

LibrarySystem 系统设计说明书

团队名称: Team6

指导老师: 张栋

学校: 福州大学至诚学院

文档修订控制

序号	版本号	修订日期	修改概述	修订人	审批人	备注
1	1.0.0	2021/5/8	初步编写	徐鑫泽、江政	姜睿喆	

目录

一、引言

1.1 编写目的

1.2 背景

1.3 参考资料

1.4 术语

二、总体设计

2.1 系统架构

2.1.1 技术架构

2.1.2 应用架构

2.1.3 功能架构

2.2 基本设计概念和业务流程

2.3 外部接口

2.3.1 用户接口

三、模块设计

3.1 功能

3.1.1 简述各功能模块的功能

3.1.2 功能模块层次设计

3.1.3 功能模块层次图

3.1.4 设计类图

3.2 性能

3.2.1 用户信息精度（学生）

3.2.2 用户信息精度（教师）

3.2.3 图书信息精度

3.3 内部接口

3.3.1 登录验证接口

3.3.2 搜索栏判断接口

四、数据库设计

4.1 数据库模型

4.1.1 概念结构模型

4.1.2 逻辑结构模型

4.1.3 物理结构模型

4.2 ER 分析

4.3 表结构分析

4.4 安全保密设计

五、系统安全和权限设计

5.1 系统安全

5.1.1 系统安全控制

5.2 数据安全

5.2.1 数据用户身份鉴别

5.2.2 访问主体、访问对象的控制策略和实现方法

5.2.3 数据加密方法

5.3 备份与恢复

5.4 出错处理

5.4.1 列举出错情况

5.4.2 出错的形式、含义、处理方法

5.4.3 出错失效后的后备措施

5.5 权限设计

5.5.1 用户认证

5.5.2 用户授权

一、引言

1.1 编写目的

本设计说明书文档包括该项目的创建背景,术语以及参考,功能设计实现约束等的描述,说明书将全面介绍本款小程序的设计方案,指导本项目的开发与测验,同时,作为本项目的核心说明,为未来的升级打下基础。

本说明书的面向用户为本次系统开发小组的成员、软件工程老师等。

1.2 背景

图书管理是高校内每个院系都必须切实面对的工作,但目前仍有少部分院系仍在使用传统的人工方式管理图书资料。这种方式存在着许多缺点,如效率低、保密性差且较为繁琐。

另外,随着图书资料数量的增加,其工作量也将大大增加,这必将增加图书资料管理者的工作量和劳动强度,这将给图书资料信息的查找、更新和维护都带来了许多困难。

经过详细的调查,目前我国各类高等学校中仍有一小部分单位图书资料管理还停留在人工管理的基础上。这样的管理机制已经不能适应时代的发展,其管理方法将浪费许多人力和物力。随着科学技术的不断提高,这种传统的手工管理方法必然被以计算机为基础的信息管理方法所取代。

图书管理作为计算机应用的一个分支,有着手工管理无法比拟的优点,如检索迅速、查找方便、可靠性高、存储量大、保密性好、寿命长、成本低等。这些优点能够极大地提高图书管理的效率。

因此,开发一套能够为用户提供充足的信息和快捷的查询手段的图书管理系统,将是非常必要的,也是十分及时的。

1.3 参考资料

1) UML 之类图

<https://www.cnblogs.com/LangZXG/p/6208716.html>

2) UML 类图详解

<https://blog.csdn.net/loumoxiaozi/article/details/81036408>

3) 详解设计模式六大原则

<https://www.cnblogs.com/toutou/p/4870926.html>

4) 数据库表结构设计的几条准则

<https://www.cnblogs.com/wyq178/p/8549715.html>

5) 数据库表设计（一对多、多对多）

<https://blog.csdn.net/fighteryang/article/details/82848505>

6) 常见 web 攻击方法及防御手段总结

<https://blog.csdn.net/qappleh/article/details/80485197>

7) 用户密码加密

https://blog.csdn.net/weixin_34221073/article/details/89652071

8) JavaWeb 项目前后端分离

<https://www.cnblogs.com/donggedada/p/11377062.html>

9) RBAC 用户、角色、权限、组设计方案

<https://blog.csdn.net/ljw499356212/article/details/81055141>

1.4 术语

表 1-1 缩写术语表

序号	缩写	定义
1	APP	应用程序，Application 的缩写，一般指手机软件。
2	Android	Android 是一种基于 Linux 的自由及开放源代码的操作系统，主要使用于移动设备，如智能手机和平板电脑，由 Google 公司和开放手机联盟领导及开发。
3	WeUI	WeUI 是一套同微信原生视觉体验一致的基础样式库，由微信官方设计团队为微信内网页和微信小程序量身设

		计，令用户的使用感知更加统一。
4	WXML	WeiXin Mark Language，微信标记语言。
5	WXSS	WeiXin Style Sheet，微信样式表。
6	Linux	Linux，全称 GNU/Linux，是一套免费使用和自由传播的类 UNIX 操作系统。

二、总体设计

2.1 系统架构

2.1.1 技术架构

小程序客户端采用 C/S（客户端—服务器）架构，管理端采用 B/S 架构（浏览器—服务器）。

2.1.2 应用架构

根据前后端交互及应用部署的环境结构，画出系统应用架构图。

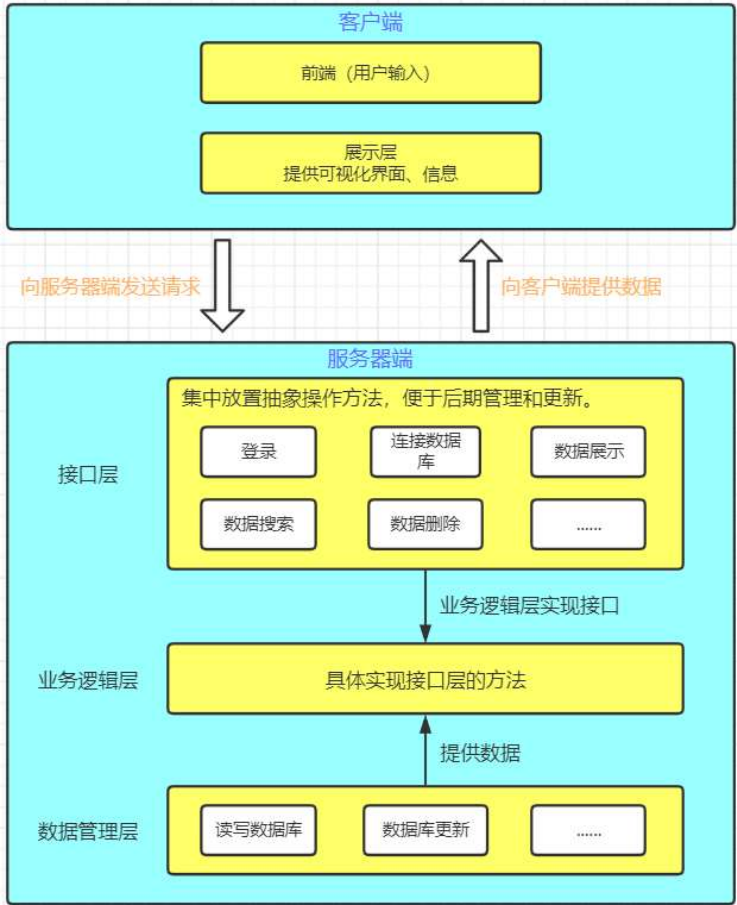


图 2-1 系统架构图

2.1.3 功能架构

通过用例图展示系统的功能架构

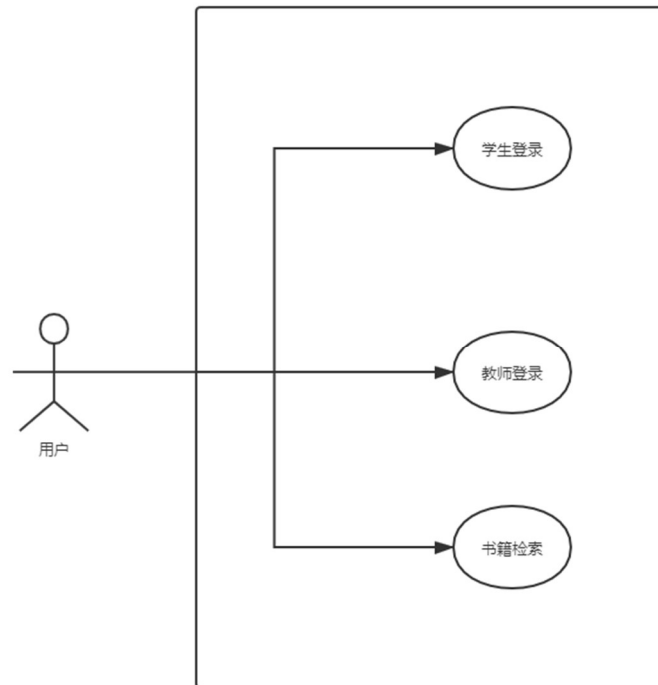


图 2-2 用例图

2.2 基本设计概念和业务流程

小组设计的活动图更直观的展示出图书馆程序的基本设计和一整套业务流程。

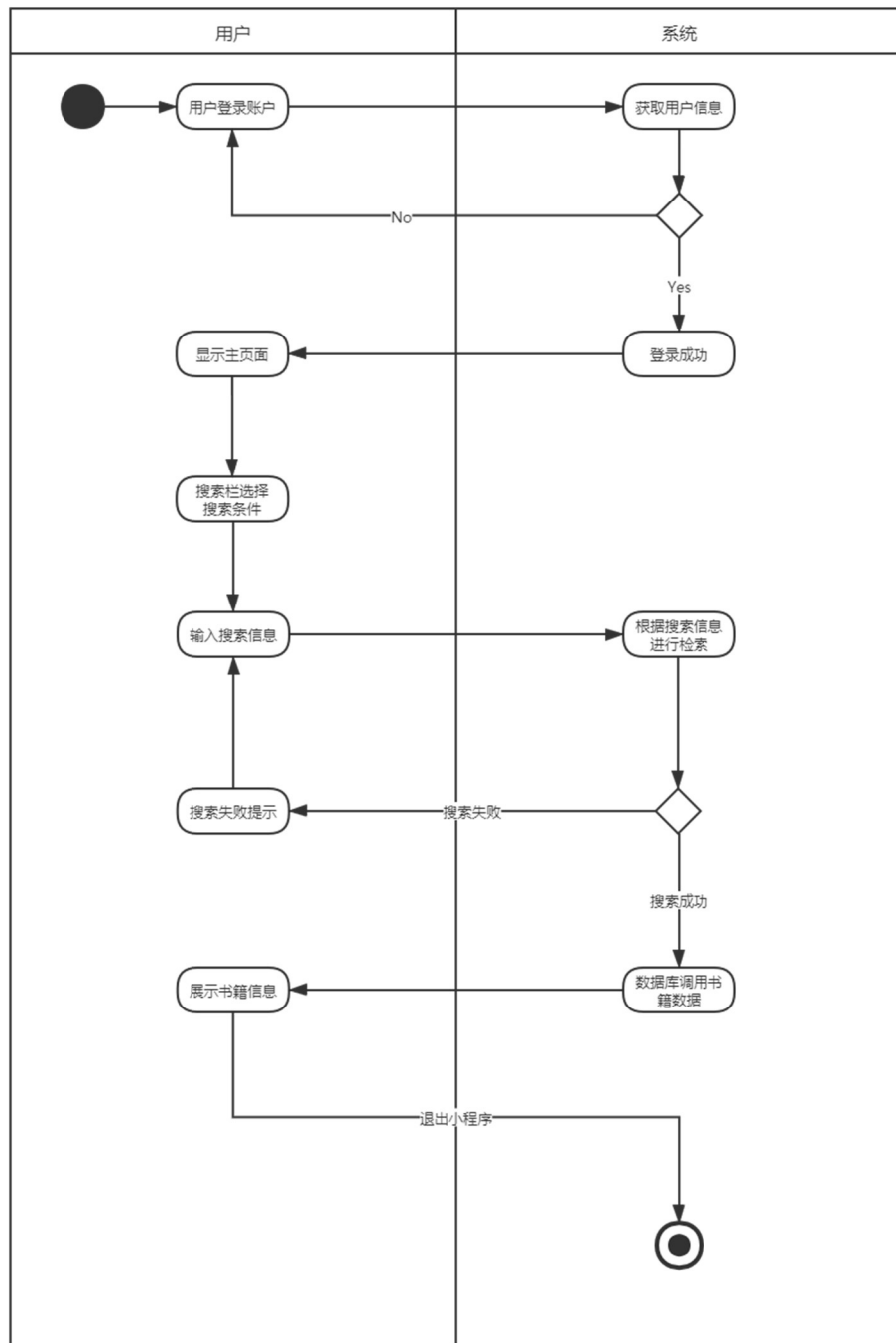


图 2-3 活动图

2.3 外部接口

2.3.1 用户接口

图书馆程序的原型设计

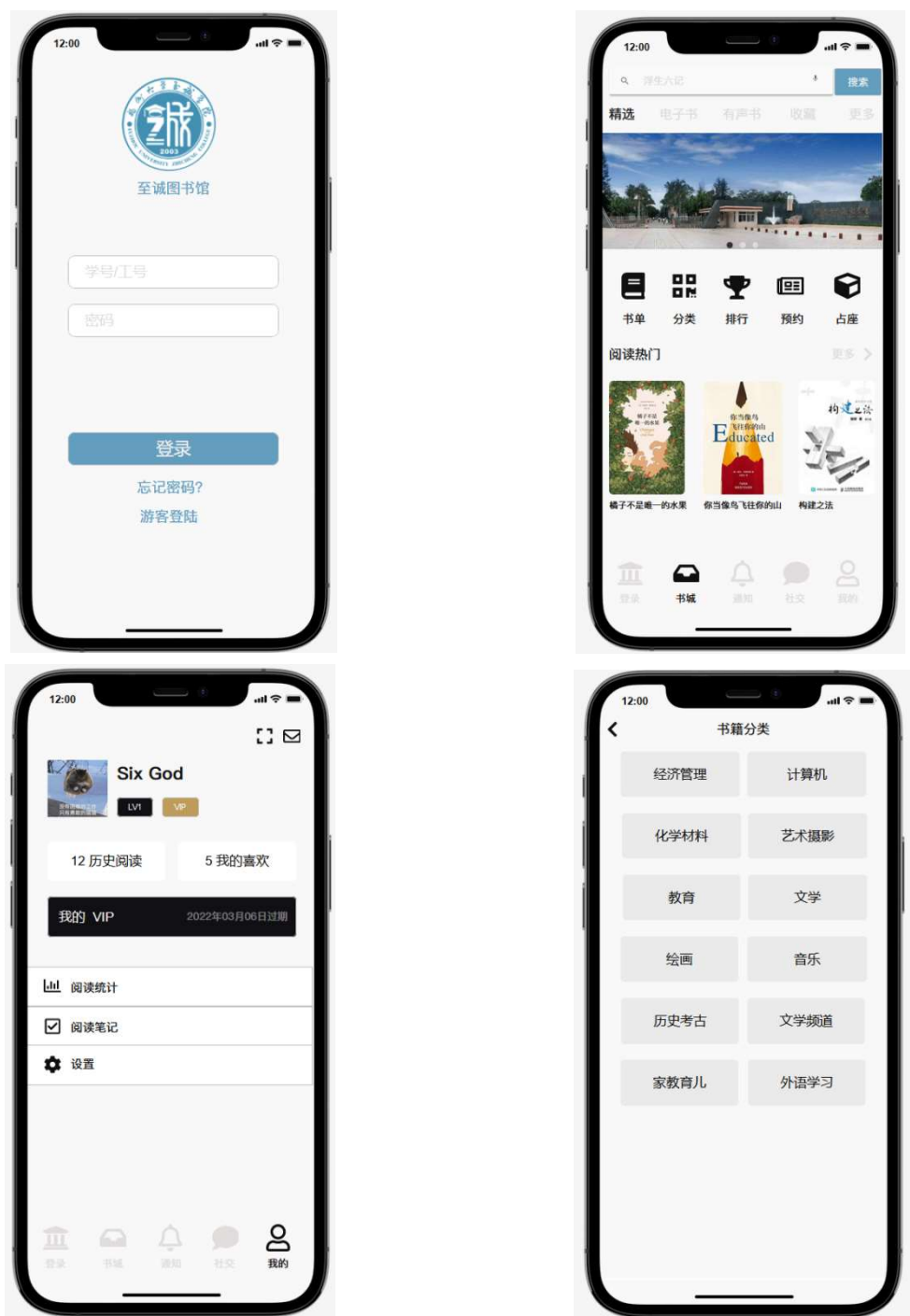


图 2-4 原型设计

三、模块设计

3.1 功能

3.1.1 简述各功能模块的功能

（1）图书搜索功能

通过小程序可以便捷的查找所需要的图书，如通过 ISBN 码进行精确查找，或是通过关键字进行模糊查找，查找结果显示图书所在楼层、书架位置，方便同学们快速找到书籍进行借阅。

（2）图书分类功能

通过分类界面将分配归类的书本，展现给用户。

（3）用户登录功能

通过小程序页面进行登录，根据用户输入的学号/工号和密码与数据库进行匹配，匹配成功即可登录查看并使用小程序其他功能。

3.1.2 功能模块层次设计

图书馆系统大体分为两类：

一类为用户登录，分为学生和教师登录；另一类为图书搜索通过搜索栏的关键字搜索，搜索到书籍显示结果，若没有则返回搜索错误。

3.1.3 功能模块层次图

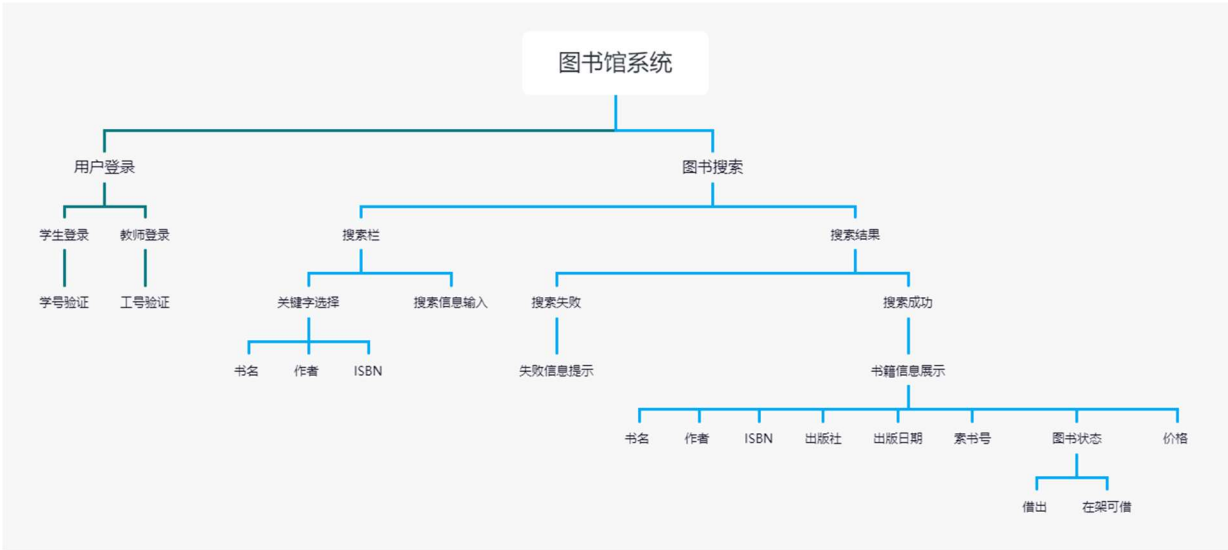


图 3-1 功能层次图

3.1.4 设计类图

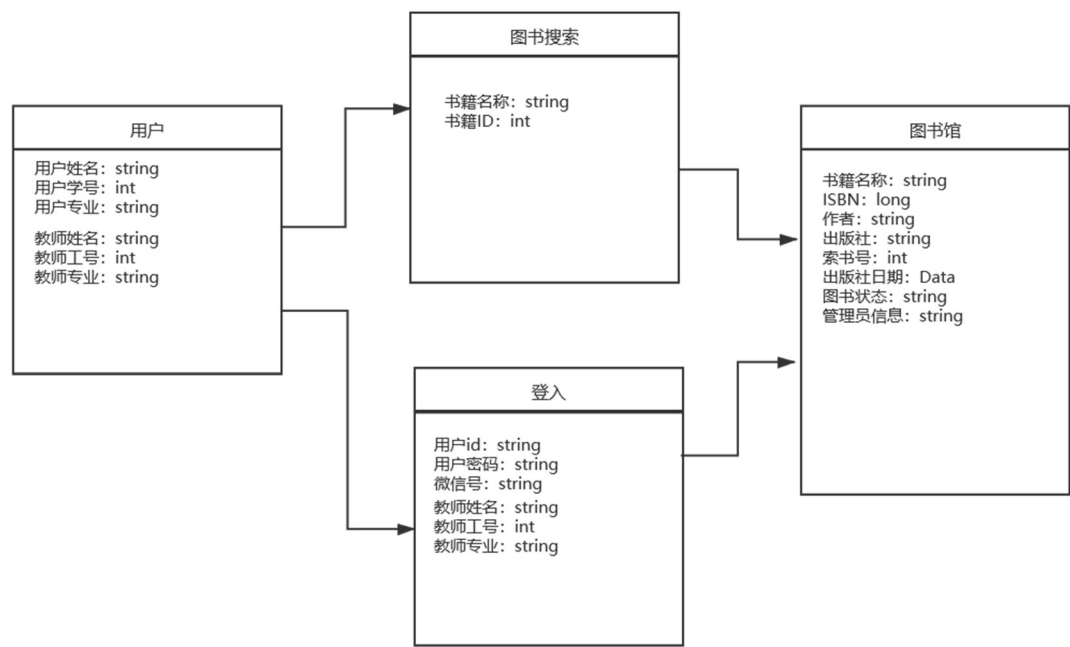


图 3-2 类图

3.2 性能

3.2.1 用户信息精度（学生）

- 用户名：9 位数字（学号），不可为空。
- 用户密码：密码由 9-20 个字符组成。密码只能由数字、英文字符、下划线组成。不可为空，初始密码同学号。
- 微信号：初始为空，根据用户登录后进行一对一绑定。

3.2.2 用户信息精度（教师）

- 用户名：9 位数字（工号），不可为空。
- 用户密码：密码由 9-20 个字符组成。密码只能由数字、英文字符、下划线组成。不可为空，初始密码同工号。
- 微信号：初始为空，根据用户登录后进行一对一绑定。

3.2.3 图书信息精度

- 序号：Int 整形，自动生成。
- 书名：String 类型，图书馆数据导入。
- 作者：String 类型，图书馆数据导入。
- 出版社：String 类型，图书馆数据导入。
- 出版日期：Date 类型，图书馆数据导入。
- 索书号：String 类型，图书馆数据导入。
- 馆藏数：Int 类型，根据数据库内的数据实时改变。
- 可借数：Int 类型，根据数据库内的数据实时改变。
- 价格：Float 类型，图书馆数据导入。

3.3 内部接口

3.3.1 登录验证接口

在验证登录时，调用该接口，传入的参数为用户唯一标识 UserId 和 PassWd 会话维持。

表 3-1

参数名	类型	长度	备注
UserId	Int	9 字节	用户名
PassWd	String	20 字节	密码

如果用户存在且密码正确返回 True，如果用户存在但密码不正确返回 False。

3.3.2 搜索栏判断接口

在搜索栏中调用搜索栏接口，传入参数为用户输入 Search。

表 3-2

参数名	类型	长度	备注
Search	String	255 字节	搜索内容
Kind	String	255 字节	检索类型

根据 Kind 选择检索类型，Search 根据数据库中的信息进行比较，若在数据库中找到对应书本信息则反馈至页面，若没有找到则返回 None。

四、数据库设计

4.1 数据库模型

4.1.1 概念结构模型

学生信息(用户名, 微信号, 用户密码, 是否删除)

其中属性约束如下: 是否删除: 0=否, 1=是

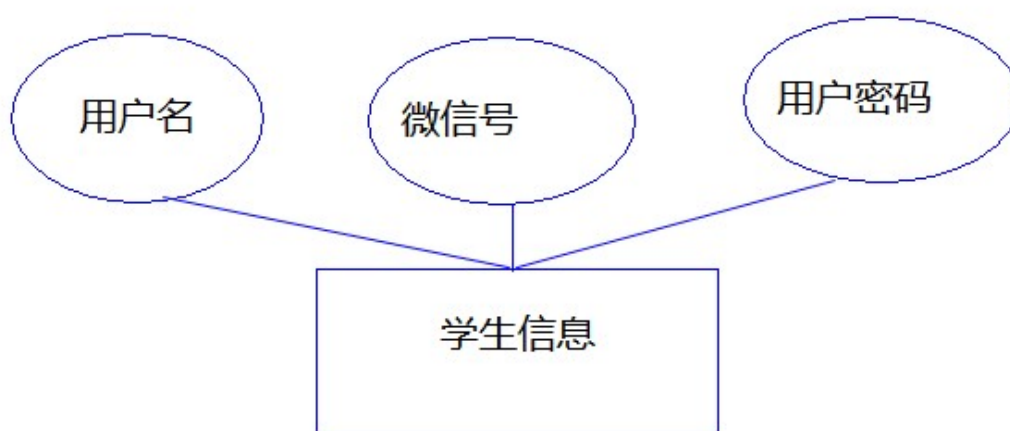


图 4-1 学生信息属性约束

老师信息(用户名, 微信号, 用户密码, 是否删除)

其中属性约束如下: 是否删除: 0=否, 1=是

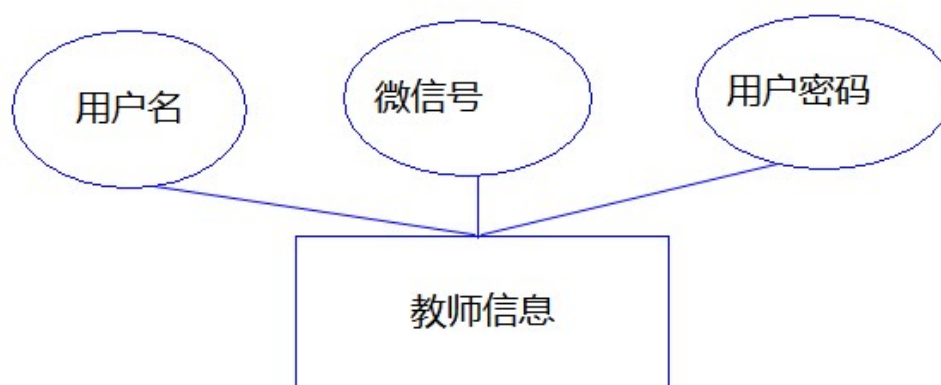


图 4-2 教师信息属性约束

图书信息(序号, 书名, 作者, 价格, 可借数, 馆藏数, 索书号, 出版日期, 出版社, 是否删除)

其中属性约束如下：是否删除：0=否，1=是

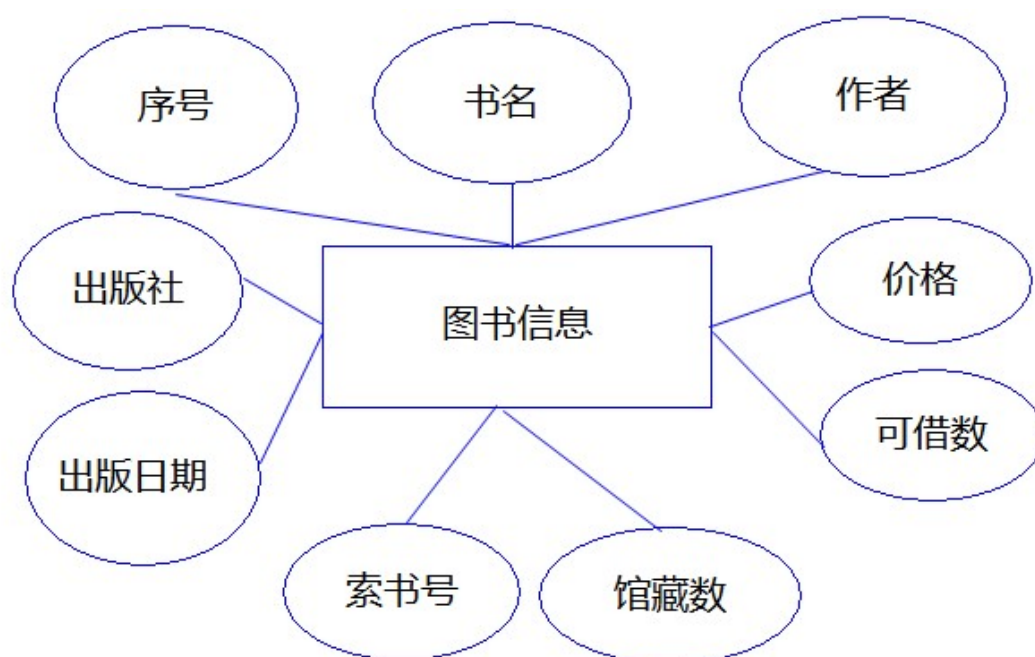


图 4-3 图书信息属性约束

4.1.2 逻辑结构模型

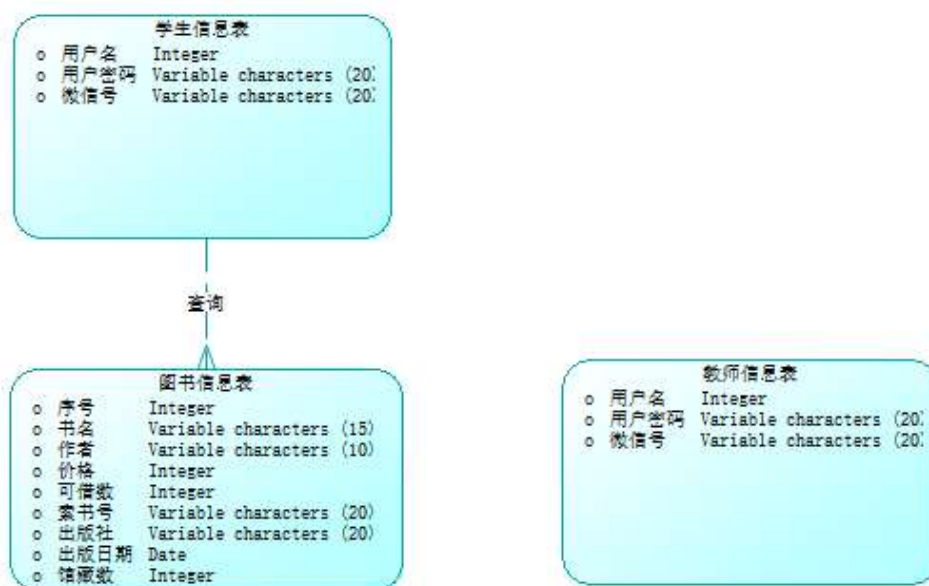


图 4-4 逻辑结构模型

4.1.3 物理结构模型

[illegible]

图 4-5 学生信息表

[illegible]

图 4-6 教师信息表

[illegible]

图 4-7 图书信息表

4.2 ER 分析

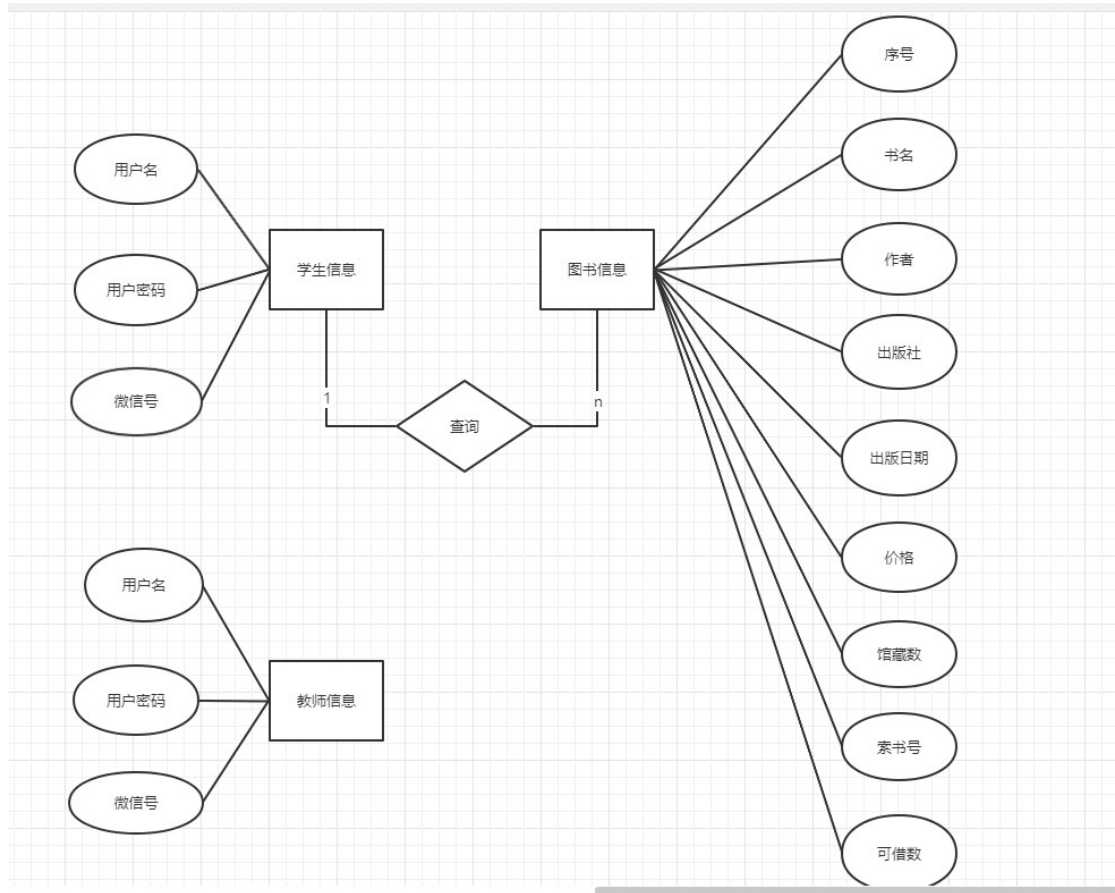


图 4-8 ER 分析图

4.3 表结构分析

[illegible]

图 4-9 学生信息表记录学生登陆账号和微信号

[illegible]

图 4-10 教师信息表记录学生登陆账号和微信号

[illegible]

图 4-11 图书信息表记录图书信息

4.4 安全保密设计

通过区分不同的访问者和不同的数据对象,进行分别对待而获得的数据库安全保密设计考虑。数据库由专门数据库管理员对数据库操作。系统管理员权限最大,可以控制所有数据,对已有数据进行增删改查,以及功能的增减。学生和教师目前权限仅开放登录功能和查询功能。

五、系统安全和权限设计

5.1 系统安全

5.1.1 系统安全控制

（1）ip 段的限制：

在程序中设置限制，当出现大批量某一字段 ip 访问部署在服务器上的域名时，为了防止被攻击，对这种类似 ip 进行禁止访问。

（2）登录时间段限制：

由于图书馆有开馆时间限制，为了和图书馆不存在矛盾，登录时间段限制在早上八点到晚上九点。

5.2 数据安全

5.2.1 数据用户身份鉴别

学生学号与教师工号采取一号一人的准则，通过学号与工号可有效鉴别用户的身份

5.2.2 访问主体、访问对象的控制策略和实现方法

采用基于角色的访问控制模型，即 RBAC 模型，对用户进行划分角色：管理员、学生用户，不同角色拥有不同的访问许可权。为了更好地管理用户，对用户进行分组归类，简称为用户分组。用户可以自己拥有权限信息，可以归属于 0~n 个角色，可属于 0~n 个组。权限集是自身具有的权限、所属的各角色具有的权限、所属的各组具有的权限的合集。

它与权限、角色、组之间的关系都是 n 对 n 的关系。

5.2.3 数据加密方法

小程序需要进行保护的数据主要是所填写的学生信息和教师信息，可通过密文的方法对输入的数据进行加密传输。

5.3 备份与恢复

通过建立两台以上的备用数据库或将数据上传至云服务器，当数据库因受到攻击或某些操作对数据库造成不可逆损坏时，可从备用数据库或云服务器进行恢复，同时多台云服务器或数据库也降低了数据库或云服务器同时宕机的概率。

5.4 出错处理

5.4.1 列举出错情况

用户的用户名或者密码输入错误

5.4.2 出错的形式、含义、处理方法

表 5-1 出错处理

错误类型	范围	处理方式
操作错误	用户进行交互时不满足输入规则、输入范围等发生的错误	校验用户输入 提示正确规则 强制其改正
运行错误	与外部资源交互发生错误，例如：网络错误、数据库错误等	记录并抛出异常 界面提示 记录错误日志信息

5.4.3 出错失效后的后备措施

用户通过绑定的微信号或联系管理员对用户的密码进行修改或初始化。

5.5 权限设计

5.5.1 用户认证

使用小程序的主体在进行身份认证时，需要提供用户信息。

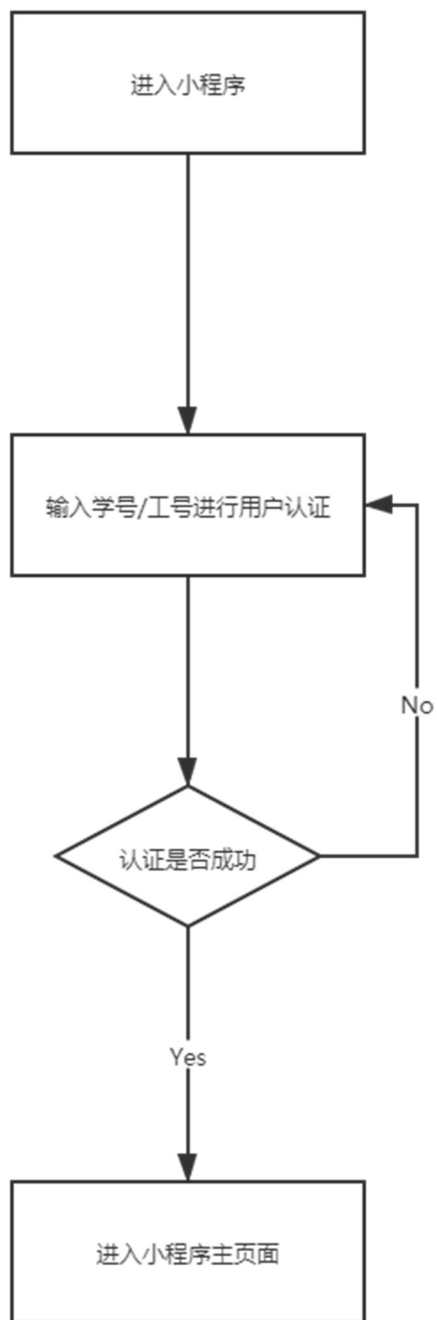


图 5-1 用户认证环节

5.5.2 用户授权

给不同用户分配不同的角色，不同的角色授权不同的资源，不同的用户就有不同的权限。

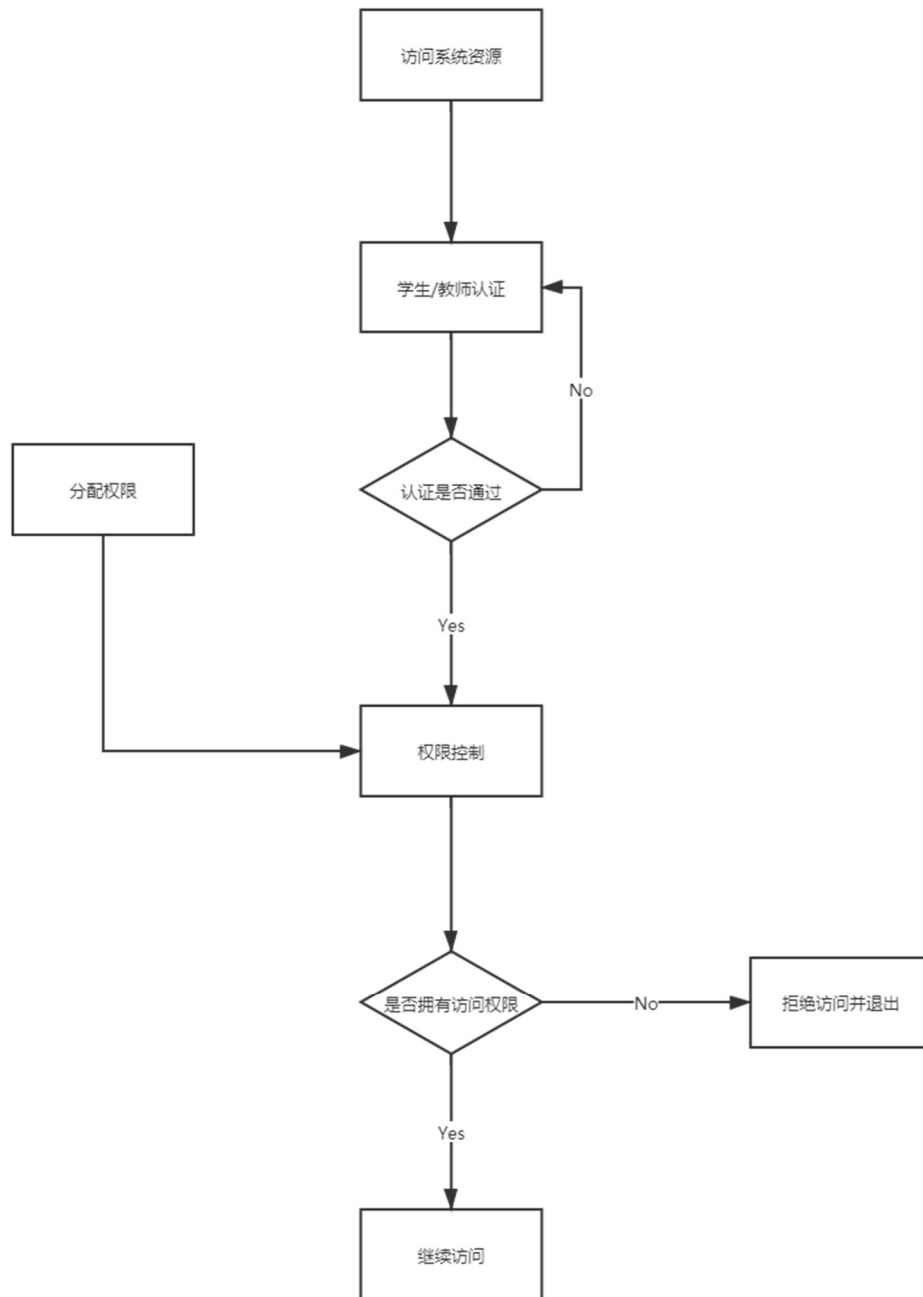


图 5-2 用户授权