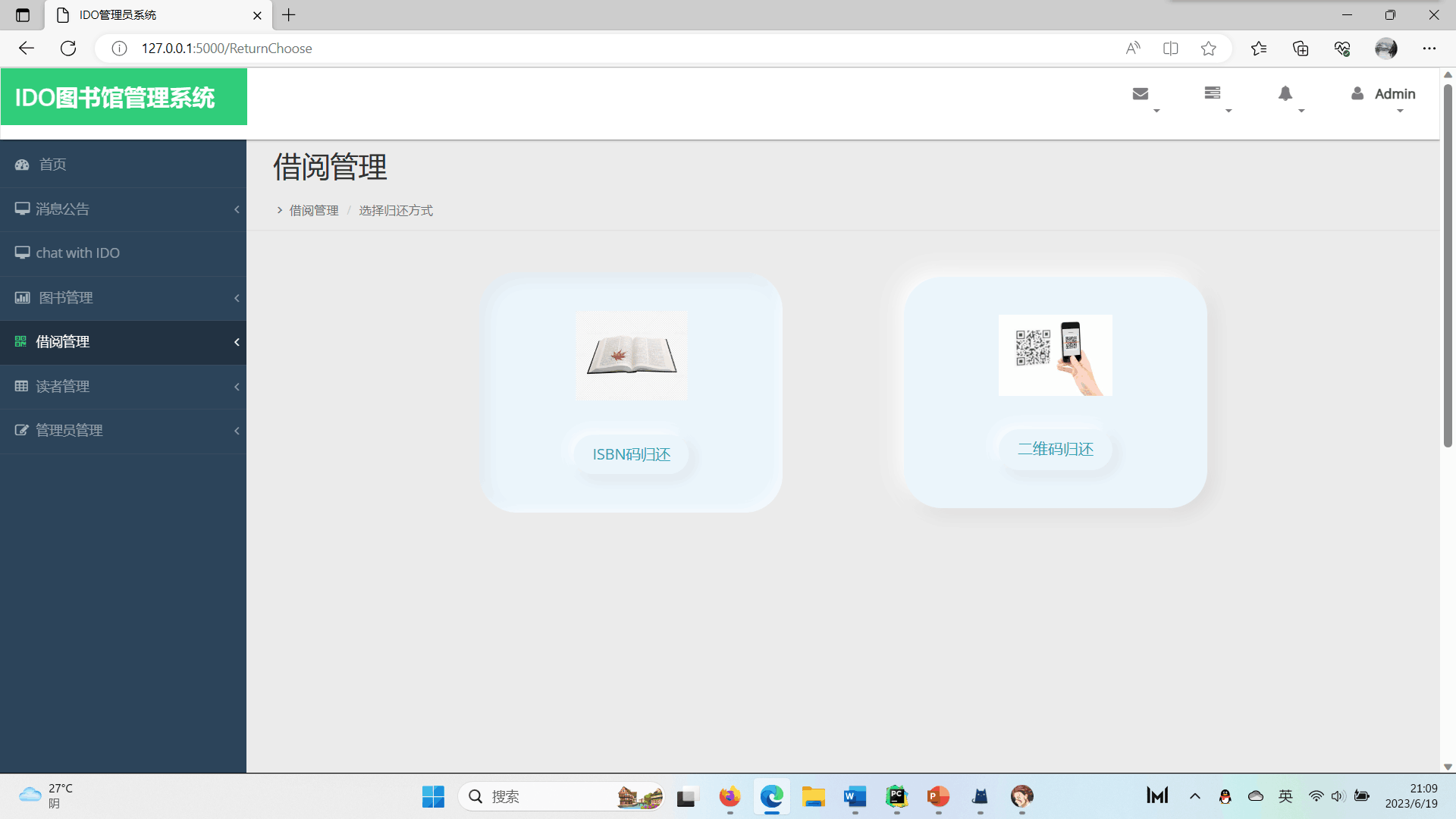
****

图2.11借阅管理界面

**10）借阅信息修改界面**

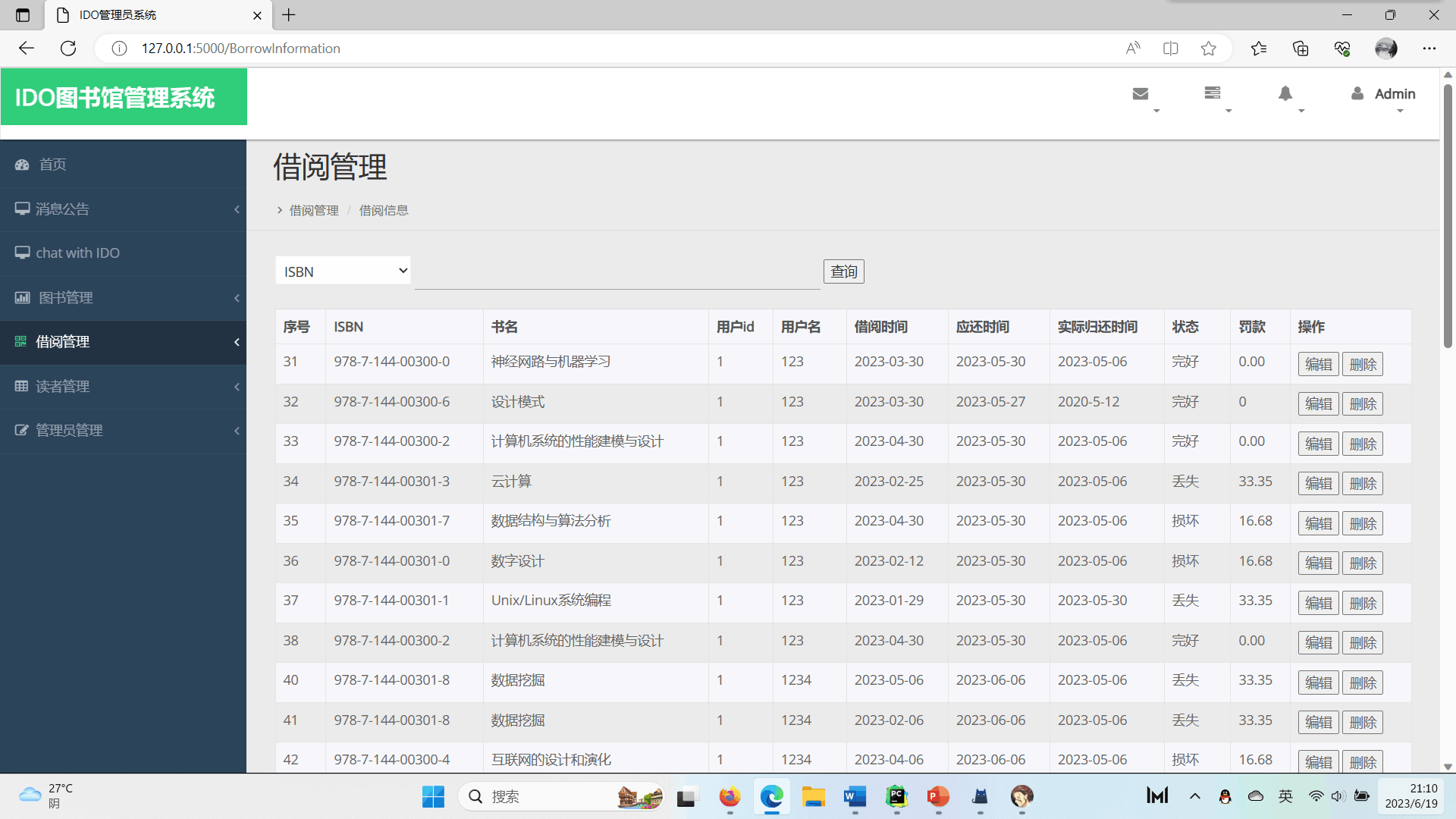
****

图2.12借阅信息修改

**11）智能聊天机器人界面**

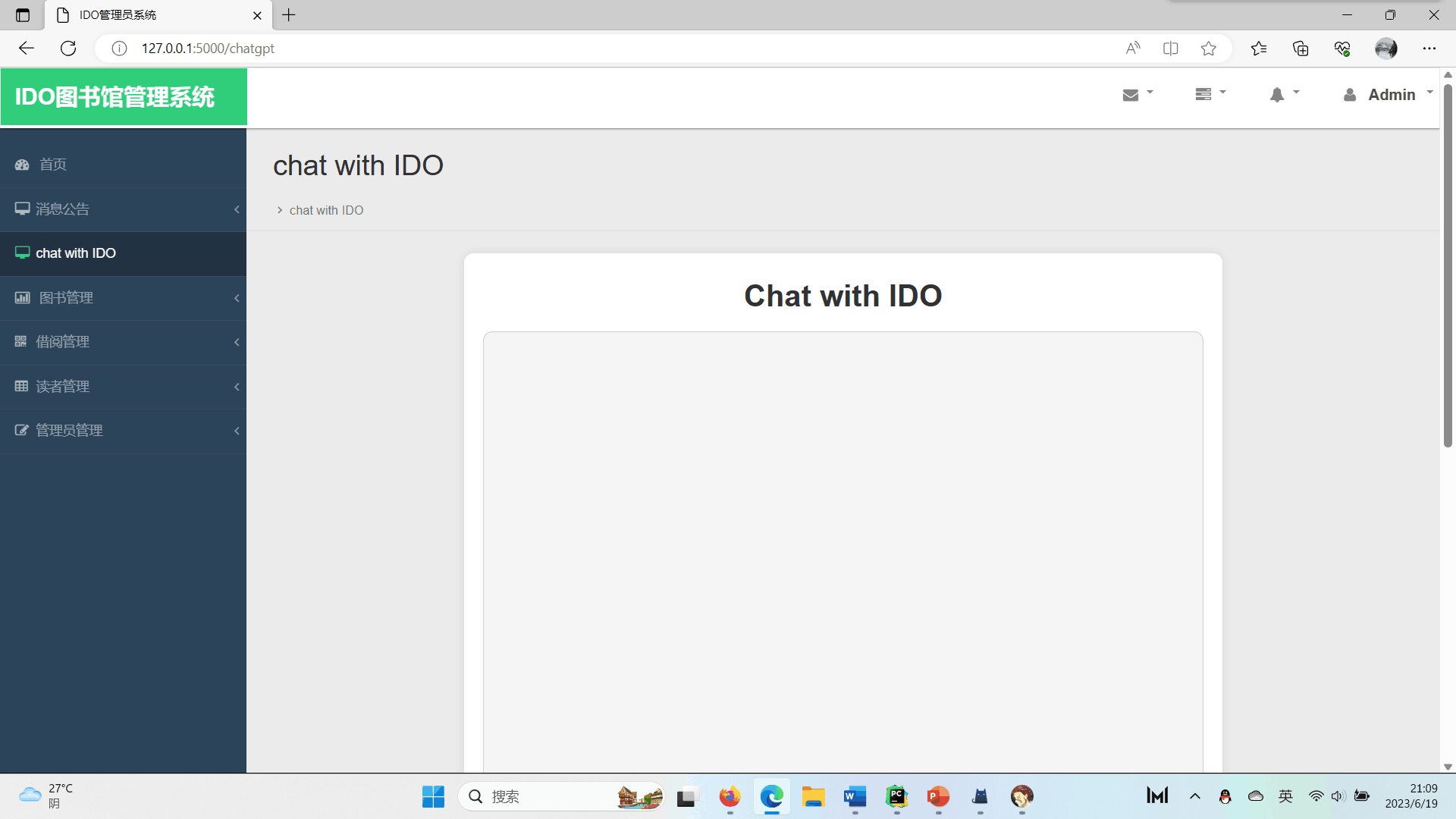
****

图2.13智能聊天机器人界面

**12）读者来信界面**

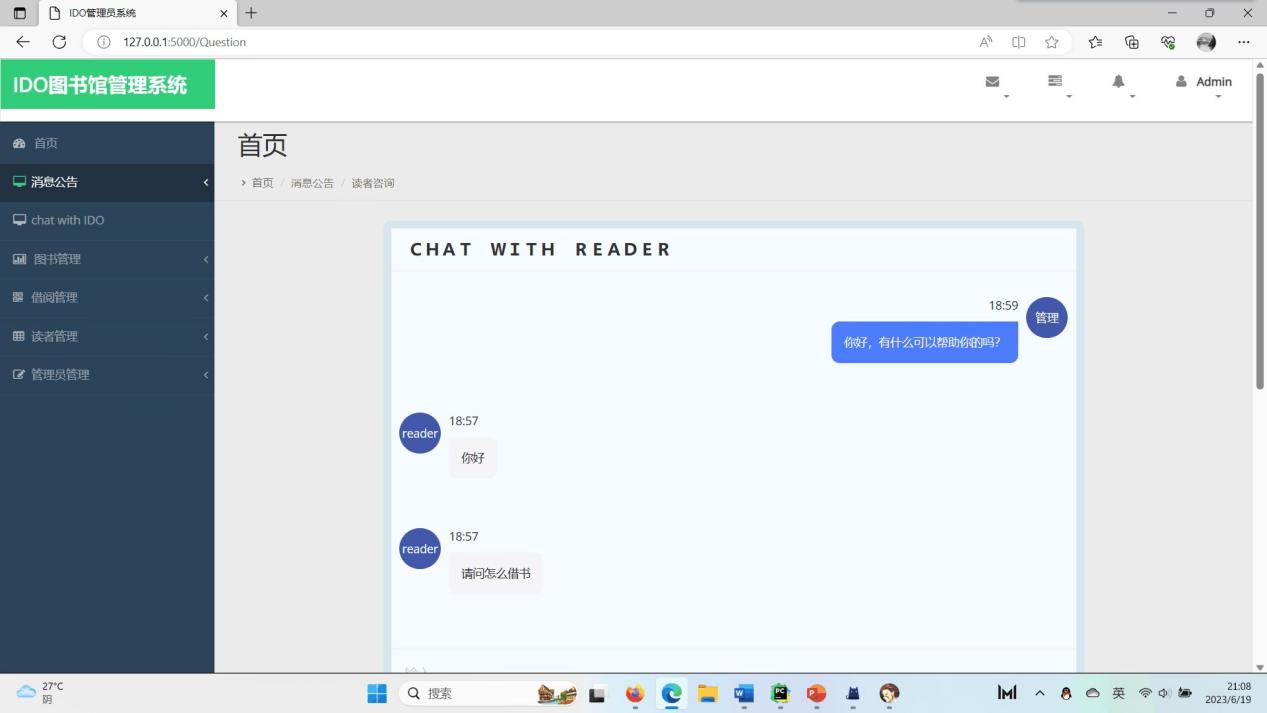
****

图2.14读者来信界面

**13）消息公告发布界面**

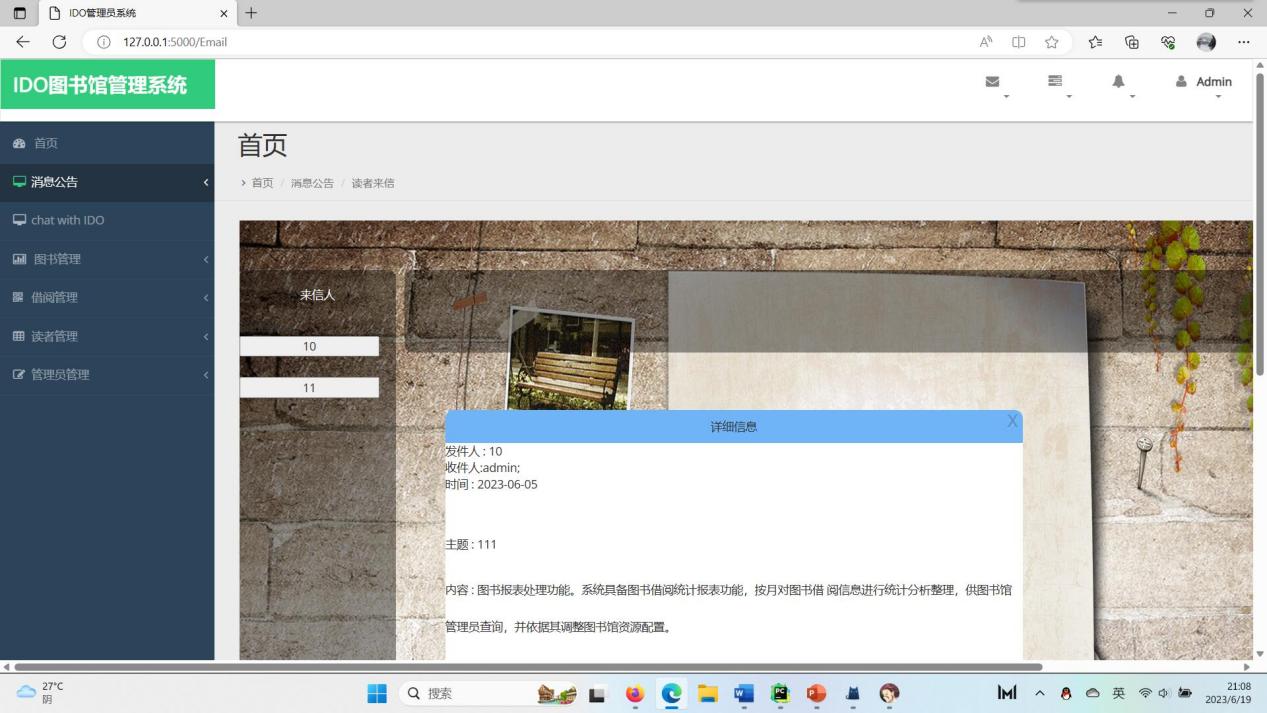
****

图2.15消息公告发布界面

**14）消息公告查看界面**

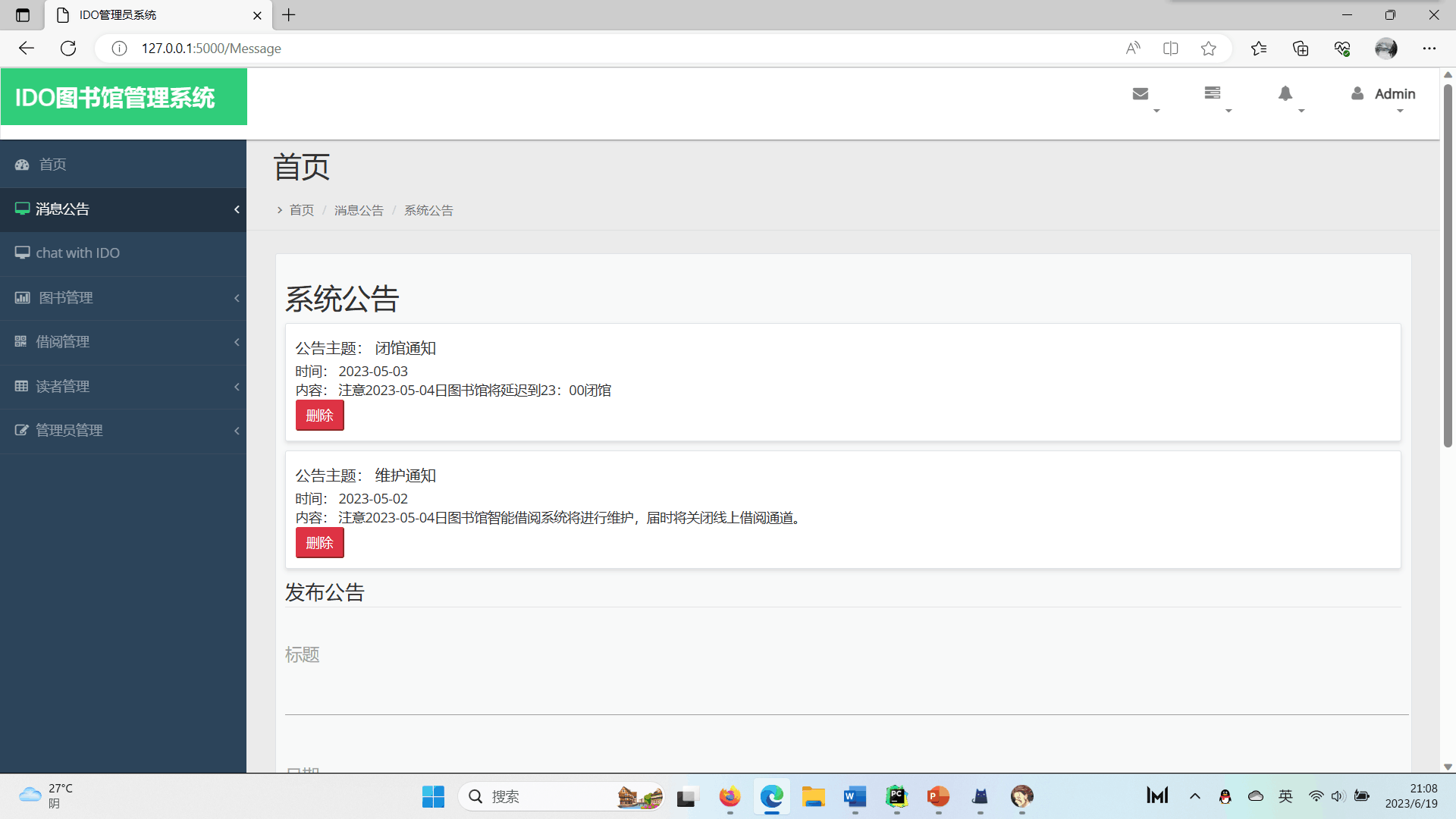
****

图2.16消息公告查看界面

### 2.4.4读者web端界面设计

**1）用户登录界面**

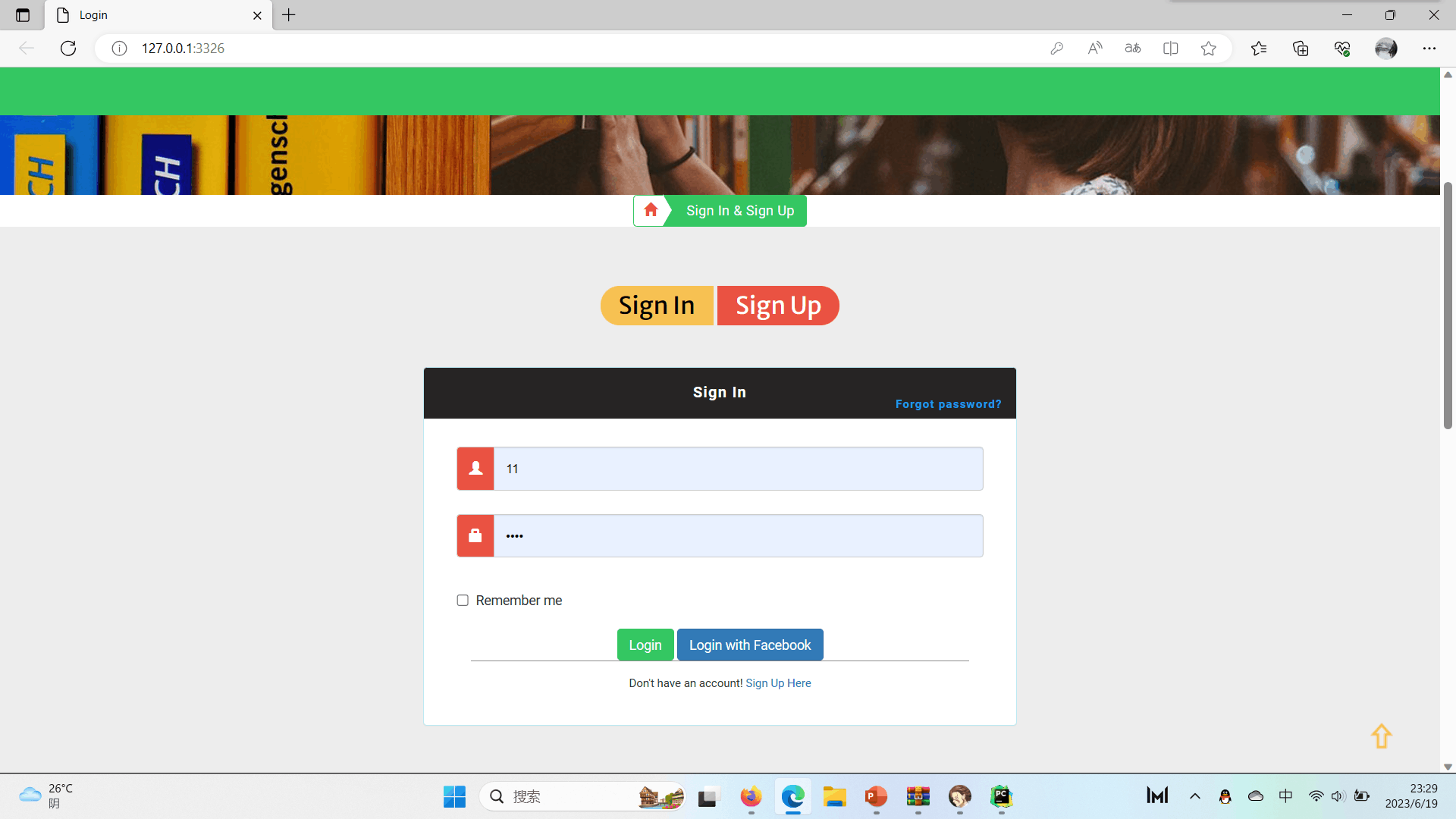


图2.17用户登录界面

**2）主页界面**

****

图2.18**主页界面**

**3）图书推荐界面**

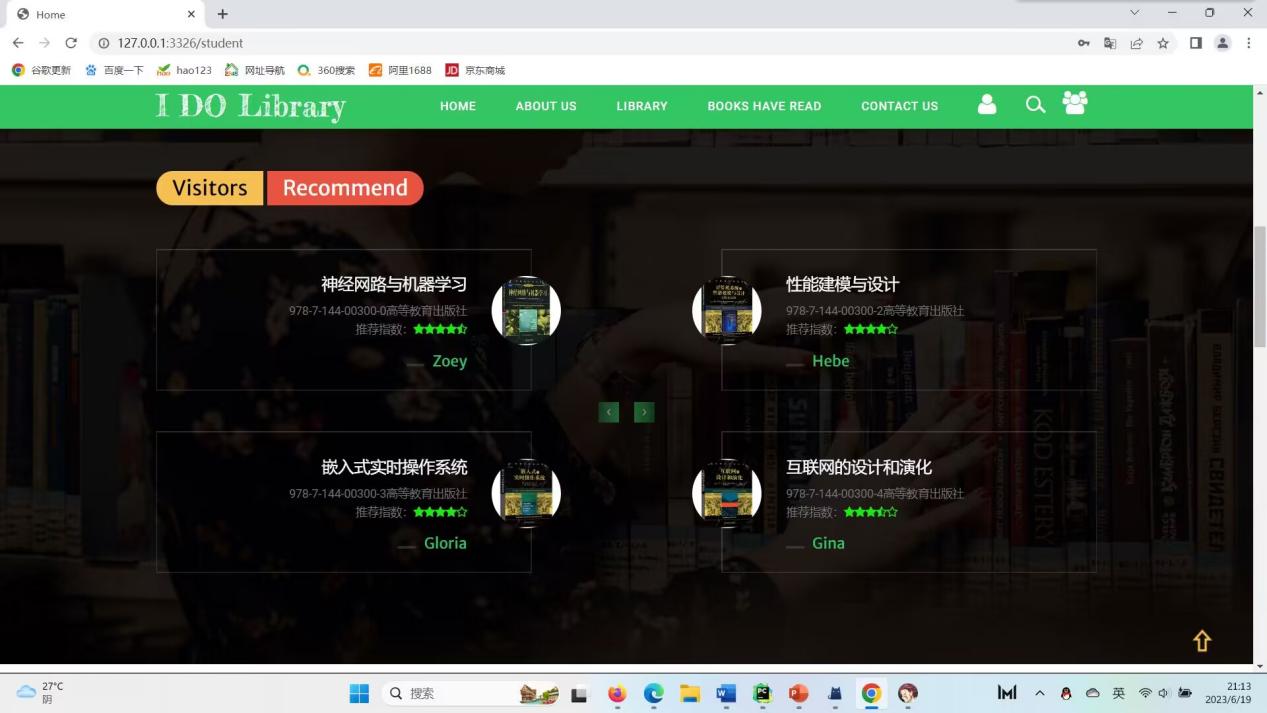


图2.19**图书推荐界面**

**4）图书馆全景布局界面**

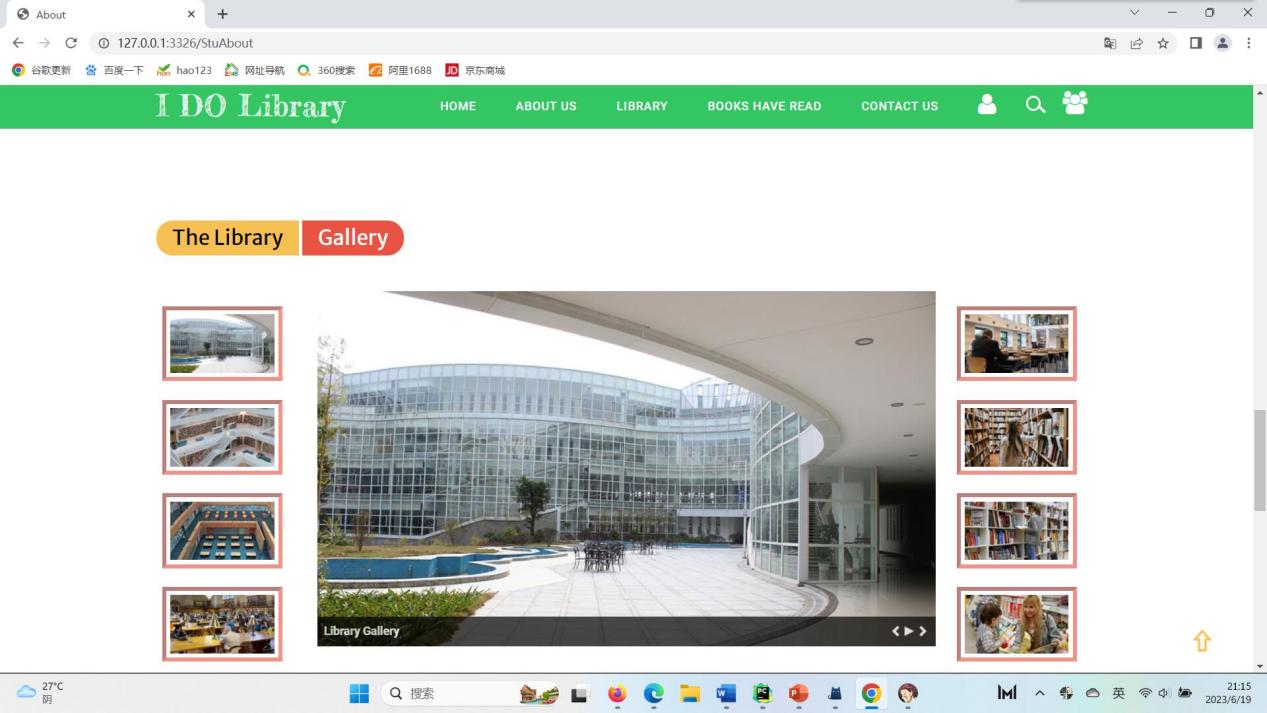


图2.20图书馆全景布局界面

**5）个人信息展示界面**

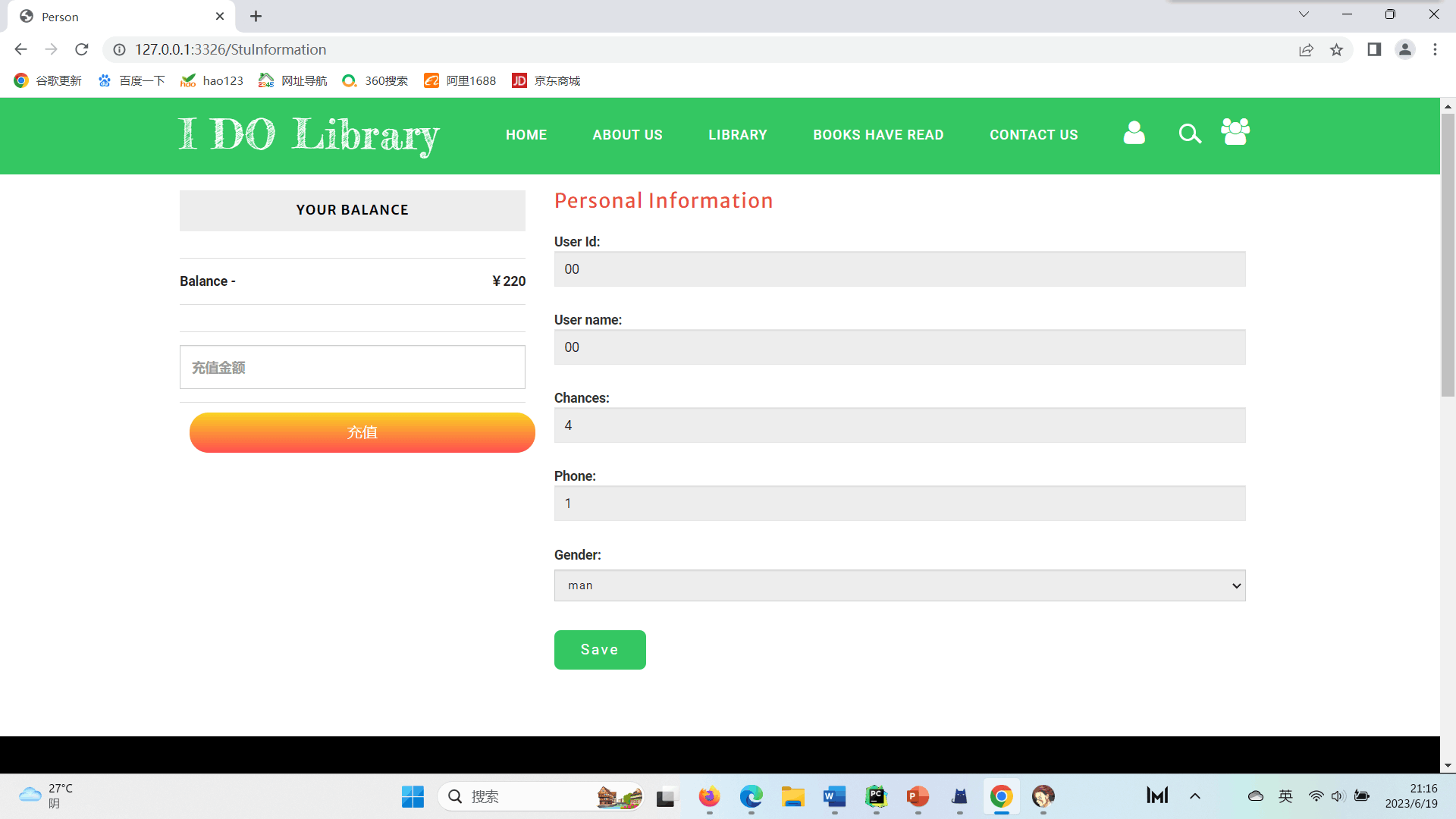


图2.21个人信息展示界面

**6）图书展示界面**

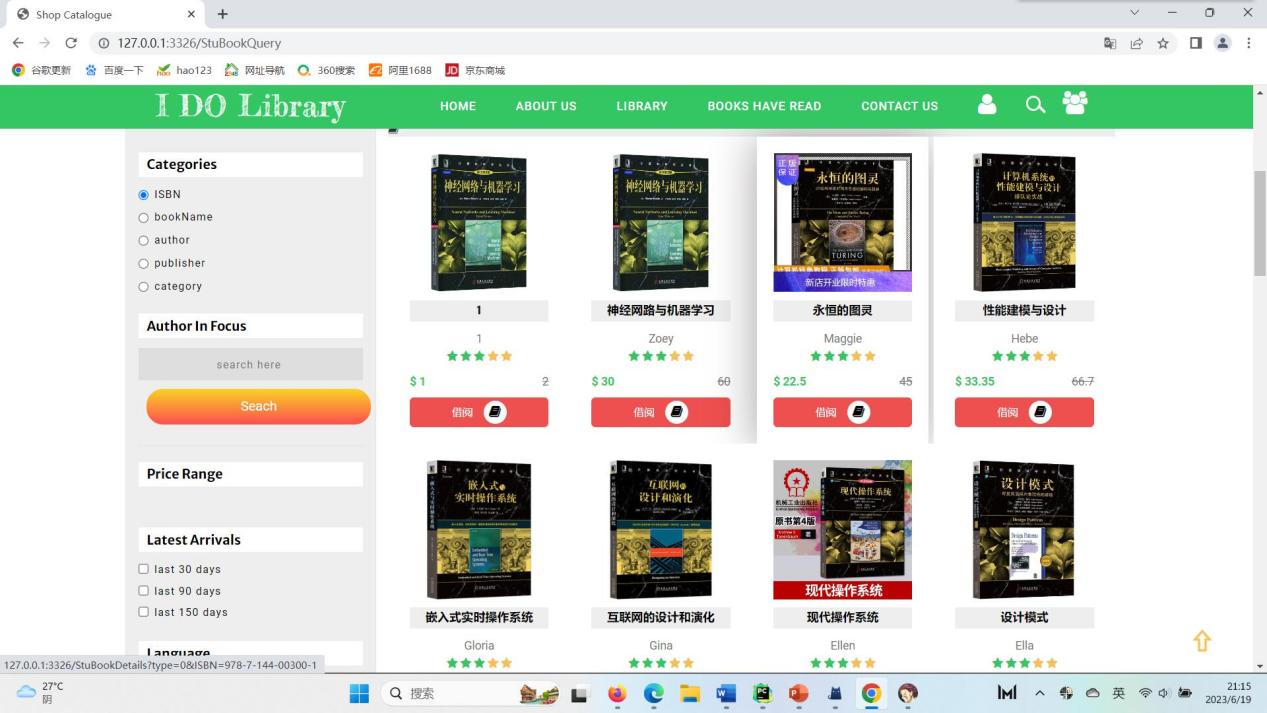


图2.22图书展示界面

**7）图书详情展示界面**

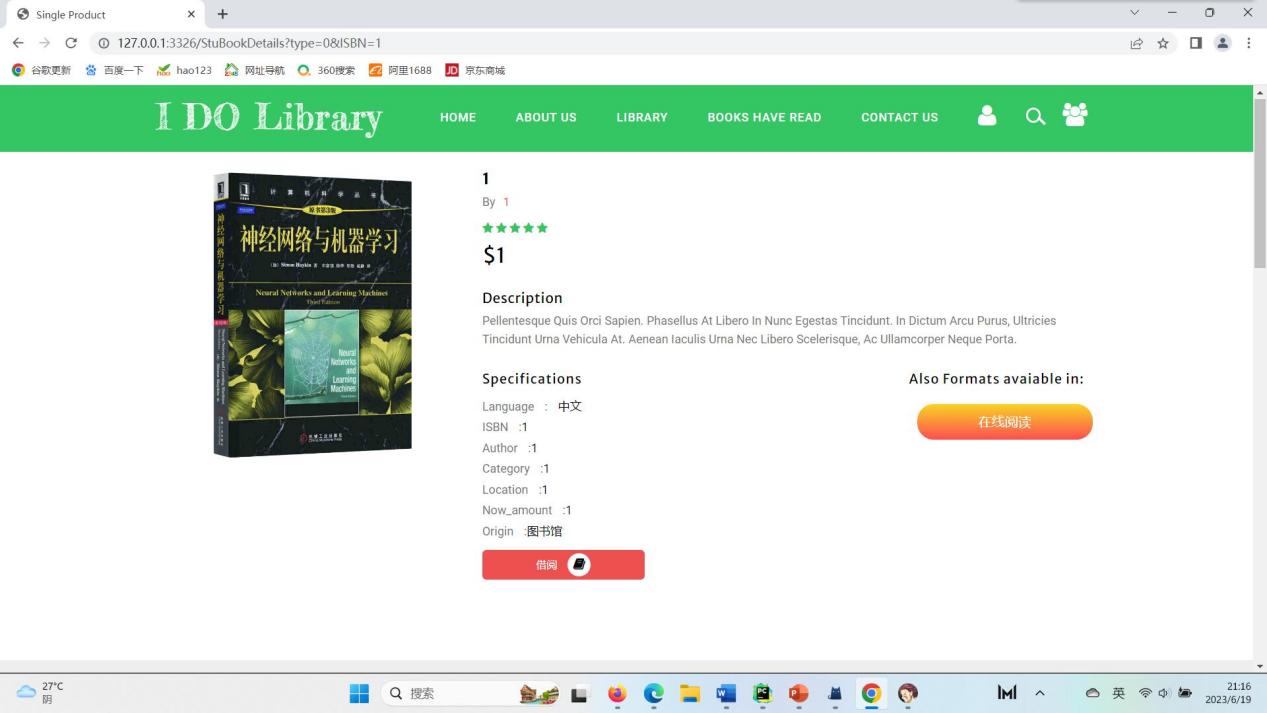
****

图2.23图书详情展示界面

**8）已借图书界面**

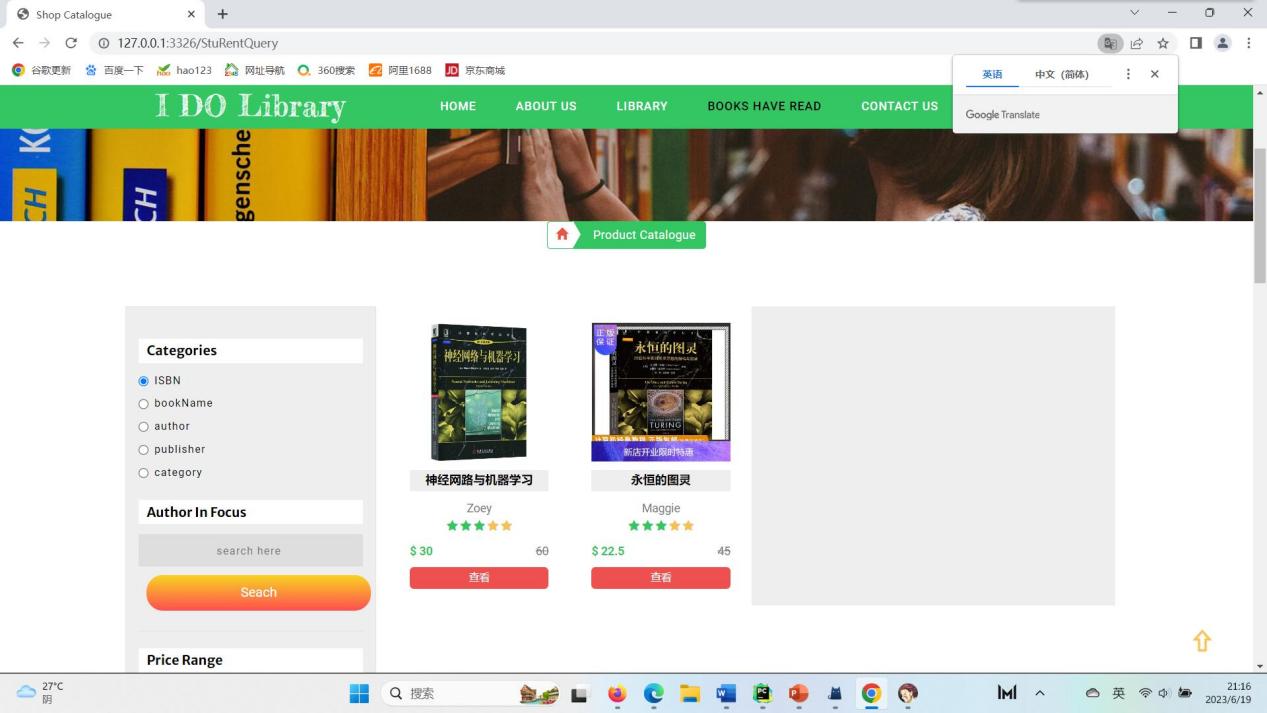


图2.24 已借图书界面

**9）读者来信咨询**

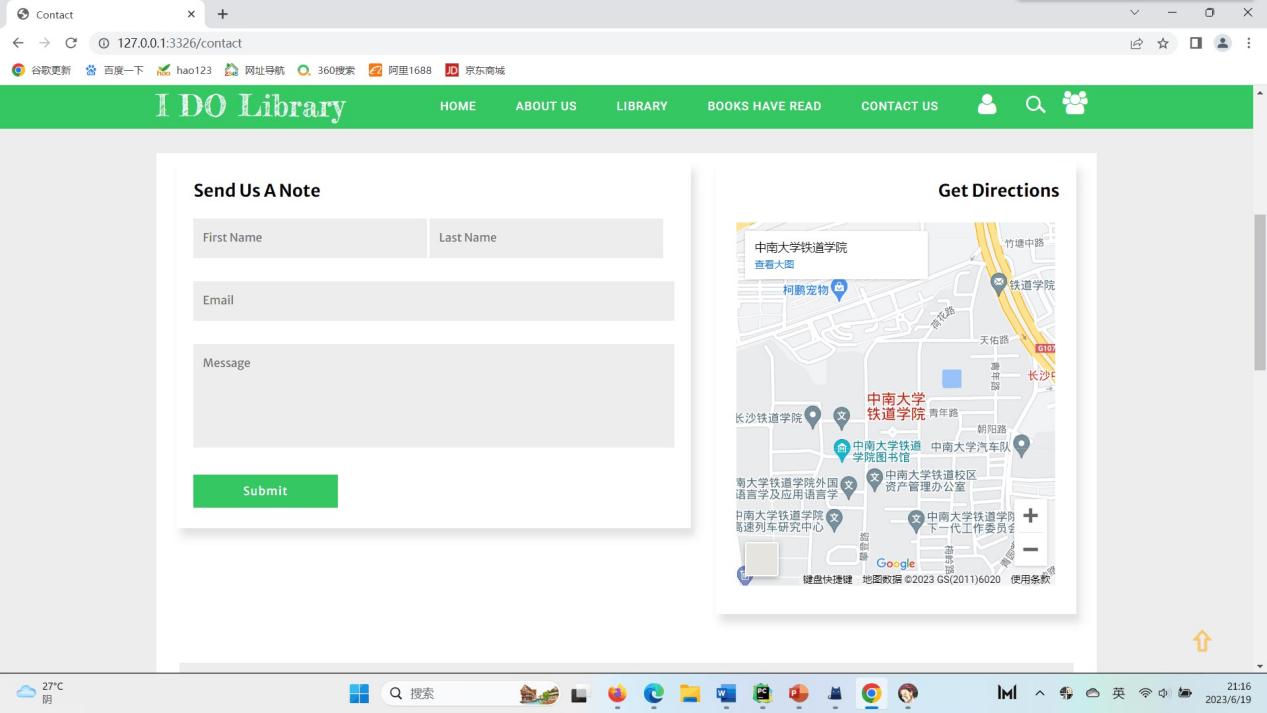


图2.25 读者来信咨询界面

**10）信息公告栏查看界面**

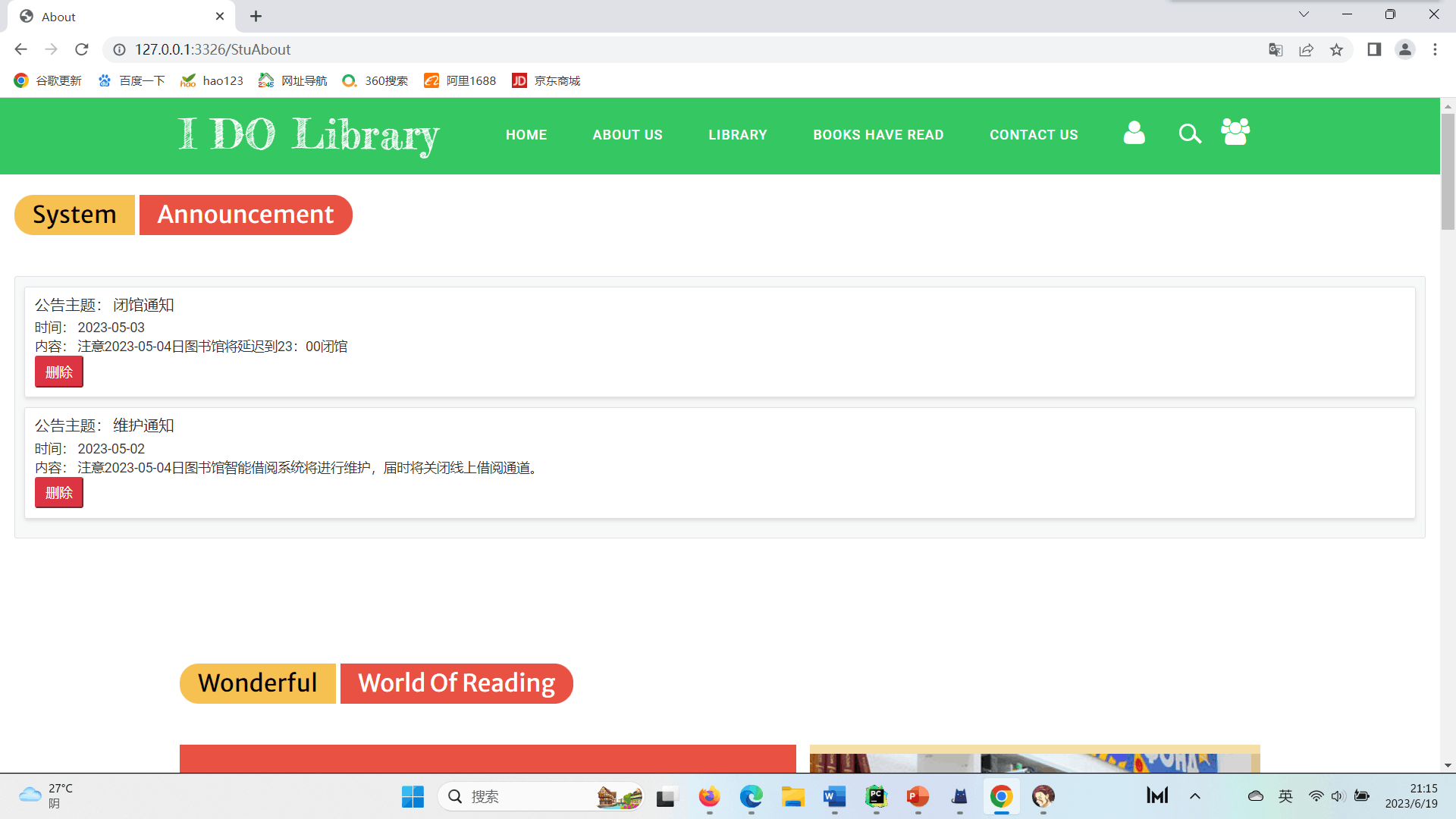


图2.26 信息公告栏查看

**11）读者聊天界面**

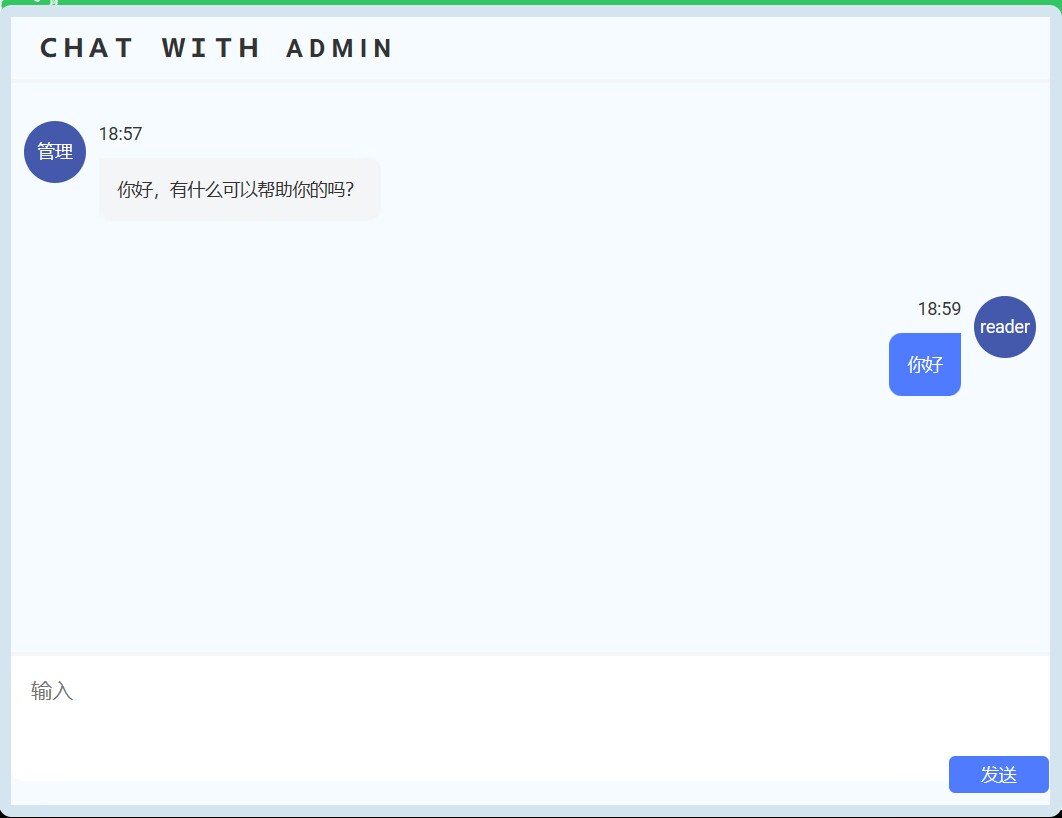


图2.27读者聊天界面

## 2.5数据结构设计

### 2.5.1逻辑结构设计

本项目采用MYSQL作为数据库，用 Redis集群进行存储。使得系统的读取性能快，性能好，更安全。系统数据库关键表的 E-R 如下图所示:

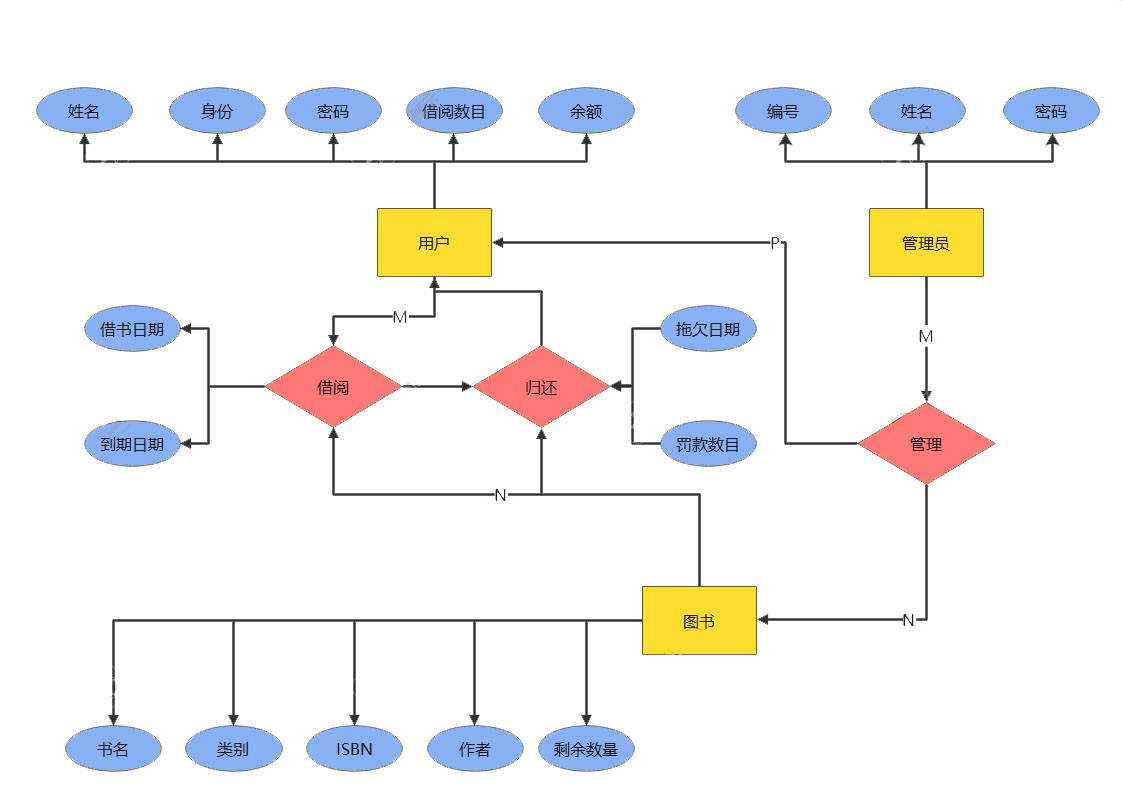
****

图2.28 数据库关键表E-R图

系统数据库关键表的UML图如下图所示:

****

图2.29 数据库关键表UML图

### 2.5.2物理结构设计

系统需要服务于多用户同时进行访问，需要保证其一次可以容纳多个并发的进程，因此使用服务器端是采用了负载均衡的技术。通过该架构使用分布式Redis集群架构。用户在发送一个请求的时候，先接收到的是Linux上的Nginx服务器进行负载均衡，保证并发性，再分发使用到两个服务器当中，服务器接收到读者的登录之后，将用户登录的SessionId放入SessionStorage中进行保存，sessionid即是读者的唯一标识，作为Redis缓存服务器当中的key，而value则是用户的准确信息，登录成功之后SessionStorage会将传输过来的值全部json序列化成json的形式进行存入Redis中，这样当多用户进行的时候使用哈希表进行便利就可以提高命中率，当读取到读者登录信息之后，接着就是用MyBatis去进行Dao层的操作。系统部署如图：

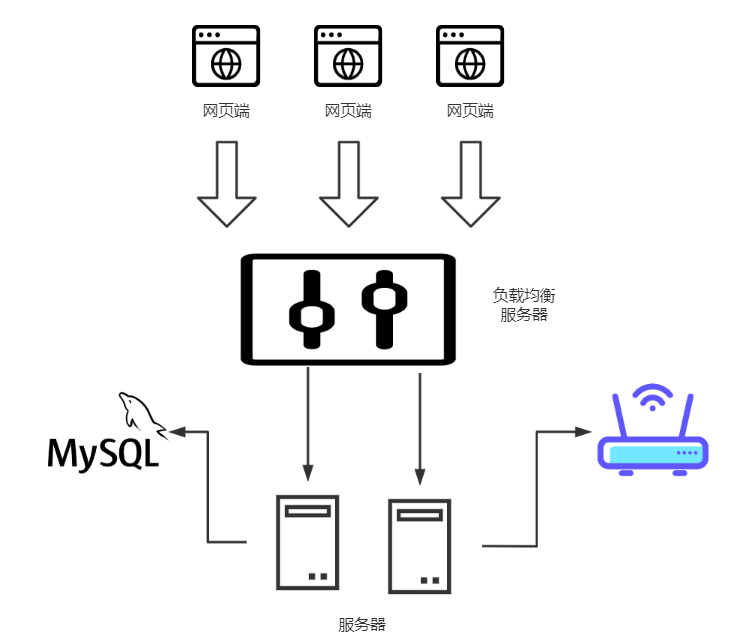


图2.30 系统部署图

### 2.5.3数据定义

系统关键数据实体表包含以下5个：用户信息表、图书信息表、管理员信息表、借阅记录信息表和公告信息表。用户信息表中包含用户级基本信息，主键为用户 ID，用来唯一识别用户。用户信息表（users） 具体描述如表2-1所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 可为空 | 描述 |
| reader\_id | varchar(20) | N | 识别用户的唯一id |
| reader\_name | varchar(30) | N | 用户自己设置的昵称 |
| password | varchar(11) | N | 用户自己设置的密码 |
| gender | varchar(5) | N | 用户的宿舍号 |
| type | varchar(50) | N | 用户的昵称 |
| balance | varchar(50) | N | 用户账号的余额 |
| chances | int（4） | N | 用户账号的可借阅次数 |
| phone | varchar(11) | N | 用户用于验证的手机号 |

表2-1用户信息表词典

图书信息表中包含图书基本信息，主键为ISBN码，用来唯一识别书籍。图书信息表 具体描述如表2-2所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 可为空 | 描述 |
| ISBN | varchar(20) | N | 识别图书的唯一id |
| book\_name | varchar(20) | N | 图书的书名 |
| author | varchar(20) | N | 图书的作者 |
| book\_concern | varchar(20) | N | 图书的出版社 |
| category | varchar(20) | N | 图书的类型 |
| location | varchar(20) | Y | 图书在图书馆存放的位置 |
| is\_rent | Double | N | 图书是否被借的标志 |
| price | varchar(20) | N | 图书的价格 |
| sum | int | N | 图书的总库存 |
| now\_amount | int | N | 图书的现有库存 |
| origin | varchar(20) | N | 图书的来源 |
| qrcode | varchar(20) | N | 图书的二维码 |
| img | varchar(20) | Y | 图书的图片 |

表2-2图书信息表词典

管理员信息表中包含管理员基本信息，主键为管理员ID，用来唯一识别管理员。管理员信息表具体描述如表2-3所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 可为空 | 描述 |
| admin\_id | varchar(20) | N | 识别管理员的唯一id |
| password | varchar(20) | N | 管理员自己设置的密码 |
| admin\_name | varchar(20) | N | 管理员的昵称 |
| gender | varchar(20) | N | 管理员的性别 |
| phone | varchar(20) | N | 管理员的电话号码 |

表2-3管理员信息表词典

借阅记录信息表中包含借阅记录基本信息，主键为用户ID和ISBN，用来唯一识别借阅信息记录。借阅记录信息表具体描述如表2-4所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 可为空 | 描述 |
| id | varchar(20) | N | 识别借阅信息的id |
| ISBN | varchar(20) | N | 识别图书的唯一id |
| book\_name | varchar(20) | N | 图书的书名 |
| reader\_id | varchar(20) | N | 识别借阅书籍的读者的id |
| reader\_name | varchar(2) | N | 借阅书籍的读者的昵称 |
| create\_time | DATETIME | N | 借阅开始时间 |
| end\_time | DATETIME | N | 借阅截至的时间 |
| return\_time | DATETIME | Y | 读者归还图书的时间 |
| status | varchar(20) | Y | 借阅记录的状态 |
| forfeit | varchar(20) | Y | 该条借阅记录的罚款 |

表2-4 借阅记录信息表词典

公告信息表中包含公告基本信息，主键为id，用来唯一识别公告信息。公告信息表具体描述如表2-5所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 可为空 | 描述 |
| id | varchar(20) | N | 记录信息记录的编号 |
| info | varchar(255) | N | 信息记录的主题内容 |
| time | DATETIME | N | 信息记录创建的时间 |
| type | varchar(20) | N | 信息记录的类型 |
| reader\_id | varchar(20) | N | 发送信息的读者id |

表2-5公告信息表词典

## 2.6运行设计

### 2.6.1运行模块关系

首先需要登录系统，此后便可进行其他模块的操作。

### 2.6.2运行控制

运行控制将严格按照模块间函数调用关系来实现。

在网络传输方面，客户端在发送数据后，将等待服务端的确认收到信号，并等待服务器发送响应数据，然后对数据进行确认。服务器在接到数据后发送确认信号，再对数据进行预处理，数据库操作，业务逻辑操作后，将返回信息送回客户端，并等待确认。

### 2.6.3运行时间

本产品在性能方面的主要需求是满足用户响应的及时性，响应时间能够满足在 Wi-Fi 网络环境下在3秒之内完成 Web 端的内容加载和渲染，通过一定的优化，响应时间同样也能够满足移动端和 PC 端上在1秒之内完成数据加载和渲染。

（1）系统的响应时间（页面跳转时间）<=1.0s

（2）数据分析相关功能响应时间 <= 3.0s

（3）硬件响应时间 <=1.0s

## 2.7错误/异常处理设计

程序分为服务器端和客户端。

在服务器端，程序异常采用 try-catch-finally 捕获异常，在该系统中对异常的处理采用将捕获的异常序列化后保存到日志系统中。具体流程是，任何程序捕获到异常后，将实例化 BackendException 类，调用 LogService 的 LogException方法，由该方法保存异常到日志中。

在客户端，程序异常采用 try-catch-finally 捕获异常，对于由程序本身问题出现的异常，将异常内容和运行环境等信息序列化后发送到服务端日志系统，并存储到日志中，方便开发人员进行复现、排查和修复。具体流程是，捕获到异常后，将不同平台的异常信息序列化为相同的 FrontendException 类，发送到服务器端 LogService，由LogException 方法保存异常到日志中。

本系统的业务主要在若干子系统中所体现，拟采用两种方法处理。

一种是采用各接口自带的验证机制进行校验，严格按照数据字典中所确定的数据类型、数据长度和数据范围完成数据属性设置。在页面层数据项后，输入数据若不符合规范，则采用对话框等方式显示异常信息，不将异常信息保存到日志。

另一种如果是由于数据库操作或业务逻辑导致异常，则通过try-catch-finally语句捕获异常，处理方式与异常处理相同，但需要将异常信息保存到日志系统备查。

除了以上两种系统内定义完成的异常处理，我们还允许系统管理员通过可视化界面完成异常处理的更新与添加。

## 2.8数据恢复方案

备份的目的是在出现故障的情况下能尽快的恢复系统，为保证能快速、准确地正常恢复，我们应该明确地定义恢复的策略。根据不同的故障，我们有以下的恢复策略。

### 2.8.1 全崩溃恢复机制

因为本平台设置了远程灾难备份中心，也有一定的冗余备份，出现此类崩溃需要启动冗余机器，然后迅速修复原有机器，就可以将系统恢复到灾难发生前的状态。恢复步骤如下：

（1）启用备份机器和系统；

（2）修复服务器硬件设备；

（3）修复网络硬件设备；

（4）从系统备份磁盘中恢复服务器软件环境；

（5）从应用系统备份磁盘中恢复应用系统；

（6）从冗余机器中恢复数据；

（7）按照需要恢复应用系统的日志。

### 2.8.2 服务器崩溃恢复机制

服务器出现崩溃，此时因为数据存储在磁盘阵列中，并没有受到影响，只要快速恢复服务器上的操作系统、数据库、应用系统即可。

（1）修复服务器硬件设备；

（2）从系统备份磁盘中恢复服务器操作系统、数据库系统；

（3）从应用系统备份磁盘中恢复应用系统；

（4）按照需要恢复应用系统的日志。

### 2.8.3 磁盘阵列崩溃恢复机制

磁盘阵列崩溃，数据将全部丢失，恢复的对象主要是数据。恢复步骤如下：

（1）修复磁盘阵列；

（2）从最近的全备份磁盘恢复到全备份时数据状态；

（3）按照全备份后进行的增量备份的次序逐个恢复。

### 2.8.4系统软件全崩溃恢复机制

软件系统全崩溃的情况下，由于数据存储在磁盘阵列中，可能受到损失，也可能为受到损失。恢复步骤如下：

（1）从系统备份磁盘中恢复服务器操作系统、数据库系统；

（2）从应用系统备份磁盘中恢复应用系统；

（3）进行数据库检查，如果数据未损坏，则恢复已完成。否则执行下述步骤：

（4）从最近的全备份磁盘恢复到全备份时数据状态；

（5）按照全备份后进行的增量备份的次序逐个恢复增量备份；

（6）按照需要恢复应用系统的日志。

### 2.8.5 操作系统崩溃恢复机制

操作系统崩溃的情况下，由于数据存储在磁盘阵列中，可能受到损失，也可能为受到损失。恢复步骤如下：

（1）从系统备份磁盘中恢复服务器操作系统；

（2）从应用系统备份磁盘中恢复应用系统；

（3）进行数据库检查，如果数据未损坏，此时恢复已完成，否则执行以下步骤：

（4）最近的全备份磁盘灰度到全备份时数据状态；

（5）按照全备份后进行的增量备份的次序逐个恢复增量备份；

（6）按照需要恢复应用系统的日志。

### 2.8.6 数据库系统崩溃恢复机制

数据库系统崩溃的情况下，由于数据存储在磁盘阵列中，可能受到损失，也可能未受到损失。恢复步骤如下：

（1）从系统备份磁盘中恢复数据库系统；

（2）进行数据库检查，如果数据未损坏，此时恢复已完成，否则执行以下步骤：

（3）从最近的全备份磁盘灰度到全备份时数据状态；

（4）按照全备份后进行的增量备份的次序逐个恢复增量备份；

（5）按照需要恢复应用系统的日志。

## 2.9数据备份方案

平台正式上线后主要包括四个部分：硬件设备，软件环境，应用系统和数据。下面从这四个方面介绍下系统的数据备份方案。

（1）硬件设备

对于硬件设备故障，并非简单的加强管理就可以避免的，往往必须增加投资，进行硬件设备的冗余，以增强系统的高可用性。律佳使用多机集群，存储设备采用 RAID。

（2）软件环境

软件的故障一般是由于人为操作因素引起的，影响应用系统的正常运行，对于该类故障，有两种解决方式：

启动冗余备份系统，即将当前生产机离线，将备份机启动，再逐步恢复原系统机器的软件环境。

从软件备份中恢复系统，在这种情况下，要求对原系统的操作系统、数据库系统等已经定时做好完整的备份这样可以是恢复成为简单机械、按部就班的工作。

（3）应用系统

应用系统的故障大多也是由于人为操作引起的。应用系统的备份和恢复和软件环境的备份恢复类似。

（4）数据

数据是本系统的关键，为了系统的完整性考虑，在条件允许的情况下，将所有系统数据全部备份。同时进行分级备份，视可重复性、重要性、数据量、处理的时间和难度据顶备份的级别。数据备份定期进行，所有数据每周备份一次，重要数据每天备份一次。

## 2.10系统配置策略

### 2.10.1客户端

Internet 连接

移动端：网页兼容移动端，移动设备只需安装浏览器即可访问。

### 2.10.2服务器端目前

本系统使用阿里云服务器作为服务器

## 2.11系统部署方案

本项目根据实际情况进行分期部署，在项目投入初期考虑用户规模可能较小，但为了保证传输速率采用阿里云服务器作为服务器来进行维持软件的正常运行，在随后发展当中考虑进行对于内网穿透和云服务器加大带宽保证传输速率

## 2.12规范

系统的设计，编码以及测试维护将严格按照需求分析中的约束和设计标准开发。所有HTML代码将遵照5.0版本。系统将采用标准的MySQL数据库引擎。Python环境统一配置为3.10。

# 3.详细设计

系统根据预期的功能与目标，将系统划分为管理员端和用户端，其中用户端分为四个子模块:信息安全模块、图书借阅模块、聊天咨询模块、推荐服务模块，而管理员端同样分为四个子模块:信息安全模块、图书信息管理模块、咨询回复模块、逾期处罚模块，以下将进行详细得介绍

## 3.1 管理员端

### 3.1.1信息安全模块

**功能描述:**

这一模块主要是对管理员对自己和读者得信息进行管理，可以自主的进行账户的注册名称等信息的选择，同时还拥有去修改读者信息的功能，通过读者允许后也可对读者的账户进行注销处理，管理员申请账户、经过系统核验后登录进入系统，就可以对自身信息和读者信息进行修改。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **用户群** | **优先级** |
| **信息安全模块** | **注册** | **管理员自主选择昵称和密码惊进行注册，同时需要选择进行手机号或者邮箱验证** | **全体** | **高** |
| **登录** | **登录IDO图书管理的系统** | **全体** | **高** |
| **信息修改** | **管理员修改自己的个人信息** | **全体** | **高** |

表3-1 信息安全模块功能描述

**性能描述:**

在500ms以内完成管理的登录与注册操作。

### 3.1.2输入

对于管理员注册：用户名、登录密码、手机。

对于管理员登陆：用户名、登录密码、验证码。

对于管理员修改个人信息：无

对于管理员修改管理读者信息：读者ID

### 3.1.3输出

对于管理员注册：管理员是否注册成功

对于管理员登陆：管理员是否登陆成功

对于管理员修改个人信息：管理员是否进行了信息的修改

对于管理员修改管理读者信息：管理员是否对读者信息进行了修改

### 3.1.4程序逻辑

模块程序流程图如图所示

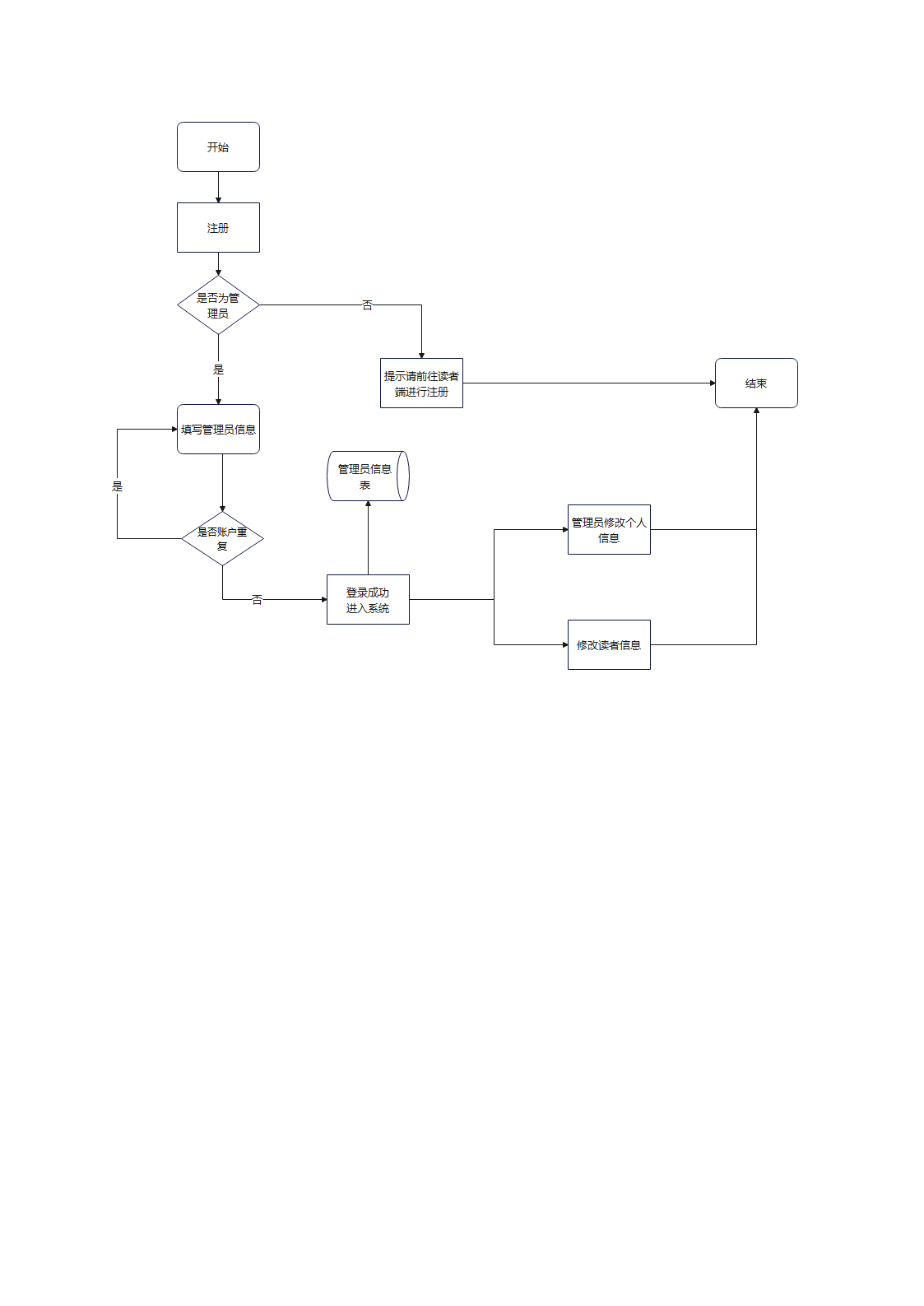


图3.1信息安全模块流程图

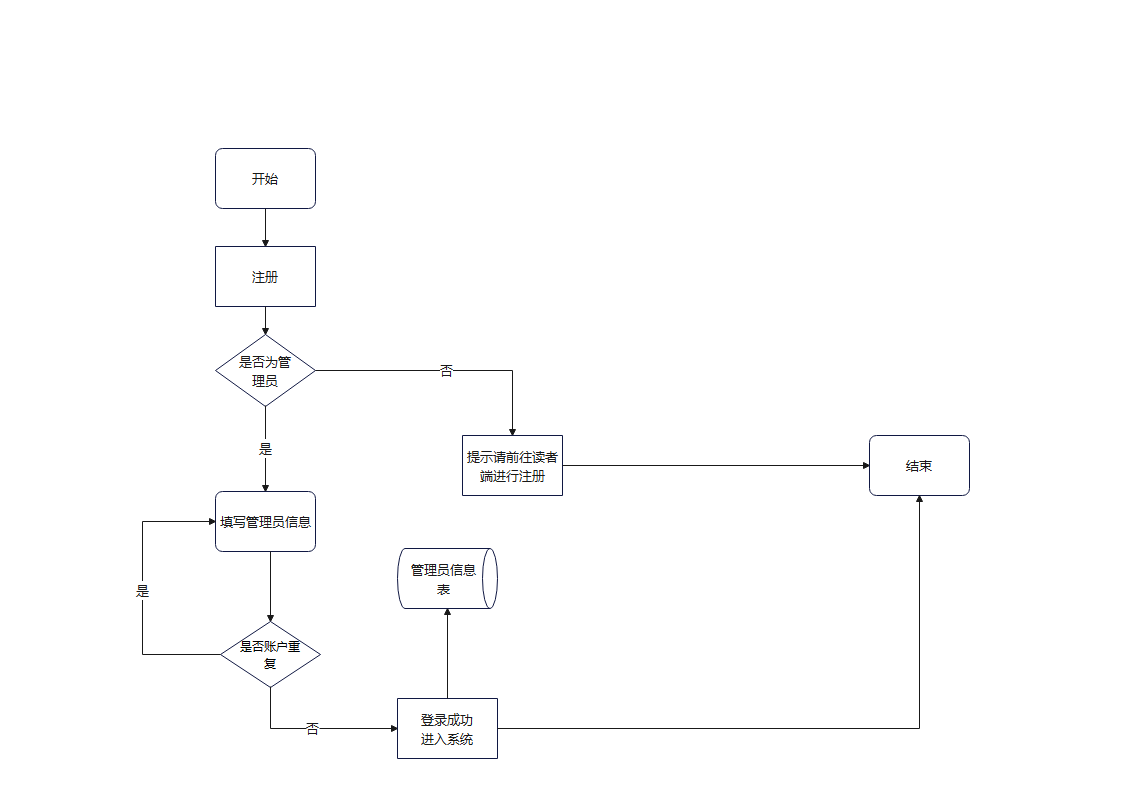


图3.2注册流程图

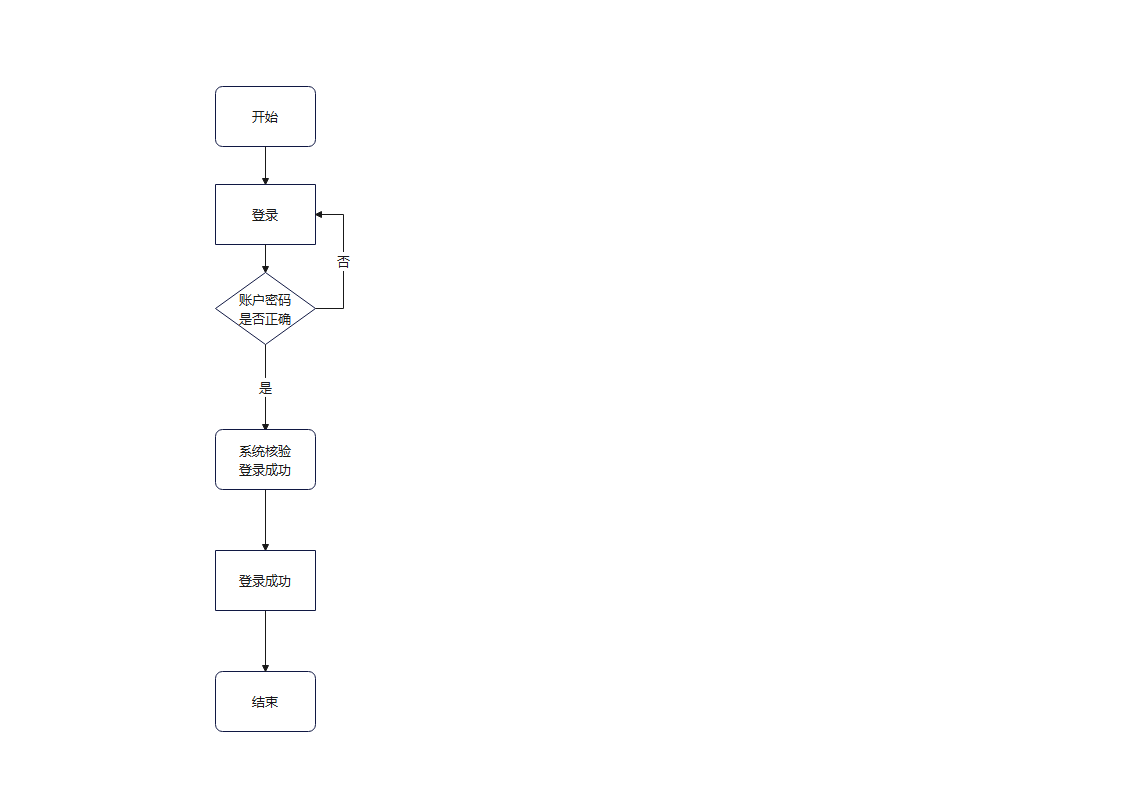


图3.3登录流程图

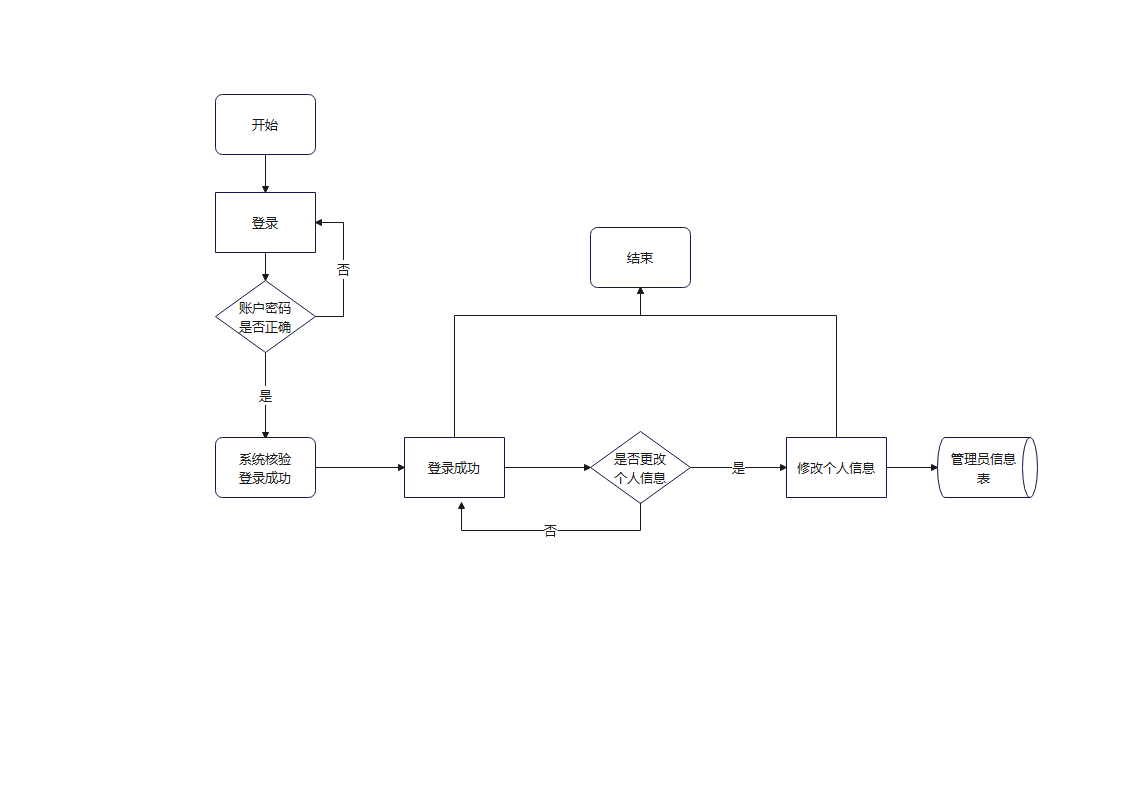


图3.4管理员修改个人信息流程图

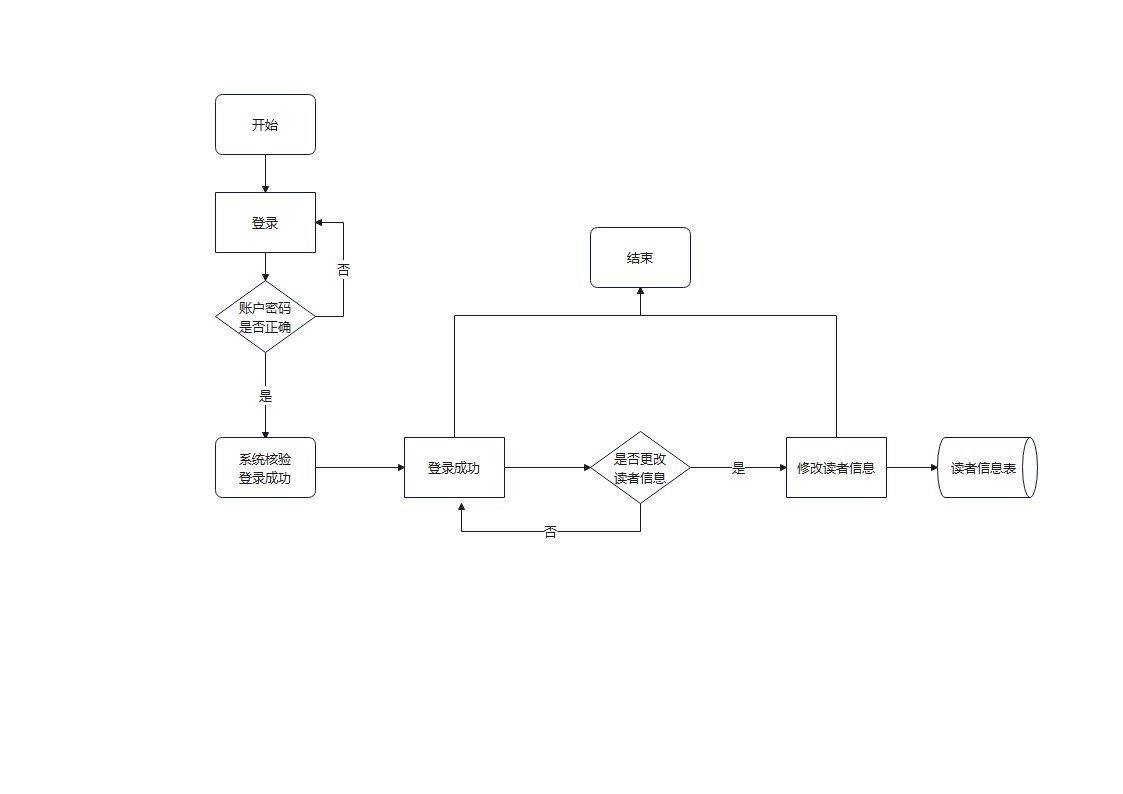


图3.5管理员修改读者信息流程图

### 3.1.6接口

输入：管理员id

数据来源：Redis后端数据库

数据格式：{管理员id，密码，姓名，性别，电话}

数量：5

输出：管理员的个人信息

### 3.1.7存储分配

数据存储到管理员信息表

### 3.1.8限制条件

对于用户名：

* 长度介于4-15字符之间
* 只能包含字母、数字、下划线
* 不能以数字开头
* 不能使用已被注册的用户名

对于密码：

* 长度介于4-15字符之间
* 只能包含字母、数字、!、"、#、$、&、(、)、\*、+、,、-、.、/、:、;、<、=、>、?、@、^、\_、`、{、|、}、~、]、\*。

对于手机号：

* 必须符合国际手机号规则

## 3.2图书信息管理模块

### 3.2.1功能描述

图书信息管理模块可以对系统当中的录入的图书以及借阅图书进行统一的管理，可以帮助用户借阅、归还、转借、产生图书报表，可以帮助图书管理员更好的管理图书馆内的书籍信息，并且加以分析借阅情况。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **用户群** | **优先级** |
| **图书信息管理模块** | **图书借阅** | **管理员核验用户资格，向用户发送二维码进行图书借阅** | **全体** | **高** |
| **图书归还** | **管理员查看书籍是否完好，通过对二维码扫描图书归还** | **全体** | **高** |
| **图书信息录入、删除** | **管理员对于图书馆的书籍信息进行增删改查** | **全体** | **高** |
| **图书互借** | **管理员审核用户的上传图书，并对借用此图书的用户提供，图书提供者的收款码** | **全体** | **高** |
| **图书报表** | **管理员对于借阅记录进行报表查看，统计分析便于后期对书籍资源的管理** | **全体** | **高** |

表3-2 图书信息管理功能描述

### 3.2.2性能描述

延迟时间小于1000ms

响应时间小于500ms

### 3.2.3输入

无

### 3.2.4输出

图书报表、图书详细、图书借阅记录

### 3.2.5程序逻辑

模块程序流程图如图所示

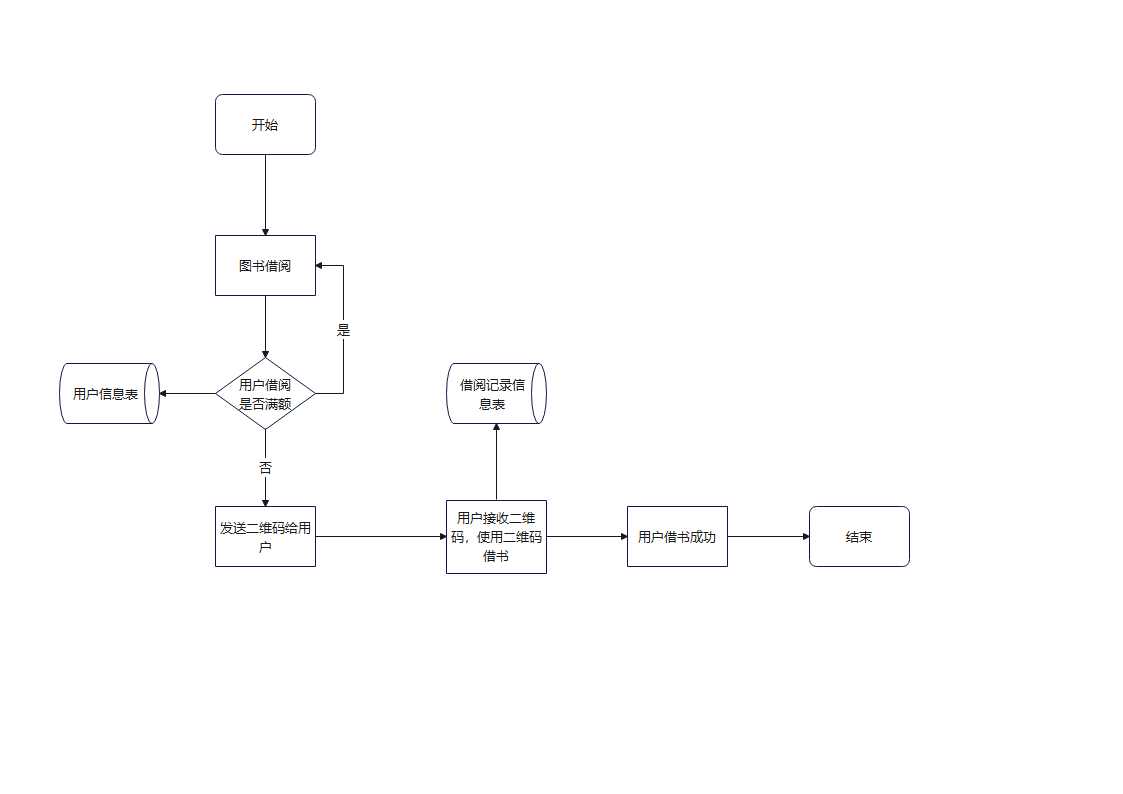


图3.6图书借阅流程图

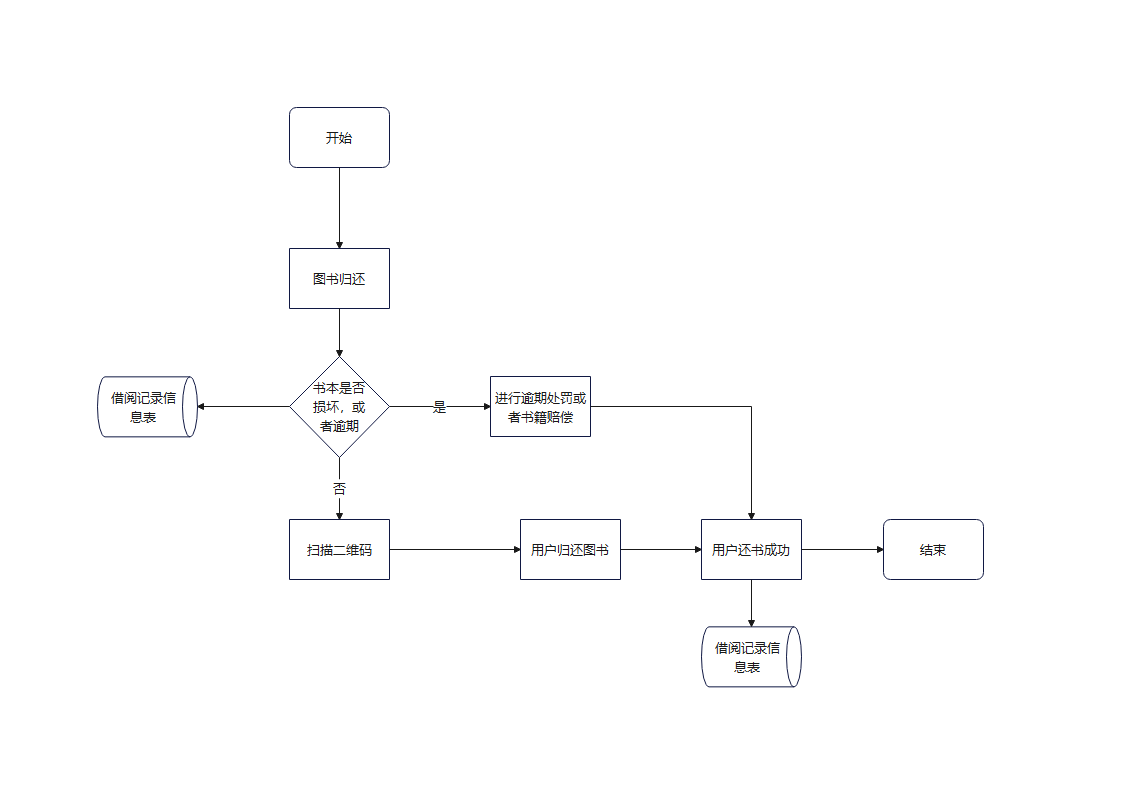


图3.7图书归还流程图

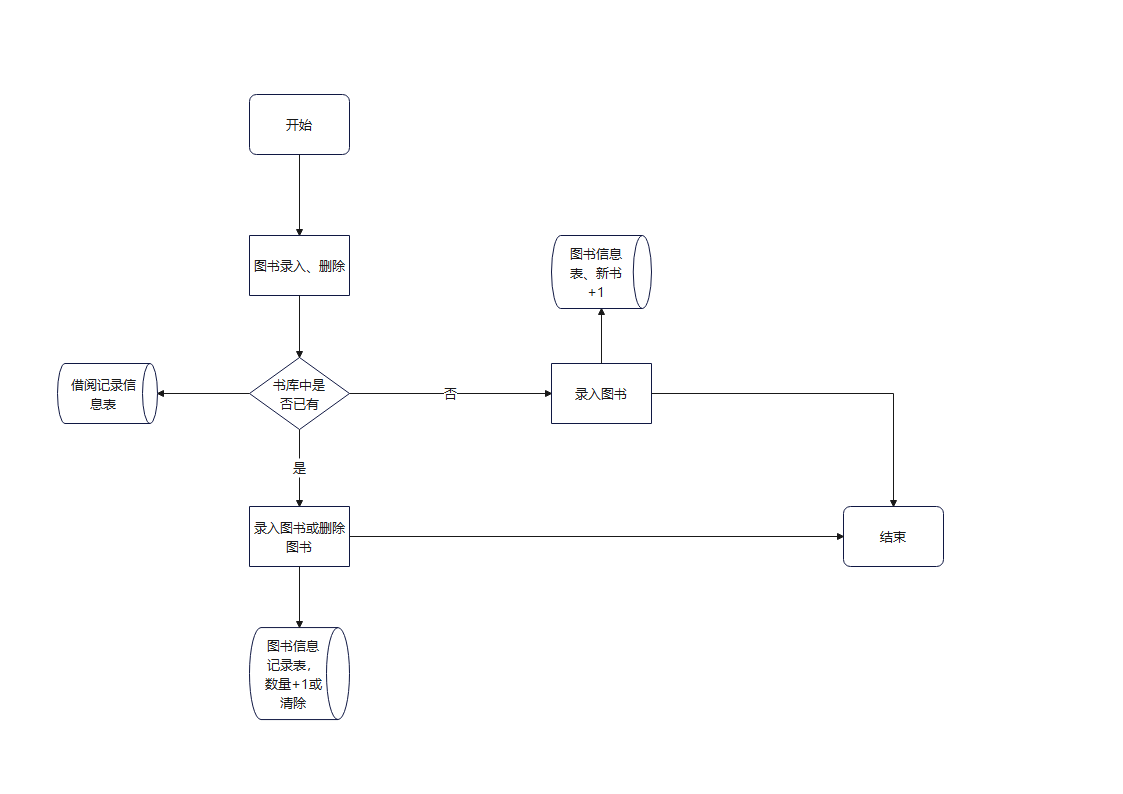


图3.8图书录入流程图

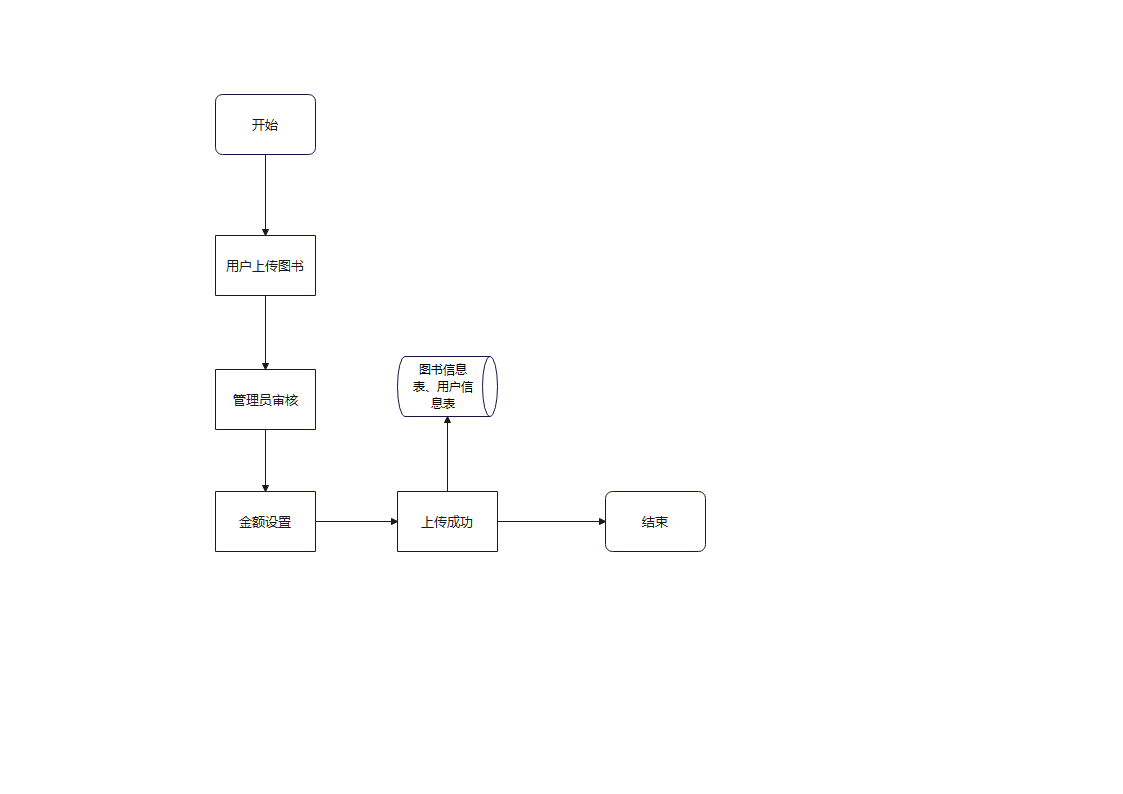


图3.9图书互借流程图

### 3.2.6接口

**1）借阅归还互借录入需求**

输入：书籍信息和读者id

数据来源：redis数据库、摄像头

数据格式：{书籍信息、读者id、二维码}

数量：1

输出：借还书成功

**2）图书报表需求**

输入：借阅记录信息

数据来源：redis数据库

数据格式：{借阅记录信息}

数量：1

输出：图书折线图和柱状图产生

### 3.2.7存储分配

数据存储到图书信息存储表

## 3.3咨询回复模块

### 3.3.1功能描述

这一模块主要是读者和管理员之间进行实时通信，以及公告的留言，帮助管理员更好的了解需求：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **用户群** | **优先级** |
| **咨询回复** | **读者回复** | **用户和管理员进行实时的聊天通信，进行信息询问** | **全体** | **高** |
| **信息公告栏** | **管理员向整个图书馆的用户发布关于图书馆的最新信息，同时得到用户新的诉求** | **全体** | **高** |

表3-3 咨询回复功能描述

### 3.3.2性能描述

1000ms内进行消息的实时传输

### 3.3.3输入

相关信息

### 3.3.4输出

语言文本

### 3.3.5程序逻辑

模块程序流程图如图所示

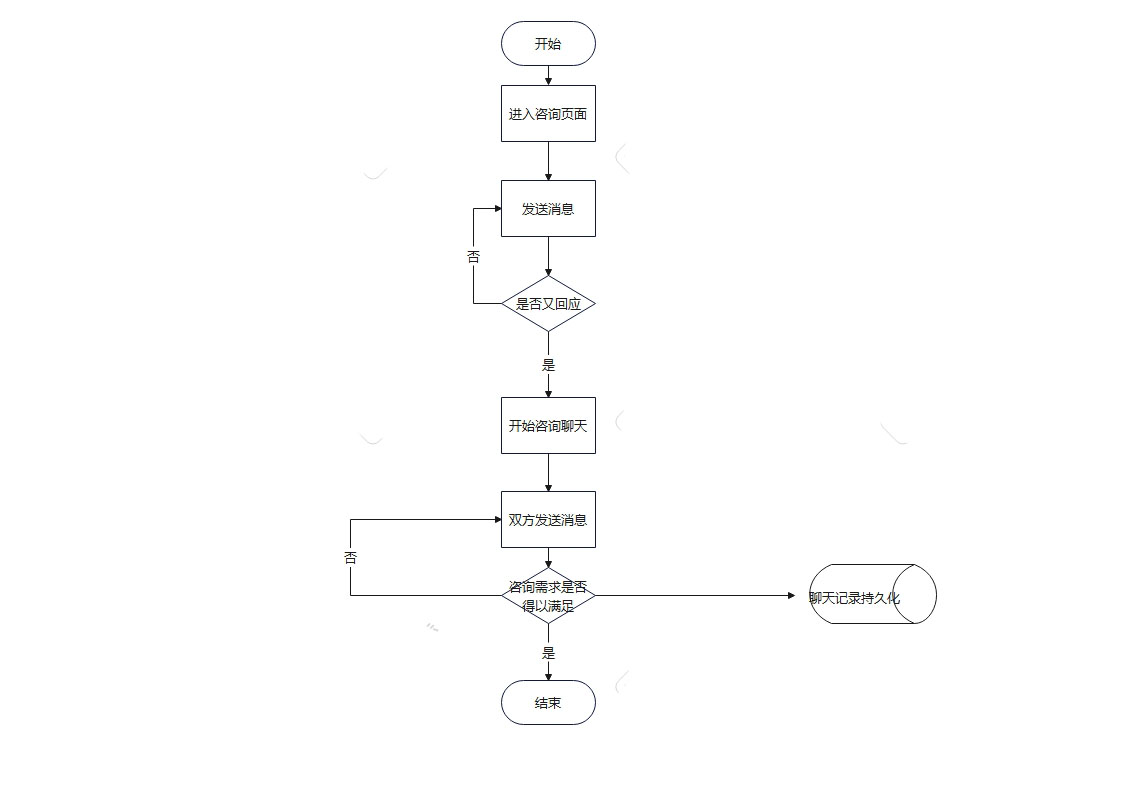


图3.10消息咨询程序流程图

### 3.3.6接口

输入：管理员想发送的信息

数据来源：文本

数据格式：{用户id，文本信息}

数量：1

输出：读者收到信息

### 3.3.7存储分配

无

## 3.4逾期处罚模块

### 3.4.1功能描述

这个模块是管理员对读者进行借书逾期的提醒，逾期进行短信发送提醒用户，用户若逾期则进行罚款，若用户损坏图书同样进行罚款

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **用户群** | **优先级** |
| **逾期处罚** | **逾期信息提示** | **管理员向接近逾期和已逾期的用户发送提示还书的短信** | **全体** | **高** |
| **处罚** | **管理员针对逾期的用户进行每日0.1元的处罚，对于丢失或者损坏图书的按照原价赔偿** | **全体** | **高** |

表3-4 逾期处罚功能描述

### 3.4.2性能描述

在1000s内能发送短信

### 3.4.3输入

用户个人id

### 3.4.4输出

逾期短信提醒、处罚罚款

### 3.4.5程序逻辑

模块程序流程图如图所示：

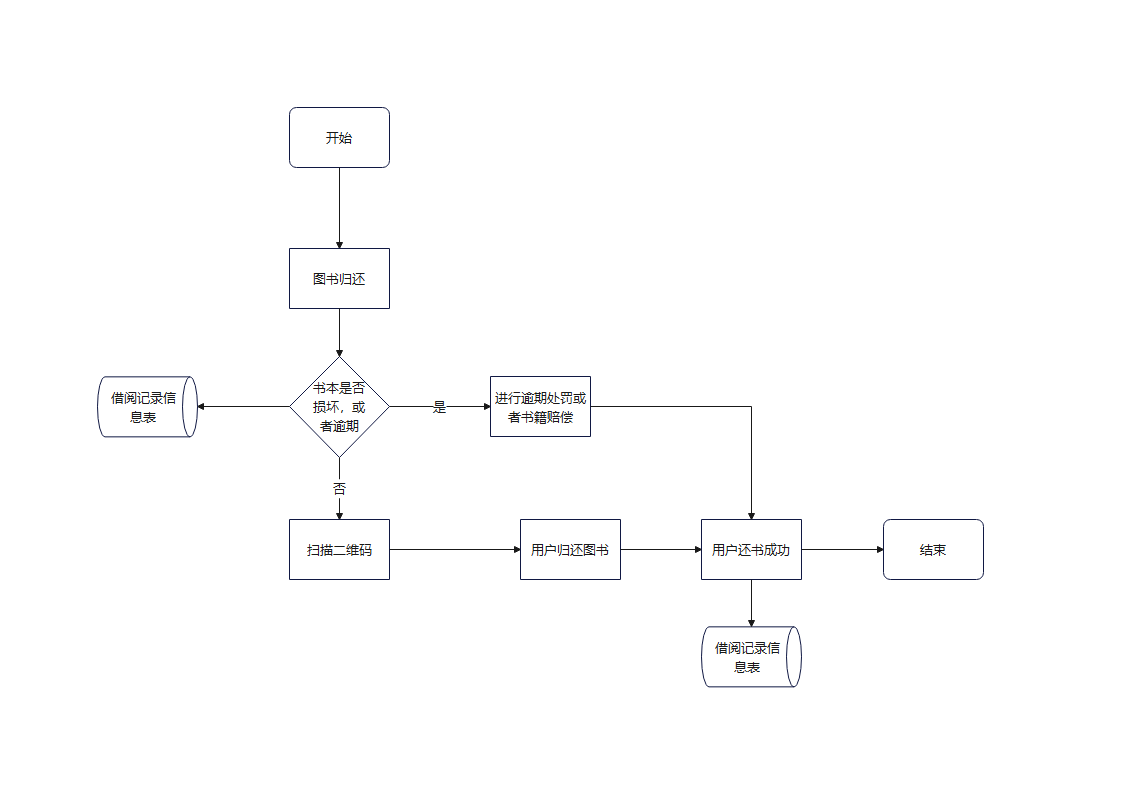


图3.11逾期还书处罚程序流程图

### 3.4.6接口

**1）短信需求**

输入：成员id

数据来源：HAOS后端数据库

数据格式：{逾期时长、读者id}

数量：1

输出：发送短信提示

**2）处罚需求**

输入：成员id

数据来源：支付宝api

数据格式：{成户id}

数量：1

输出：罚款处理

### 3.4.7存储分配

数据存储到用户信息表

## 3.5读者端

### 3.5.1信息安全模块

**功能描述：**

这一模块主要是读者对于自己进入图书馆系统进行账户的登录和注册、需要输入合格手机号邮箱等方式，通过系统的核验之后才可有正常进行系统当中，并且在个人中心当中可以进行修改个人的信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **用户群** | **优先级** |
| **信息安全模块** | **注册** | **用户自主选择昵称和密码惊进行注册，同时需要选择进行手机号或者邮箱验证** | **全体** | **高** |
| **登录** | **登录IDO图书管理的系统** | **全体** | **高** |
| **信息修改** | **用户修改自己的个人信息** | **全体** | **高** |
| **账户充值** | **用户对自己的账户余额进行充值** | **全体** | **高** |

表3-5 信息安全模块功能描述

### 3.5.2性能描述

在500ms以内完成读者的登录与注册操作。

### 3.5.3输入

对于读者注册：用户名、登录密码、手机。

对于读者登陆：用户名、登录密码、验证码。

对于读者修改个人信息：无

### 3.5.4输出

对于读者注册：管理员是否注册成功

对于读者登陆：管理员是否登陆成功

对于读者修改个人信息：管理员是否进行了信息的修改

### 3.5.5程序逻辑

模块程序流程图如图所示

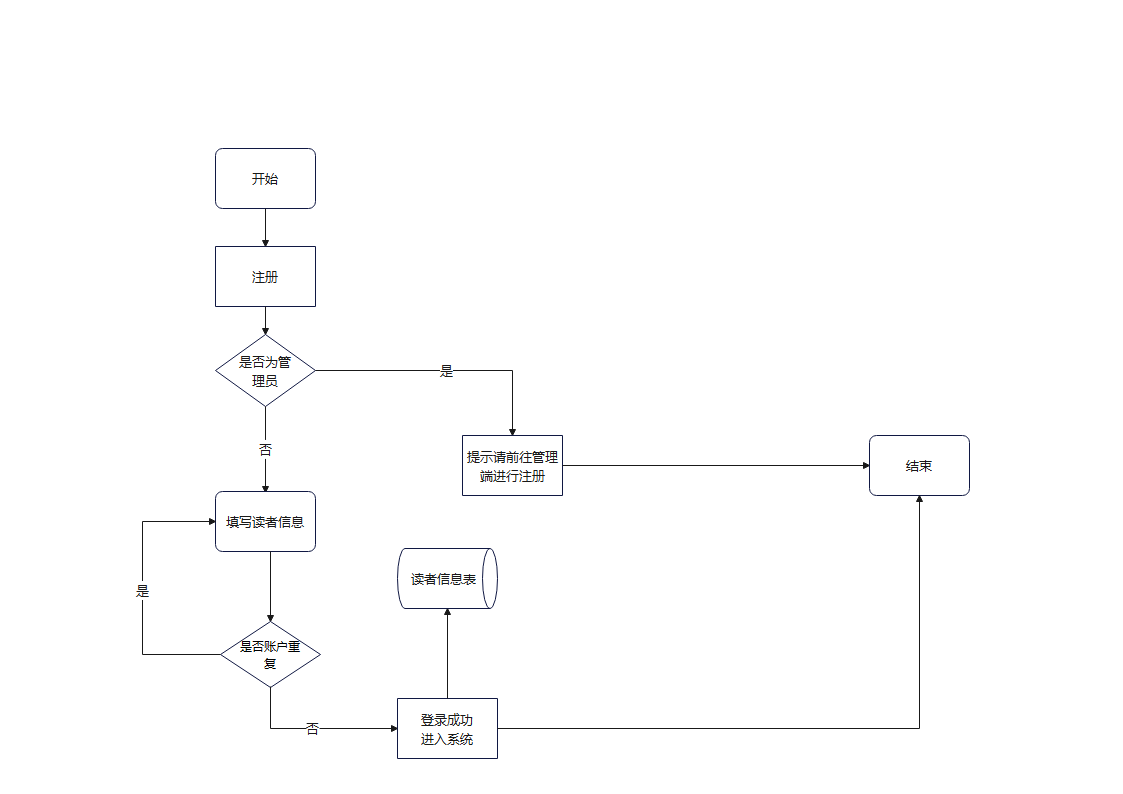


图 3.12用户注册流程图

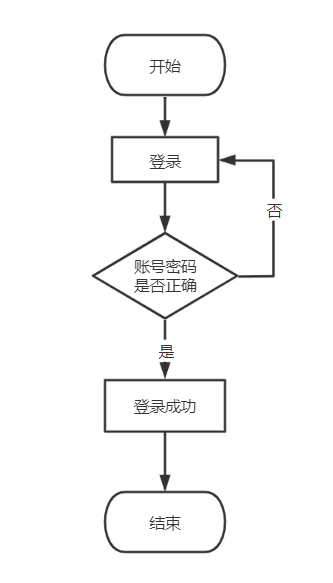


图3.13登录流程图

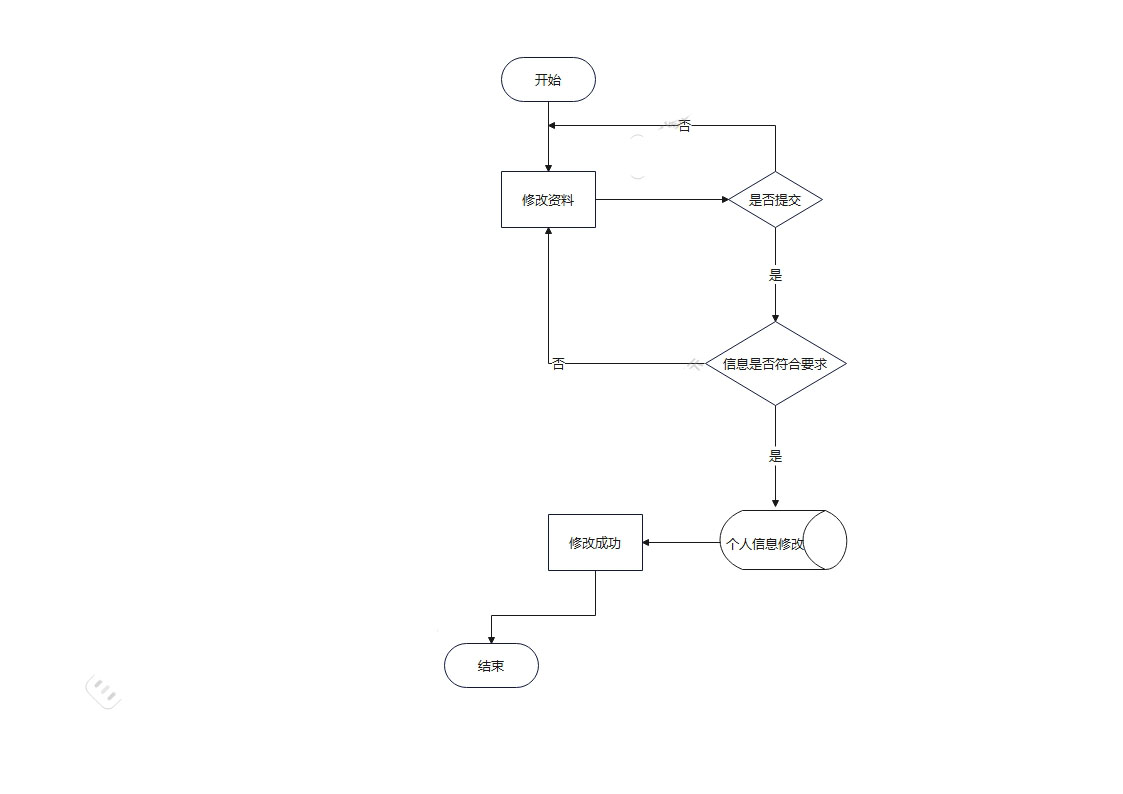
、

图3.14信息修改流程图

### 3.5.6接口

输入：用户id

数据来源：Redis后端数据库

数据格式：{用户id，密码，姓名，性别，电话}

数量：5

输出：用户的个人信息

### 3.5.7存储分配

数据存储到用户信息表

### 3.5.8限制条件

对于用户名：

* 长度介于4-15字符之间
* 只能包含字母、数字、下划线
* 不能以数字开头
* 不能使用已被注册的用户名

对于密码：

* 长度介于4-15字符之间
* 只能包含字母、数字、!、"、#、$、&、(、)、\*、+、,、-、.、/、:、;、<、=、>、?、@、^、\_、`、{、|、}、~、]、\*。

对于手机号：

* 必须符合国际手机号规则

## 3.6图书借阅管理模块

### 3.6.1功能描述

图书信息管理模块可以对系统当中的录入的图书以及借阅图书进行统一的管理，可以帮助用户借阅、归还、转借、产生图书报表，可以帮助图书管理员更好的管理图书馆内的书籍信息，并且加以分析借阅情况

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **用户群** | **优先级** |
| **图书借阅模块** | **图书借阅** | **用户通过二维码进行图书借阅** | **全体** | **高** |
| **图书归还模块** | **用户通过而二维码扫描进行图书归还** | **全体** | **高** |
| **逾期提示处罚** | **用户收到逾期提示信息，及时对图书进行归还，逾期进行一定金额罚款** | **全体** | **高** |
| **图书互借** | **用户上传图书给其他用户进行借阅并且收取一定的费用** | **全体** | **高** |

表3-6 图书借阅管理功能描述

### 3.6.2性能描述

延迟时间小于1000ms

响应时间小于500ms

### 3.6.3输入

无

### 3.6.4输出

图书详细、图书借阅记录

### 3.6.5程序逻辑

模块程序流程图如图所示

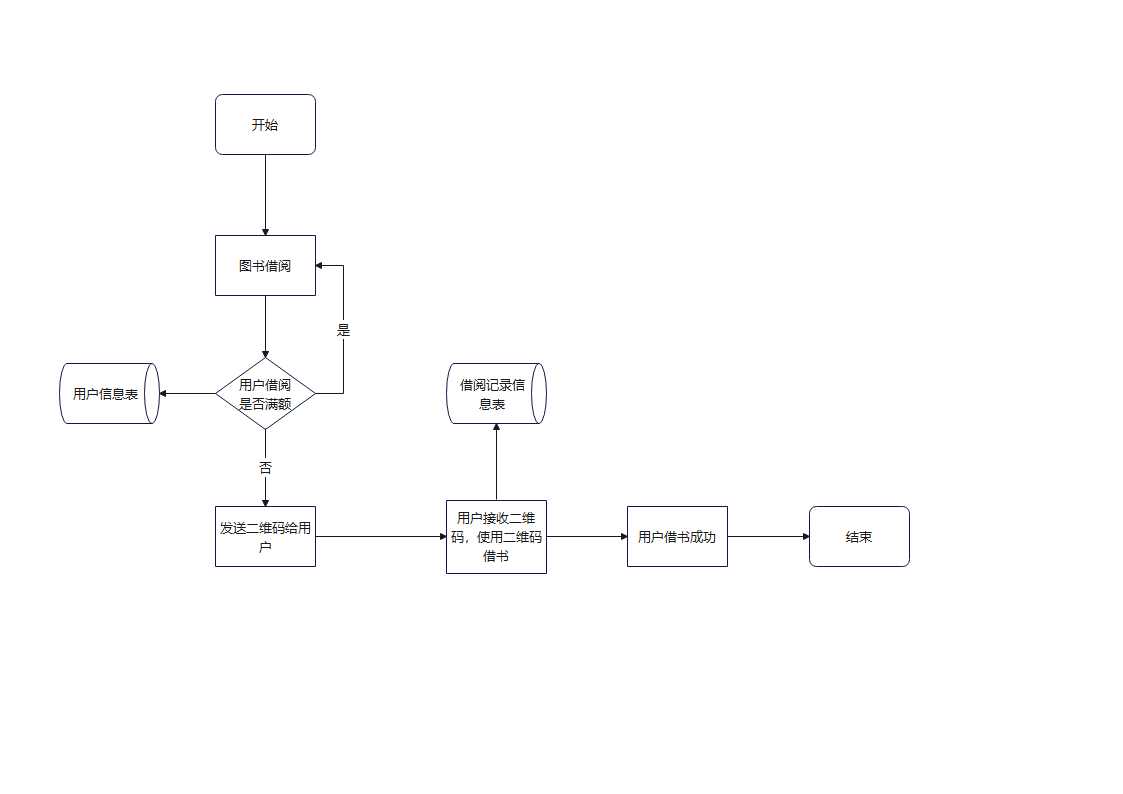


图3.15图书借阅流程图

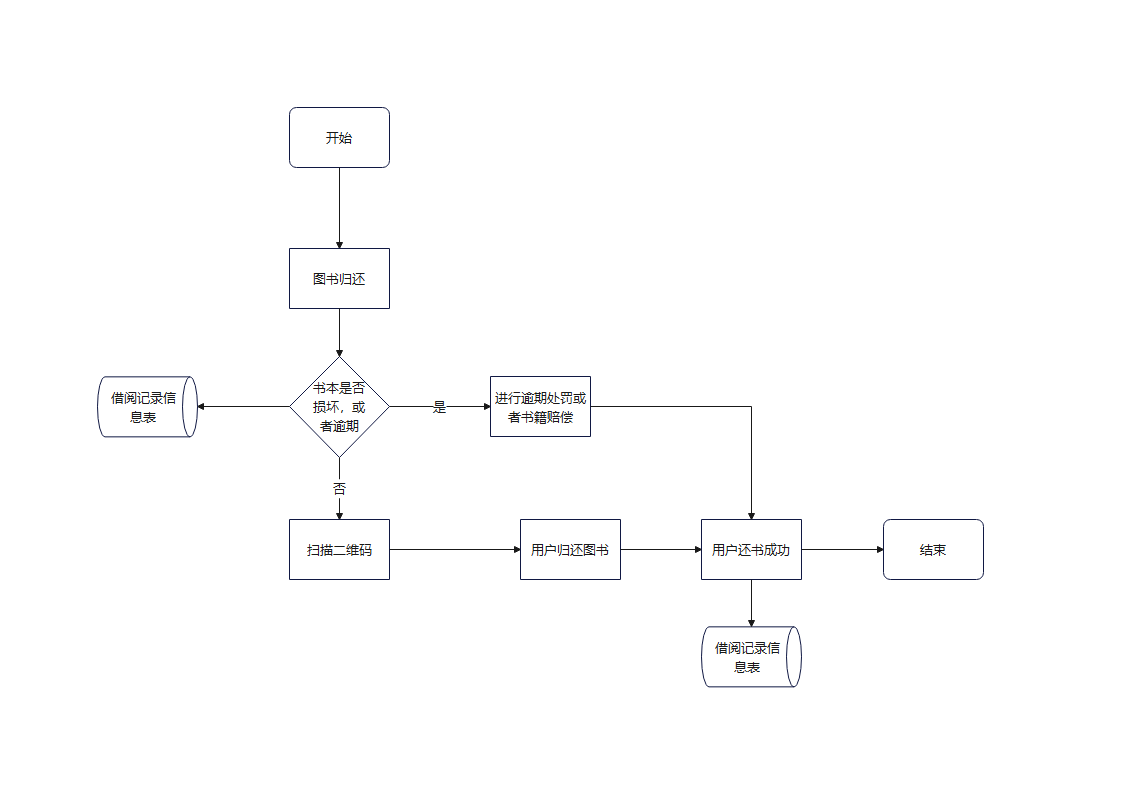


图3.16图书归还流程图

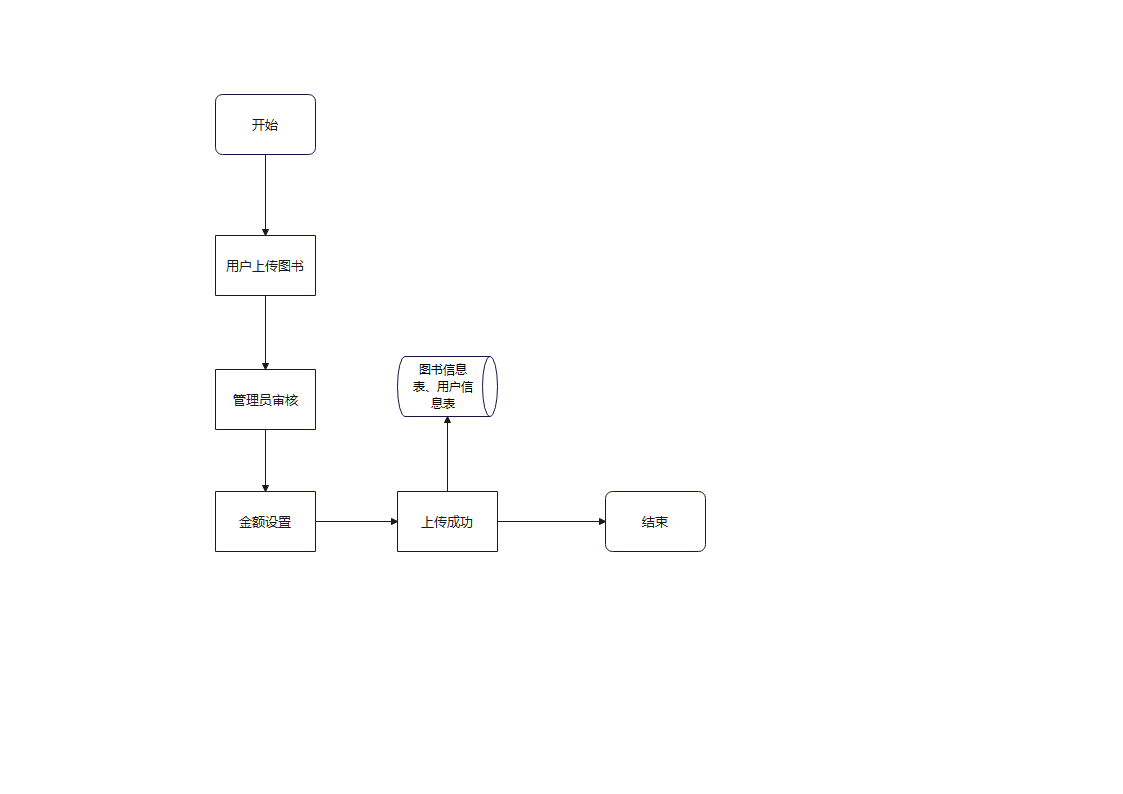


图3.17图书互借流程图

### 3.6.6接口

**1）借阅归还互借录入需求**

输入：书籍信息和读者id

数据来源：redis数据库、摄像头

数据格式：{书籍信息、读者id、二维码}

数量：1

输出：借还书成功

### 3.6.7存储分配

数据存储到图书信息存储表。

## 3.7聊天咨询模块

### 3.7.1功能描述

这一模块主要是读者和管理员之间进行实时通信，以及公告的留言，帮助用户更快的掌握图书馆的第一手信息。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **用户群** | **优先级** |
| **聊天咨询** | **读者咨询** | **用户和管理员进行实时的聊天通信，进行信息询问** | **全体** | **高** |
| **信息公告栏** | **用户通过公告栏对图书馆发布的信息进行了解和评价以提出个人诉求** | **全体** | **高** |

表3-7 聊天咨询功能描述

### 3.7.2性能描述

1000ms内进行消息的实时传输

### 3.7.3输入

相关信息

### 3.7.4输出

语言文本

### 3.7.5程序逻辑

模块程序流程图如图所示

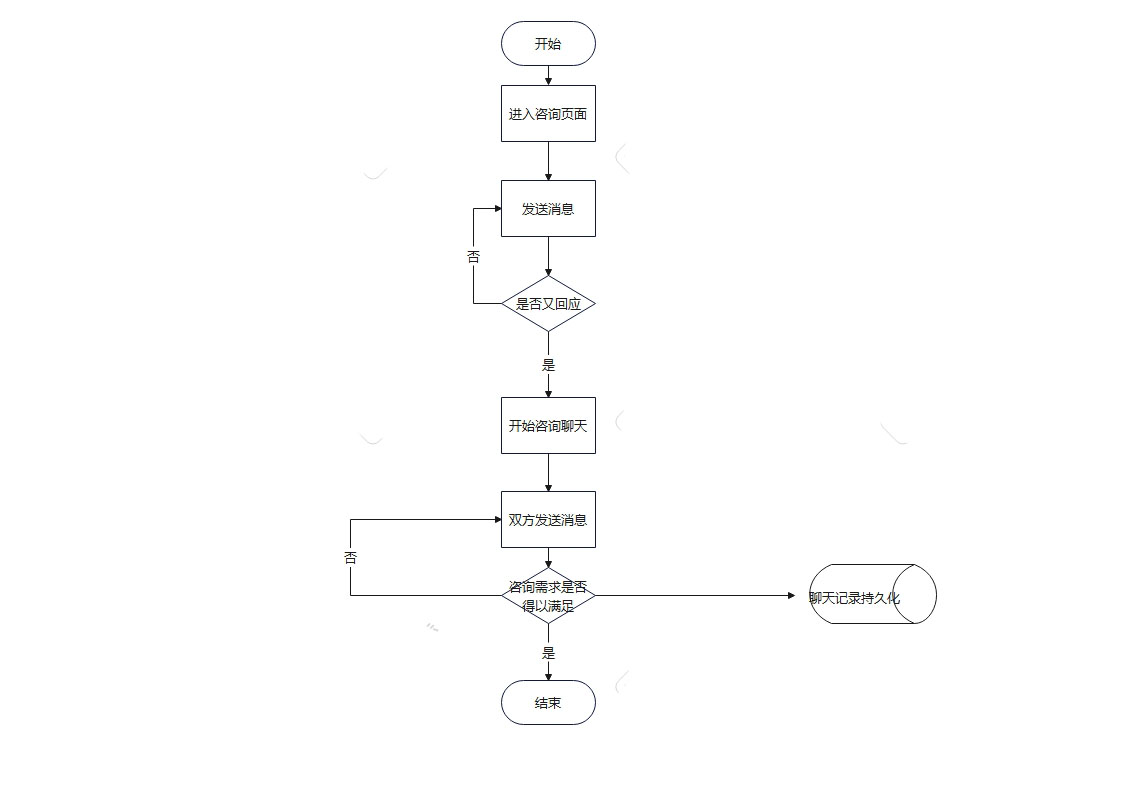


图3.18消息咨询程序流程图

### 3.7.6接口

输入：读者想发送的信息

数据来源：文本

数据格式：{管理员id，文本信息}

数量：1

输出：管理员收到信息

### 3.7.7存储分配

无

## 3.8便捷服务模块

### 3.8.1功能描述

这个模块是为了方便用户使用而创建的功能，包括了智能机器人聊天以及图书推荐等相关便捷性的服务功能

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **用户群** | **优先级** |
| **便捷服务** | **智能聊天机器人** | **用户可以与智能机器人进行聊天，了解推荐图书、今日图书以及其他图书馆相关信息** | **全体** | **中** |
| **图书推荐** | **系统通过机器学习，通过用户平日的借阅量搜索量等来进行图书推荐** | **全体** | **高** |

表3-8 便捷服务功能描述

### 3.8.2性能描述

在1000s内能准确机器人相应和推荐

### 3.8.3输入

用户个人id

### 3.8.4输出

推荐的图书书籍

### 3.8.5程序逻辑

模块程序流程图如图所示

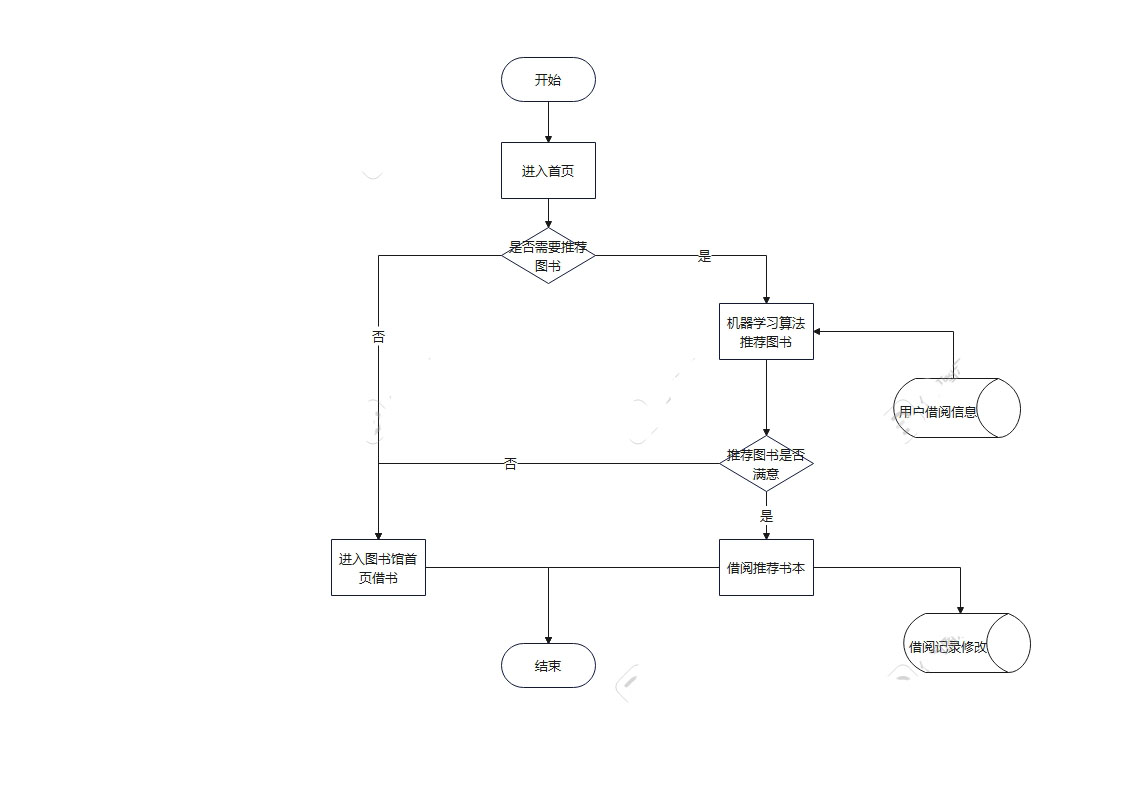


图3.19便捷服务程序流程图

### 3.8.6接口

**1）chatgpt需求**

输入：文本

数据来源：文本

数据格式：{文本}

数量：1

输出：gpt回复

**2）图书推荐需求**

输入：成员id

数据来源：redis数据库

数据格式：{成户id}

数量：1

输出：图书推荐

### 3.8.7存储分配

**无**