

四川大學

本 科 生 课 程 设 计



题 目 某书店书刊出租和零售管理系统

课 程 名 数据库系统与信息管理

任课教师 吴成恩

学 院 软件学院

专 业 软件工程

学生姓名 陈枕熙

学 号 2018141463009 年级 2018

目录

一、	系统介绍	4
1.1	系统简介	4
1.2	功能性需求	4
1.2.1	存储书刊信息	4
1.2.2	查询商品库存	4
1.2.3	顾客信息登记	4
1.2.4	零售信息记录	4
1.2.5	库存管理功能	4
1.2.6	零售信息查询	5
1.2.7	借阅信息记录	5
1.2.8	归还信息记录	5
1.2.9	借阅信息查询	5
1.2.10	图书销售情况查询	5
二、	数据库相关	5
2.1	数据需求	5
2.2	关系模型	6
2.2.1	book 表	6
2.2.2	customer 表	7
2.2.3	operator 表	7
2.2.4	order 表	7
2.2.5	rent 表	7
2.2.6	sell 表	8
2.3	函数依赖	8
2.3.1	book 表	8
2.3.2	customer 表	8
2.3.3	operator 表	9
2.3.4	order 表	9
2.3.5	rent 表	9
2.3.6	sell 表	9
三、	系统开发	9
3.1	系统结构介绍	9
3.2	系统开发环境介绍	10
3.3	典型模块的实现	10

3.3.1 DBMS 操作	10
3.3.2 登录/注册模块.....	11
3.3.3 查询模块.....	11
3.3.4 记录模块.....	11
四、 系统实现	12
4.1 系统使用	12
4.2 系统界面截图.....	12
4.2.1 登录界面.....	12
4.2.2 注册界面.....	13
4.2.3 库存查询界面.....	13
4.2.4 添加库存界面.....	14
4.2.5 零售查询界面.....	14
4.2.6 零售记录界面.....	15
4.2.7 借阅查询界面.....	15
4.2.8 记录借出界面.....	16
4.2.9 记录归还界面.....	16
4.2.10 顾客注册界面.....	17
4.2.11 书本销量查询	17

一、系统介绍

1.1 系统简介

本次课程设计我所选的题目是《某书店书刊出租和零售管理系统》，该系统实现了对客户信息、书刊信息的管理，以及借阅、归还、零售等行为的记录与管理。本系统采用了 B/S 架构，采取前后端分离的开发方式，前端主要采用 Vue.js 作为开发框架，而后端则采用 Django 作为 Web 框架，数据库方面，本项目采用了 MySQL 作用数据库管理系统。

1.2 功能性需求

1.2.1 存储书刊信息

该系统需要提供存储书刊信息的功能，应该存储的书刊信息主要有，书刊的 ISBN、书名、作者、售价、类别。

1.2.2 查询商品库存

该系统需要提供商品库存查询功能，显示每种书籍与刊物的库存情况。

1.2.3 顾客信息登记

该系统需要为新顾客登记信息，诸如其 ID、姓名、住址等。

1.2.4 零售信息记录

该系统需要记录书店的零售信息，主要包括每个订单的订单号、交易时间、工作人员、顾客、金额以及所购买的图书。

1.2.5 库存管理功能

该系统需要提供对库存进行管理的功能，主要为商品进货，需要为现有商品添加库存。

1.2.6 零售信息查询

该系统需要提供零售信息记录的查询功能，并支持根据客户姓名来搜索交易记录。

1.2.7 借阅信息记录

该系统需要提供客户借阅图书的相关记录，包括借阅图书的客户信息、借阅的时间、应还的时间以及登记的工作人员信息。

1.2.8 归还信息记录

当客户归还所借阅的图书或期刊后，该系统应该支持工作人员标记用户归还了这本图书，并记录用户归还书刊的时间。

1.2.9 借阅信息查询

该系统需要提供给工作人员查询图书借阅信息的能力，应支持按照顾客姓名进行搜索。

1.2.10 图书销售情况查询

该系统需要提供图书在一段时间内的销售情况以及借阅情况，以便管理人员判断是否需要增加进货量。应支持按照指定时间进行统计。

二、数据库相关

2.1 数据需求

该项目的 E-R 图如下图所示：

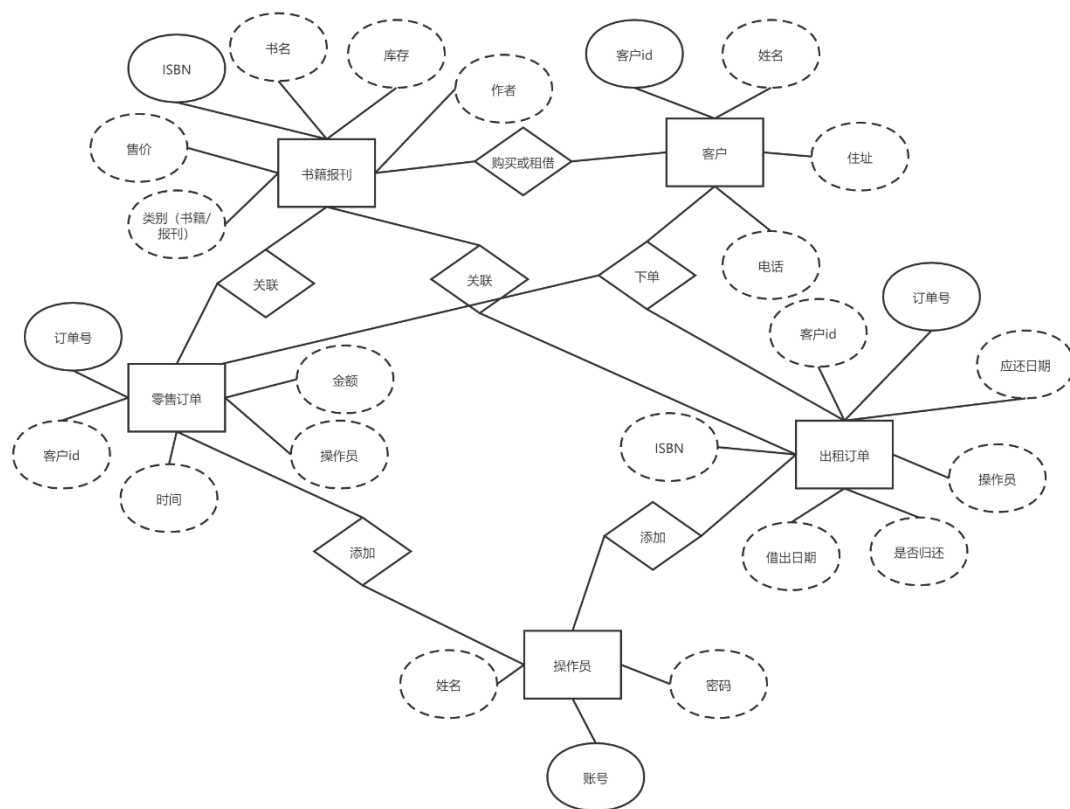


图 2-1 E-R 图

2.2 关系模型

2.2.1 book 表

本项目通过 book 表来记录书刊的基本信息以及库存、售价等信息。

	列名	类型	长度	备注	可空	主键
1	ISBN	varchar	20	书籍的 ISBN		√
2	title	varchar	30	书名		
3	author	varchar	30	作者		
4	type	tinyint	1	类别		
5	number	int	11	库存		
6	price	float		价格		

2.2.2 customer 表

本项目通过 customer 表来记录各个注册客户的基本信息。

	列名	类型	长度	备注	可空	主键
1	cid	int	11	客户 ID		√
2	name	varchar	12	客户姓名		
3	address	varchar	50	客户地址	√	

2.2.3 operator 表

本项目通过 operator 表来记录工作人员账号的相关信息。

	列名	类型	长度	备注	可空	主键
1	username	varchar	12	操作员 ID		√
2	password	varchar	30	密码		
3	name	varchar	10	操作员姓名		

2.2.4 order 表

本项目通过 order 表来记录书刊零售订单，该表格仅记录订单基本信息，每个订单具体交易了哪些书将通过 sell 表来记录

	列名	类型	长度	备注	可空	主键
1	OrderID	varchar	20	订单号		√
2	OperatorID	varchar	12	操作员 ID		
3	amount	float		订单金额		
4	cid	int	11	顾客 ID		
5	time	datetime		交易时间		

2.2.5 rent 表

本项目通过 rent 表来记录书刊借阅的相关情况。

	列名	类型	长度	备注	可空	主键
1	OrderID	varchar	20	订单号		√
2	OperatorID	varchar	12	操作员 ID		
3	ISBN	varchar	20	书本 ISBN		

4	cid	int	11	顾客 ID		
5	rent_time	datetime		借出时间		
6	due_date	date		应还时间		
7	Return_date	date		归还时间	√	

2.2.6 sell 表

本项目通过 sell 表来记录不同零售订单具体销售了哪些书刊。一个订单号可以对应多个不同的 ISBN，因此，我们需要另外新建一个自增列来作为主键。

	列名	类型	长度	备注	可空	主键
1	key	int	255	自增以充当主键		√
2	OrderID	varchar	20	订单号		
3	ISBN	varchar	20	书本 ISBN		

2.3 函数依赖

在上一节中，列出了本项目的数据库表结构。而这一节中，我将对这些表结构的函数依赖进行分析，并分析这些表结构的设计是否符合 3NF 的要求。

2.3.1 book 表

首先，我们给出 book 表的函数依赖如下：

ISBN→author, title, type, number, price

由于 ISBN 是 book 表的主键，由此我们不难看出，book 表显然符合 3NF 的要求。

2.3.2 customer 表

接下来我们对 customer 表进行分析，customer 表的函数依赖如下：

cid→name, address

由于 cid 是 customer 表的主键，因此，customer 表也符合 3NF 的要求。

2.3.3 operator 表

接下来我们对 operator 表进行分析，operator 表的函数依赖如下：

$username \rightarrow password, name$

username 是 operator 表的主键，因此 operator 表也符合 3NF 的要求。

2.3.4 order 表

接下来我们对 order 表进行分析，order 表的函数依赖如下：

$OrderID \rightarrow OperatorID, amount, cid, time$

OrderID 是 order 表的主键，故 order 表也符合 3NF 的要求。

2.3.5 rent 表

我们给出 rent 表的函数依赖如下：

$OrderID \rightarrow OperatorID, ISBN, rent_time, due_date, return_date$

OrderID 是 rent 表的主键，因此 rent 表也符合 3NF 的要求。

2.3.6 sell 表

我们给出 sell 表的函数依赖如下：

$key \rightarrow OrderID, ISBN$

由于 key 是 sell 表的主键，因此 sell 表也符合 3NF 的要求。

三、系统开发

3.1 系统结构介绍

该系统采用的是 B/S 架构，即浏览器/服务器架构，其中，前端部分通过 Vue.js 框架进行开发，而后端部分通过 Django 框架进行开发，服务器的搭建方面则选用了 Nginx 和 UWSGI 进行配合。

虽然本项目所采用的后端 Web 框架 Django 是 MVC 框架，但是本项目的开发并未采用 MVC 架构，而是采用了前后端分离的开发方式。后端通过 Django 框架实现 API 接口，前端页面由 Vue.js 独立完成，而并未使用 Django 的模板功能，

跨域问题则通过 Nginx 的反向路由来解决。这样子进行开发的好处在于可以使得开发过程中发生的错误被迅速定位，前后端之间不会产生相互影响。

该项目的文件结构如下图所示：

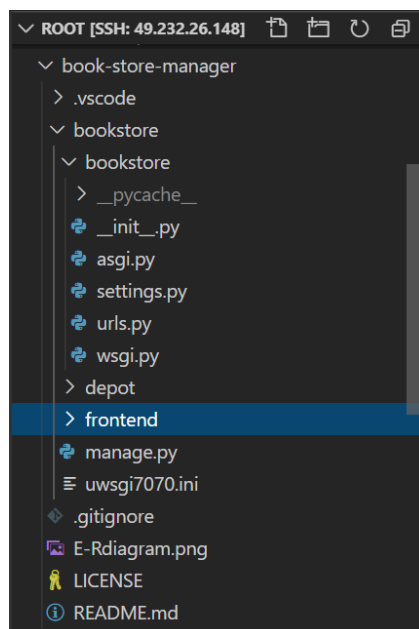


图 3-1 项目文件结构

其中，depot 目录下的内容主要用于实现后端 API 接口，而 frontend 目录下的文件则为前端的源代码。

3.2 系统开发环境介绍

本系统是 B/S 结构的系统，在开发过程中所使用的开发工具是 Microsoft Visual Studio Code，通过 Visual Studio Code 提供的远程操作功能直接对服务器进行操作，在服务端完成了开发过程。前端开发采用了 Vue.js 的框架，后端开发采用 Django 框架，DBMS 则选用了 MySQL。

3.3 典型模块的实现

3.3.1 DBMS 操作

本项目与 DBMS 的连接采用的是 Django 提供的 models 功能，通过 models 来实现对数据库的连接以及开发过程中所需要进行的增删查改操作。

3.3.2 登录/注册模块

首先通过访问数据库，查找用户输入的用户名是否存在。可通过以下 SQL 指令实现：

```
SELECT * FROM operator
WHERE username = 'XXX'
```

转换为 models 的语句如下：

```
models.Operator.objects.filter(username= username)
```

如果数据库返回的 Query Set 长度为 0，则说明不存在该用户名。那么登录界面就应该返回错误，而注册界面则应该进入下一步。反之，则登录模块进入下一步运算，注册界面返回错误。

接下来，如果登陆模块进入了下一步运算，则将返回的 Query Set 中的密码与用户输入的密码比较，若相同，则登陆成功，否则登陆失败。

若注册模块进入了下一步运算，则将用户输入的其他信息都录入到数据库当中。

3.3.3 查询模块

该项目提供了查询库存、查询零售记录以及查询借阅记录三个查询模块，其实实现方式相差不多，这里只选取查询库存模块进行说明。

查询库存模块实际上即为通过数据库获取 book 表中的库存信息，对应的 SQL 语句如下：

```
SELECT * FROM book
```

转换为 models 的语句如下：

```
models.Book.objects.all()
```

3.3.4 记录模块

对应于查询库存、查询零售记录和查询借阅三个查询模块，本项目也提供了对应的三个记录模块，即记录进货、记录零售以及记录借阅。这里采用记录零售模块。

首先，我们检测前端传来的数据是否正确，即需要检测顾客 ID 等信息是否存在，若前端信息错误，则返回异常，否则进入下一环节。

接下来，我们需要分别向 order 表和 sell 表插入数据，这里只列出对 order 表

的操作，所使用的 models 语句如下：

```
models.Order.objects.create(orderid=orderid, operatorid= oid, amount=amount,  
                             cid=cid, time=time)
```

同时，我们需要修改库存信息，所采用的 models 语句如下：

```
book.update(number= number)
```

四、系统实现

4.1 系统使用

该项目已经通过腾讯云服务器进行了配置，可以通过访问以下地址进行使用。

<http://49.232.26.148:7007/>

账号可使用测试账号（fy1666, 123），也可以进入注册界面自行注册。

4.2 系统界面截图

4.2.1 登录界面

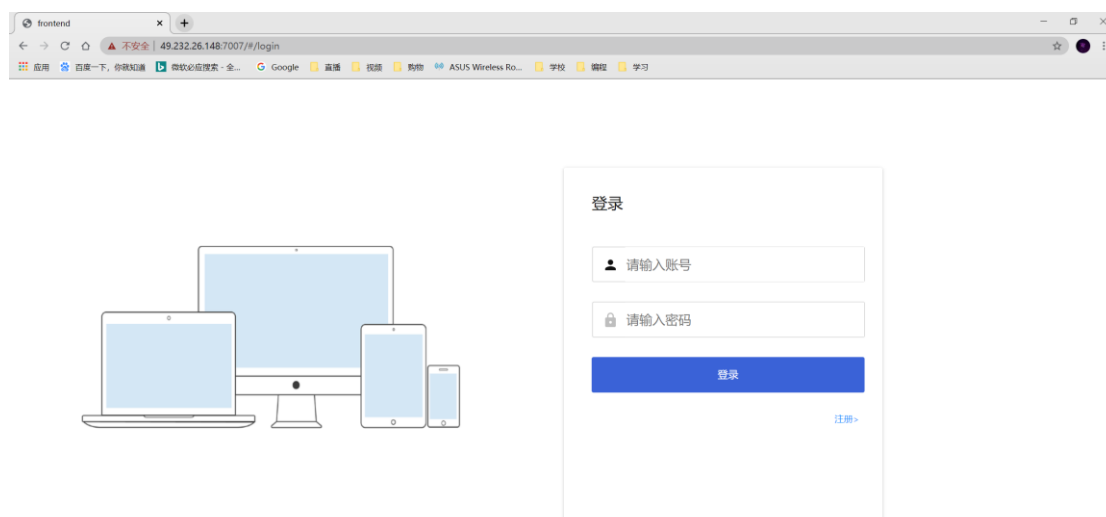


图 4-1 登录界面

4.2.2 注册界面

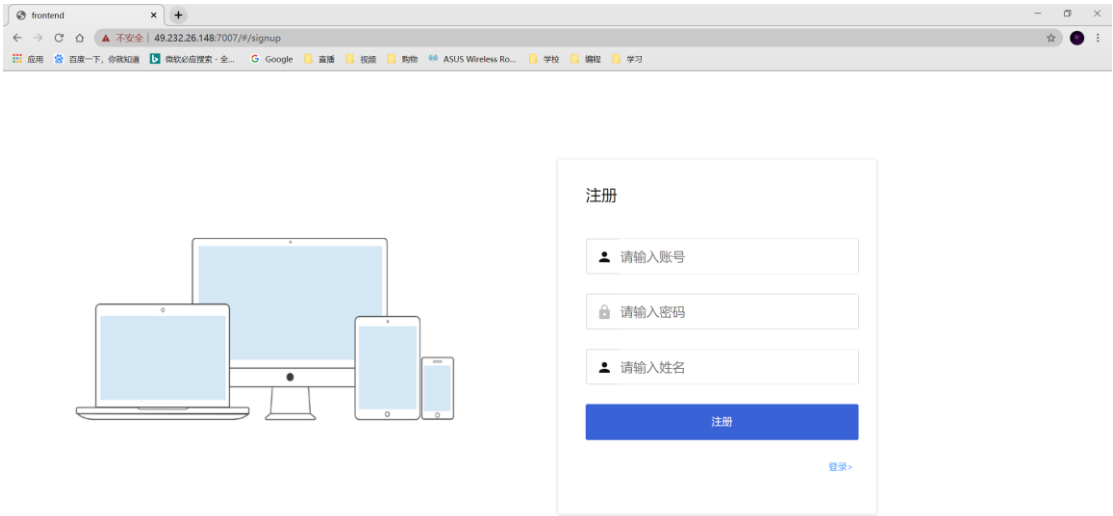


图 4-2 注册界面

4.2.3 库存查询界面

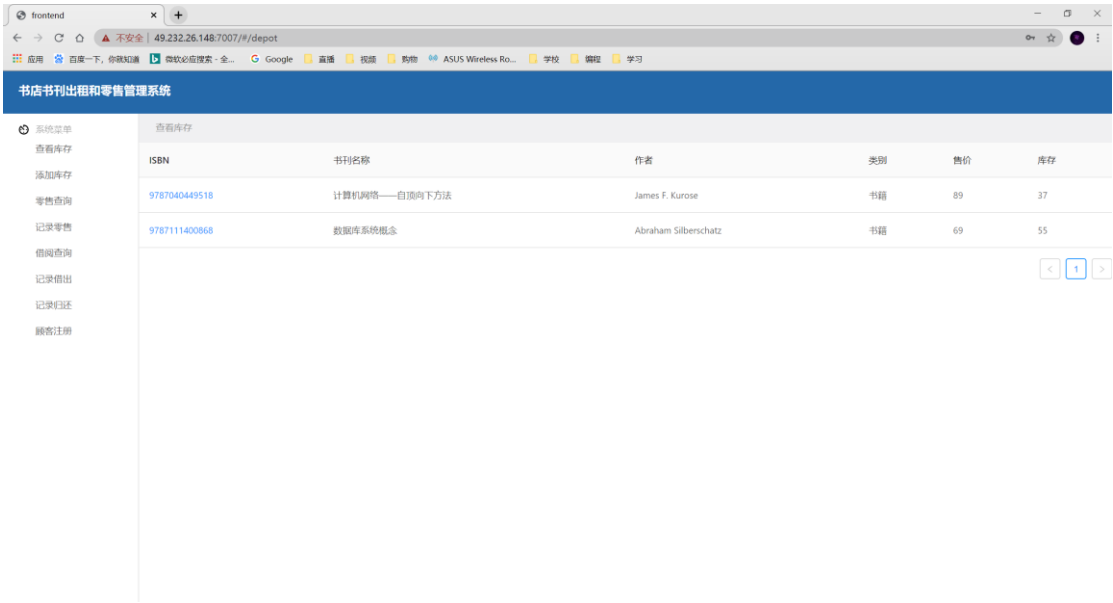
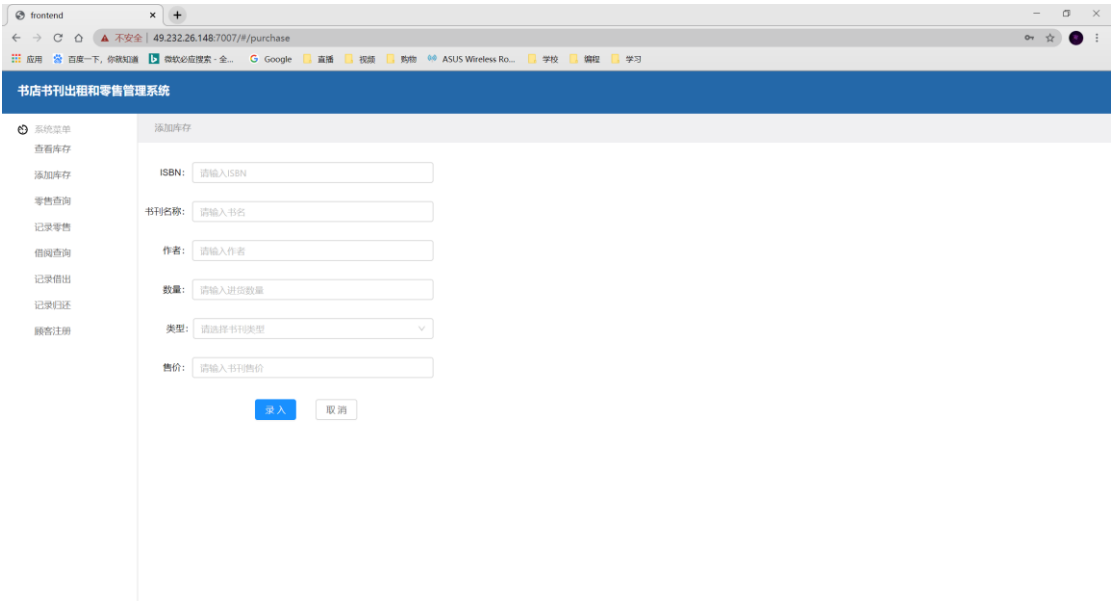


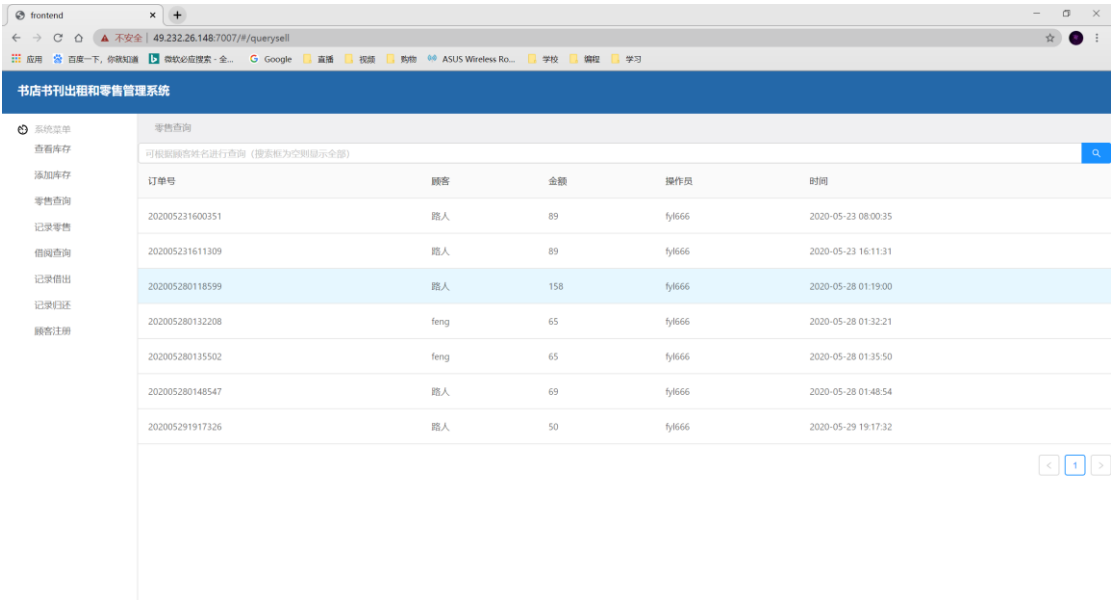
图 4-3 库存查询界面

4.2.4 添加库存界面



4-4 添加库存界面

4.2.5 零售查询界面



4-5 零售查询界面

4.2.6 零售记录界面

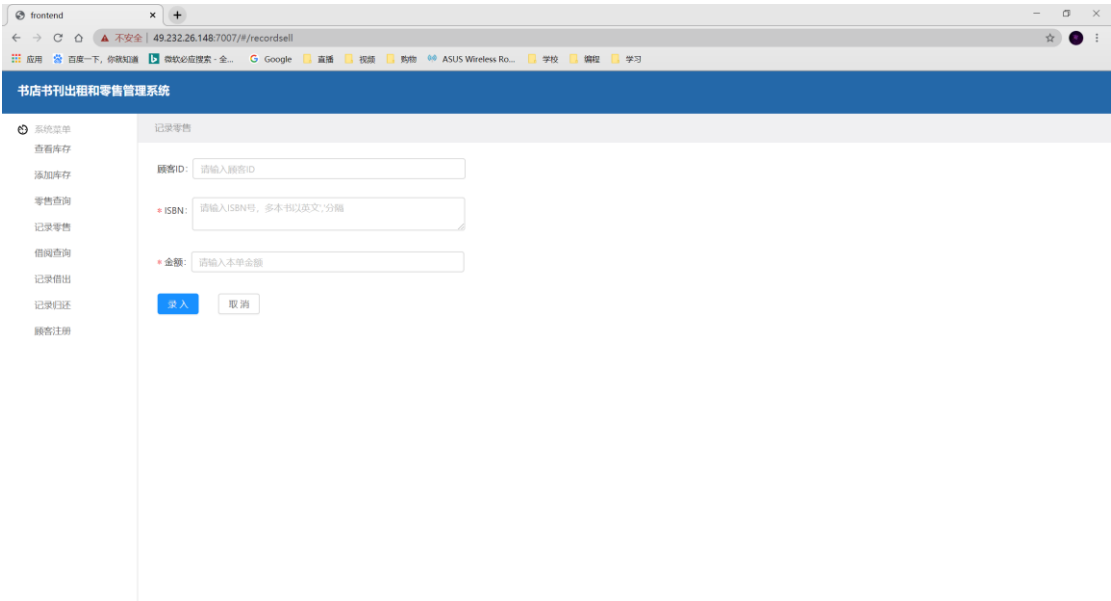


图 4-6 零售记录界面

4.2.7 借阅查询界面

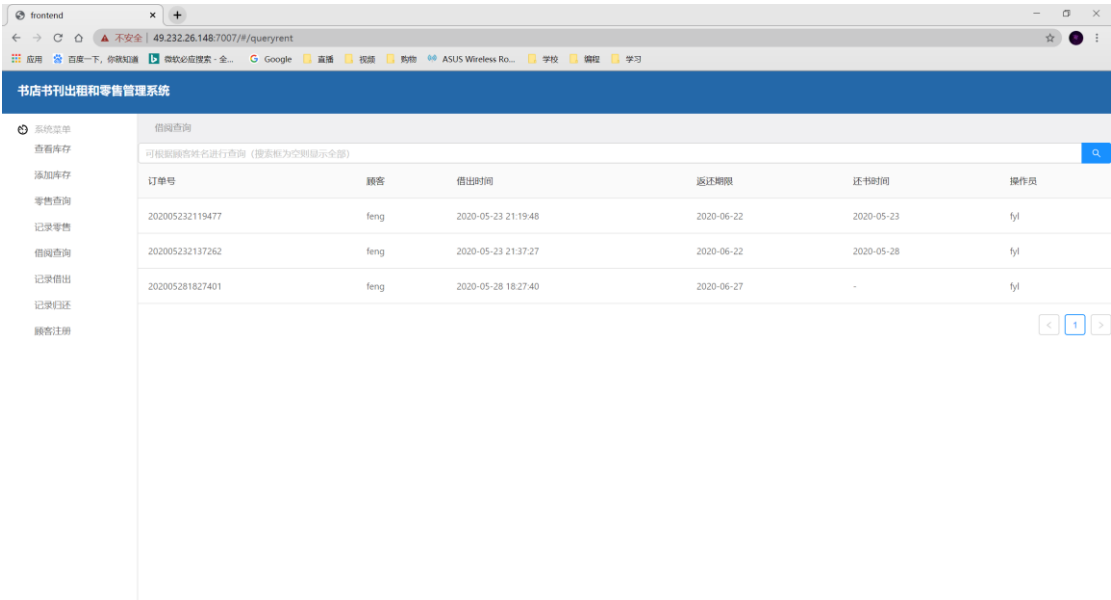


图 4-7 借阅查询界面

4.2.8 记录借出界面

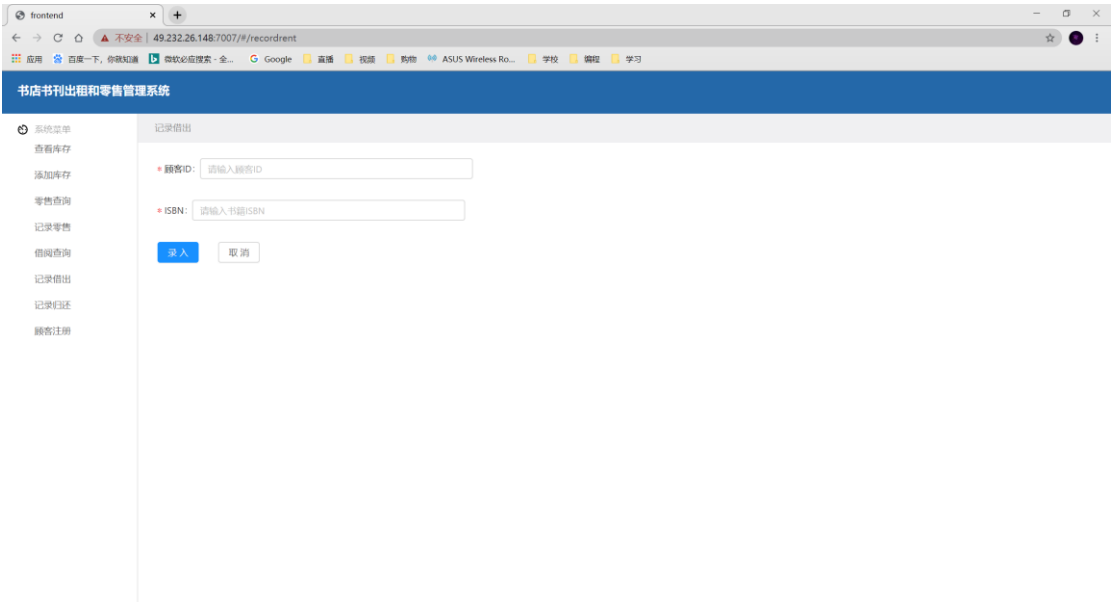


图 4-8 记录借出界面

4.2.9 记录归还界面

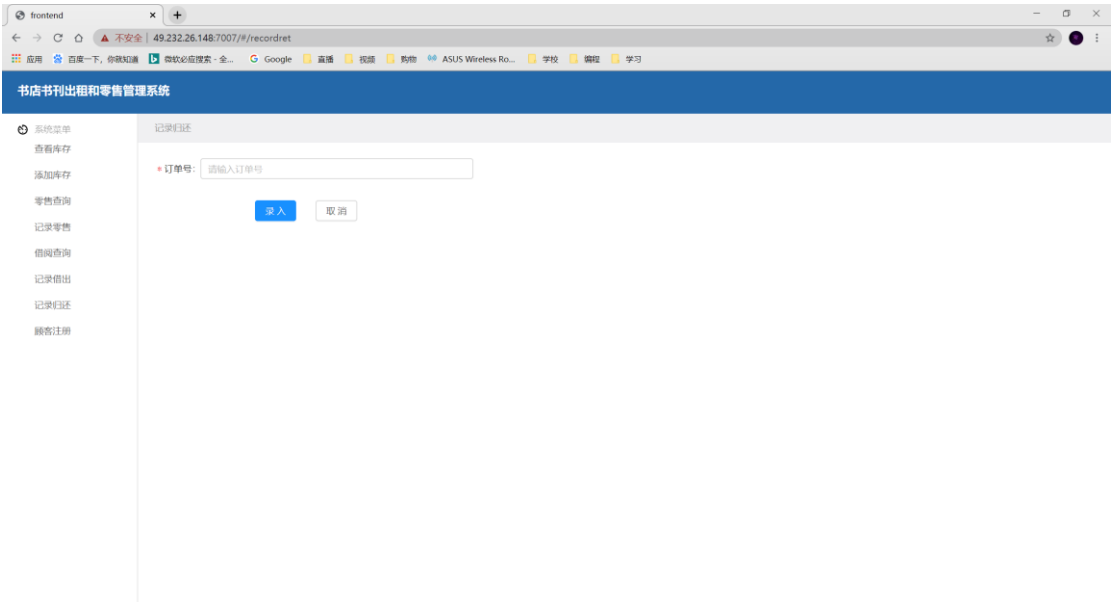


图 4-9 记录归还界面

4.2.10 顾客注册界面

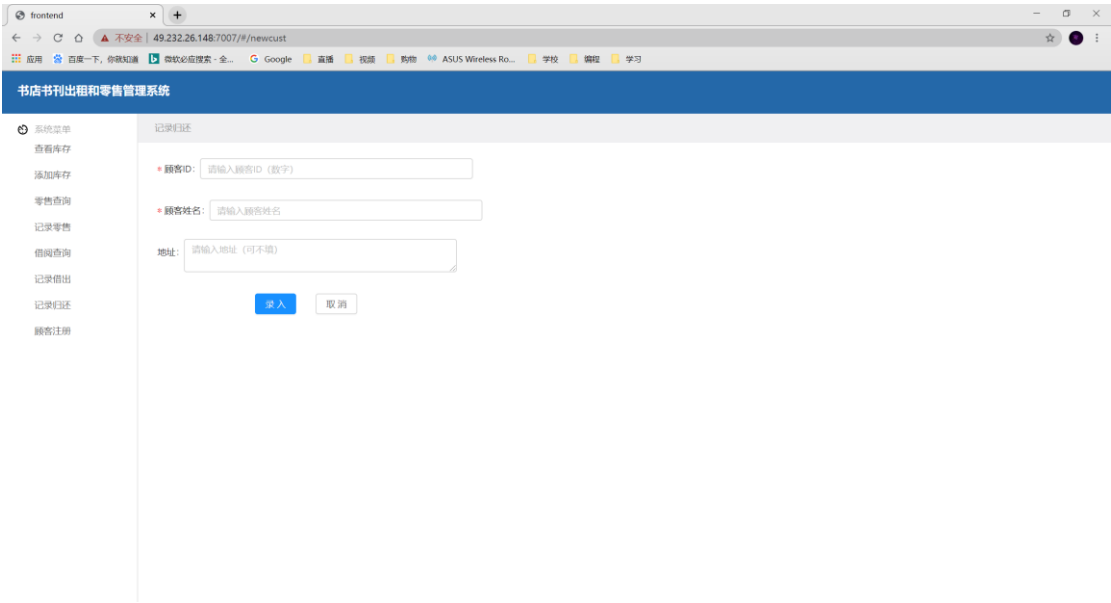


图 4-10 顾客注册界面

4.2.11 书本销量查询

在库存界面点击书本的 ISBN 号，即可看到书本的近期销量，可以指定日期范围。

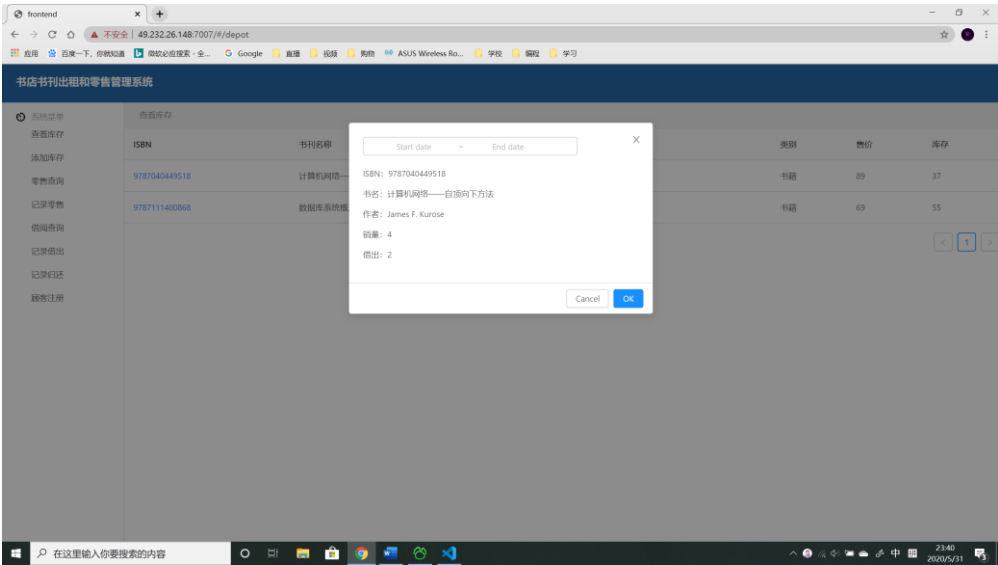


图 4-11 书本销量查询