

实验 5 图书管理系统

实验报告

3180102829 谢博洋 数字媒体技术 1801

1、 实验目的

设计并实现一个精简的图书管理系统，要求具有图书入库、查询、借书、还书、借书证管理等功能。通过该图书馆系统的设计与实现，提高学生的系统编程能力，加深对数据库系统原理及应用的理解。

2、 开发工具

主要使用 PHP/HTML/CSS 开发，数据库平台采用 MySql PHP APIs。

3、 总体设计

3.1 数据库表设计

3.1.1 收集需求

管理员登陆	输入管理员 ID, 密码; 登入系统 或 返回 ID/密码 错误.
图书入库	单本入库或 txt 格式批量入库
图书查询	根据书的各项信息进行查询, 根据用户选择排序
借书	查询已借书籍; 库存不为零时允许借书, 否则返回最近还书时间。
还书	查询已借书籍; 判断图书是否存在, 有则入库, 无则报错。
借书证管理	增加或删除一个借书证.

3.1.2 标志对象 (Entity) 和属性 (Attribute)

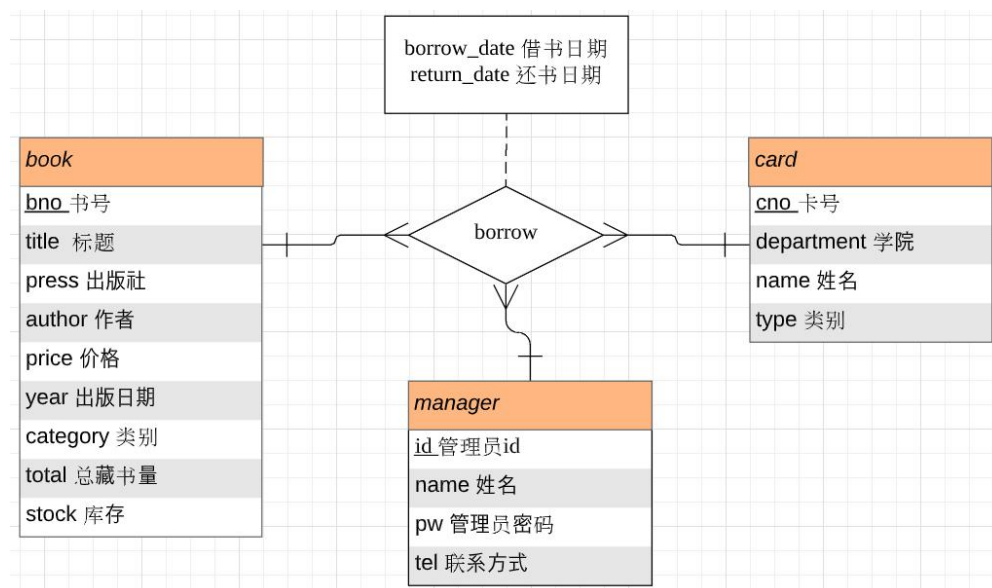
对象名称	包含属性
书	书号, 类别, 书名, 出版社, 年份, 作者, 价格, 总藏书量, 库存
借书证	卡号, 姓名, 单位, 类别 (T, G, U, O, 分别代表教师、研究生、本科生、管理人员)
管理员	管理员 ID, 密码, 姓名, 联系方式 (手机号)
借书记录	卡号, 书号, 借书日期, 还书日期, 经手人 (管理员 ID)

3.1.3 标志对象之间的关系 (Relation) 以及 E-R 图绘制

借书记录与书: 书库存为零时, 不能外借; 书不存在时, 不能对此书进行借书或还书操作。

借书记录与借书证: 借书证不存在时, 不能产生借书记录; 借书记录显示有书未还时, 借书证不能销毁。

借书记录与管理员: 管理员 ID 必须存在。



图书管理系统的 ER 图

3.2 流程设计

3.2.1 操作流程

本操作系统整体上分为**管理员后台**、**读者查询**两大模块。

读者查询不需要登录，通过入口后进入查询页（search 页），直接使用“reader”角色连接数据库，仅有 select 权利，可以使用模糊查询、复合查询和针对价格、出版年份的闭区间查询。对查询结果可以进行自定义单一标准的排序。

管理员后台不提供注册，新增管理员需直接操作数据库添加，防止恶意注册。用户输入的 ID 和 password 会传入数据库的 manager 表进行对比，匹配成功则跳转至后台。

后台的界面必须从 home 页登录进入，否则打印“非法进入”字样。将操作最为频繁的借书还书放在后台第一页（manage 页）。借书还书都要检查卡号、书号是否存在，另外借书还要检查是否有库存量、还书还要检查改卡持有者是否的确借过此书。出现任意检查报错，整个事务进行回滚。

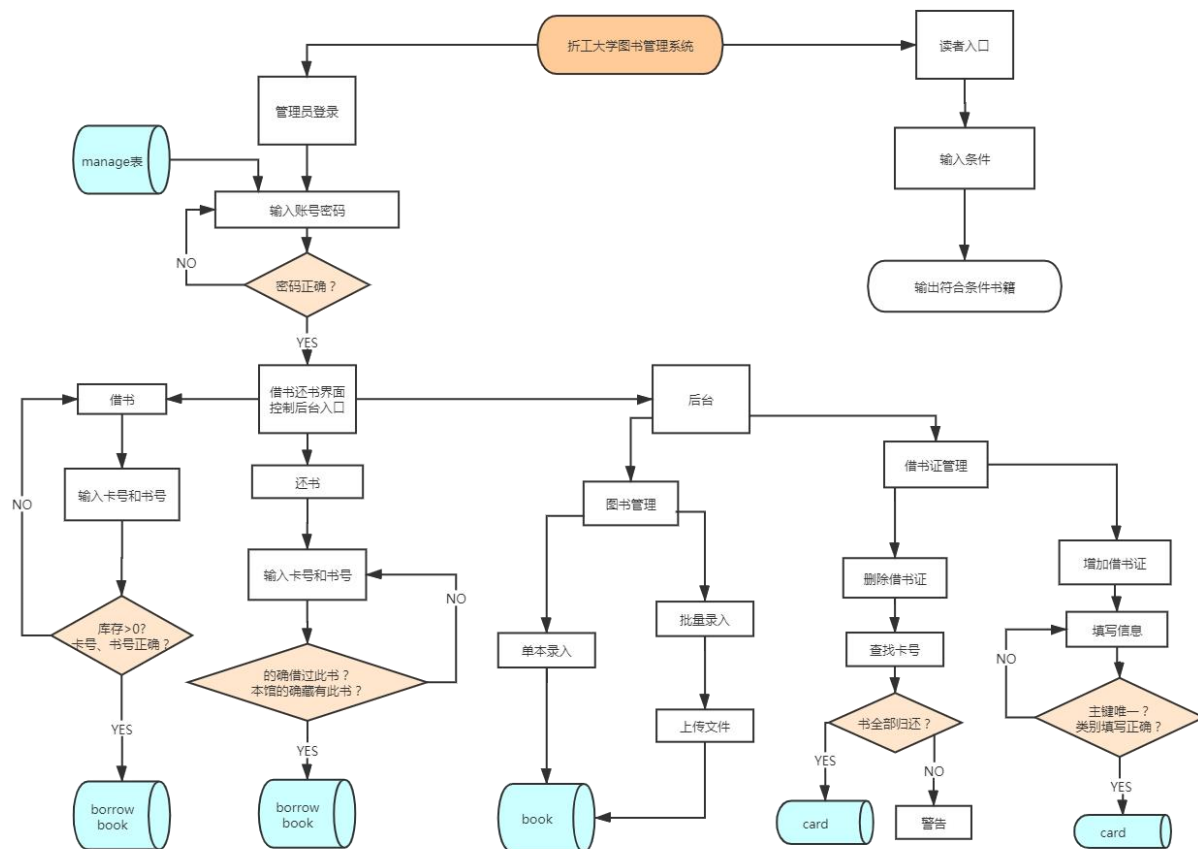
Manage 页同时有二级管理入口，管理员可以通过“图书管理”或“借书证管理”进入二级管理页面，或“退出”按钮返回 home 页。

借书证管理分增加和删除。

增加借书证时信息必须填写完整，传入数据库会对卡号进行唯一性检查，检查报错则回滚。删除借书证同时会导致该卡对应的借书记录一并删除，该删除不能通过重新添加相同信息借书卡修正，所以必须先查询，支持模糊查询，在查询结果中对要删除的借书证进行二次确认。

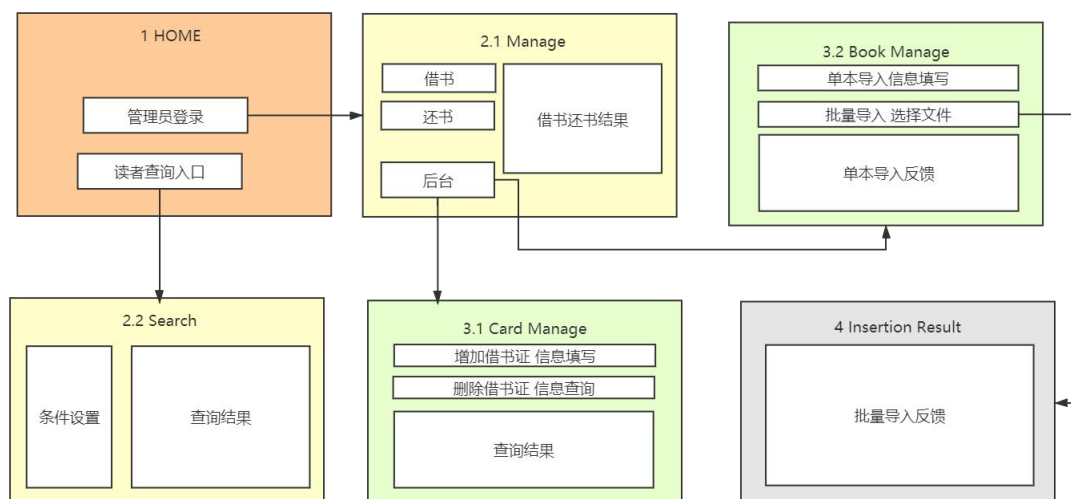
图书管理分单本录入和批量录入。

单本录入信息必须填写完整，传入数据库会对书号进行唯一性检查，如已存在，对比其他信息，一致则直接增加总藏书量（total）和库存量（stock），若存在信息不一致则报错。批量录入目前仅支持 UTF-8 编码的 txt 文件。单本和批量录入后均有反馈，单本录入的反馈信息直接打印在本页面，批量录入跳转显示。



操作流程圖

3.2.2 页面流程



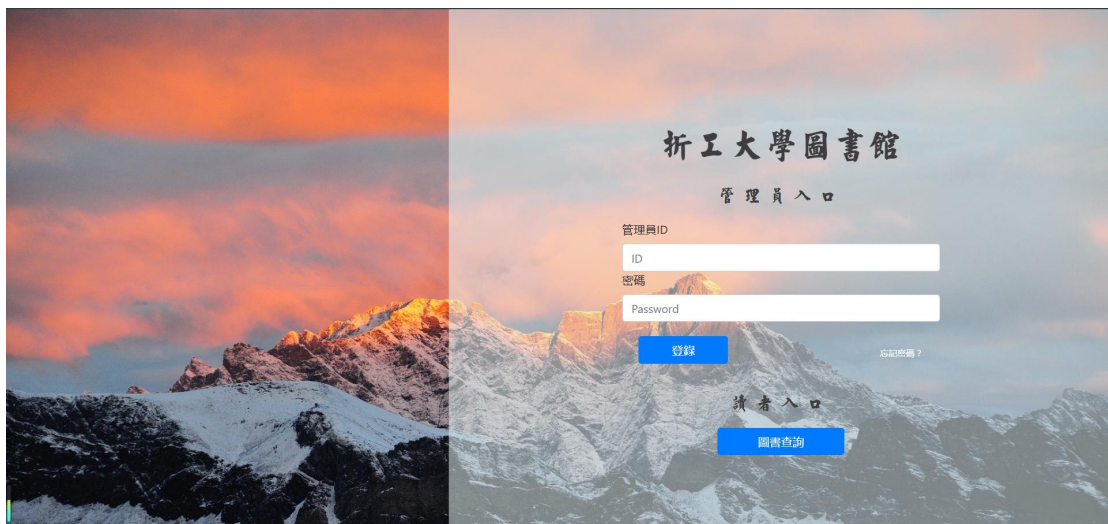
4、 详细设计

说明：

- * 流程图中黄色三角表示弹出错误提示框。
- * 背景框表示实现语言。
- * 数据为程序随机生成，不是操作失误生成的乱码。

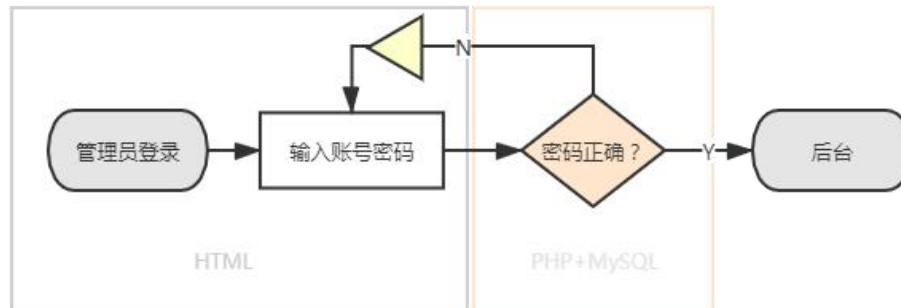
4.1 home 页

有管理员入口和读者查询入口选择。



4.2 管理员登录模块

4.2.1 流程图



4.2.2 实现技术说明

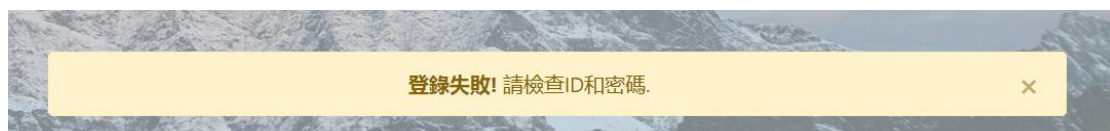
1. 管理员输入 ID 和密码信息
 2. 调用 MYSQL，判断跳转或弹出提示框。关键代码展示：
- 其中\$username/\$password 由前端传入，\$conn 为数据库连接相关信息。

```
$suprequest_sql = "select id from manager where id=$username
                  and pw = $password"; //申请检查username和password
$suprequest_query = mysqli_query($conn, $suprequest_sql); //发送数据库请求
if ($result = mysqli_fetch_array($suprequest_query, MYSQLI_NUM)) {
    //登录成功，全局保存用户信息，跳转
    $_SESSION['USER'] = $username;
    header("略");
} else {
    //登录失败
    if ($_POST['username'] != null) //防止打开界面就弹框，必须检查是否已经填写表单
        echo "略";
}
```

使用到的 sql 语句：

```
select id from manager where id=$username and pw = $password
```

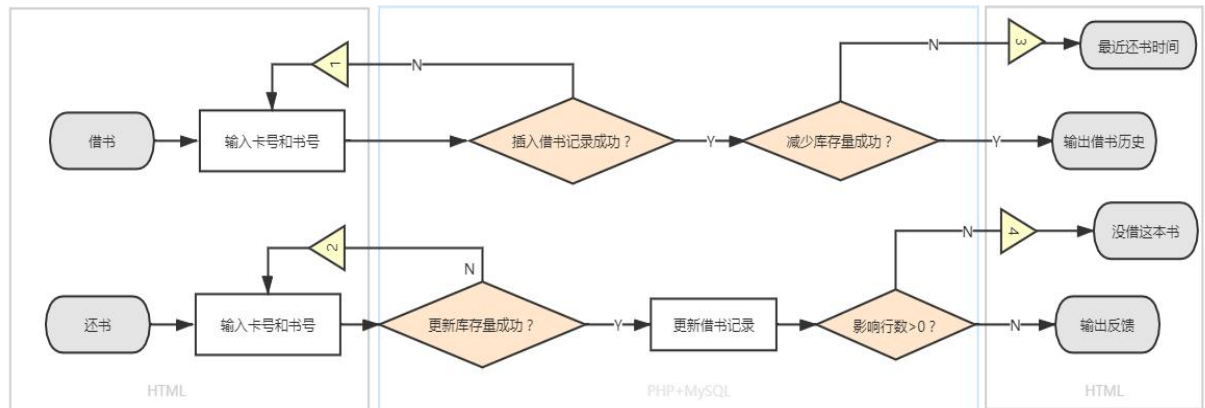
4.2.3 界面展示



登录失败弹框

4.3 借书还书模块

4.3.1 流程图



4.3.2 实现技术说明

1. 检查非法访问。在 home 对应的 php 文件中，如登录成功，会将全局变量 USER 赋值为当前管理员 ID。如未赋值，认为访问非法。关键代码：

```
<?php
error_reporting(E_ALL & ~E_NOTICE & ~E_WARNING);
session_start();
//如果不是从login进入，提示非法访问
if (!isset($_SESSION['USER'])) {
    die("非法访问" . $_SESSION['USER']);
}
?>
```

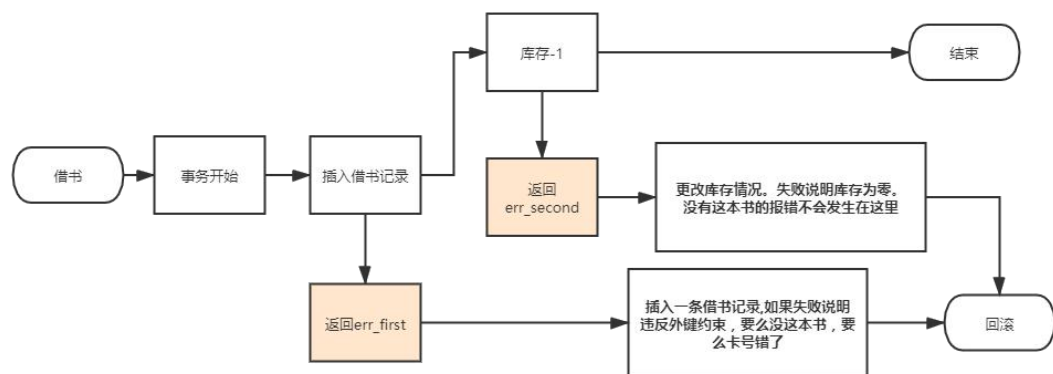
2. 借书、还书部分根据管理员填入的信息，分别生成 sql 语句。这部分除查询历史借书信息之外，二者的处理均不同，所以需分别构造 sql 语句。

使用到的 sql 语句：

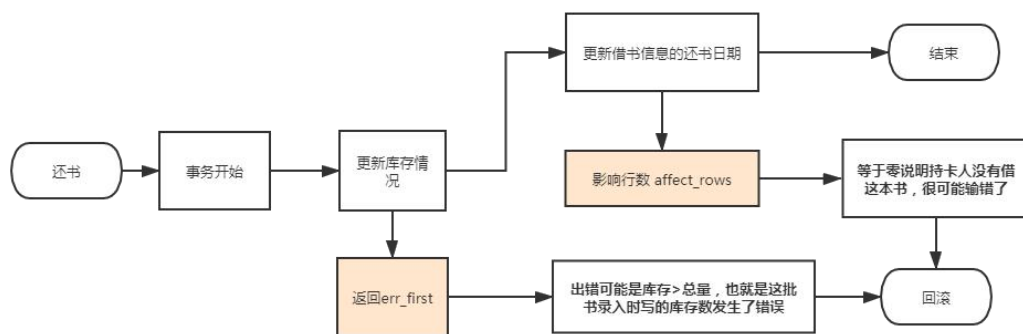
```
-- 借书:
select distinct * from borrow natural join book where cno= '$cno1' ORDER BY borrow_date;
INSERT into borrow VALUES('$cno1','$bno1',now(),null," . DB_MANAGE . " );
-- DB_MANAGE为管理员编号
update book set stock = stock-1 where bno = '$bno1';

-- 还书:
select distinct * from borrow natural join book where cno= '$cno2' ORDER BY borrow_date;
update book set stock = stock+1 where bno = '$bno2';
update borrow set return_date = now() where return_date is null and bno = '$bno2' and cno = '$cno2';
```

3. 分别执行 sql 语句，获取返回信息
4. 根据返回信息判断是否发生了步骤 1 中列举的错误，用 ifelse 语句依次检查。



借书的查错逻辑



还书的查错逻辑（还书就不考虑没有这本书了）

```

if (!$err_second || !$err_first || !$affect_rows) {
    if (!empty($cno1)) {
        if (!$err_first) {echo "插入借书记录操作失败弹框（略）";}
        elseif (!!$err_second)) {
            //查询最后的还书记录
            $latest_date_sql =
                "SELECT return_date FROM borrow WHERE bno ='$bno1'
                ORDER by return_date DESC LIMIT 1;";
            $err_date = mysqli_query($conn, $latest_date_sql);
            $date_result = mysqli_fetch_array($err_date)[0];
            echo "借书失败和最近归还时间（略）";
        }
    }
    elseif (!empty($cno2)) {
        echo "弹窗（略）请检查：借书卡是否正确；持卡人是否持有本书；书库库存量是否登记有误";
    }
    mysqli_query($conn, "ROLLBACK");
}
else {
    echo "操作成功的弹窗（略）";
    mysqli_query($conn, "commit");
}

```


4.3.3 界面展示



借书成功界面

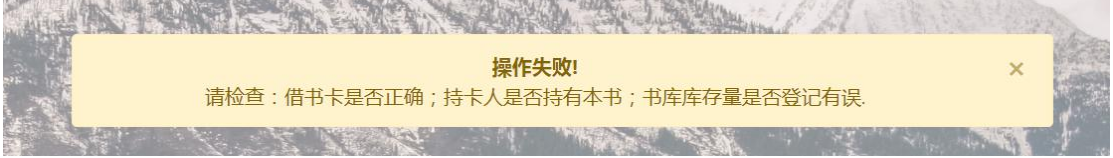
框选区分别是：成功提示、本次查询记录、历史借书记录、滚轮



库存为零导致的借书失败



还书成功 刷新借书记录



还书失败的提示

4.4 查询模块

查询模块的存在意义主要是做一个统计。

4.4.1 实现技术说明

主要实现借书总量、未还书总量和历史借书记录的查询，方便不进行还书借书操作时的信息获取。代码较简单。涉及到的 sql 语句有：

```
select distinct * from borrow natural join book where cno= '$cno3'
ORDER BY borrow_date DESC;
select count(bno) from borrow where cno='$cno3';
select count(bno) from borrow where cno='$cno3' and return_date is null;
```

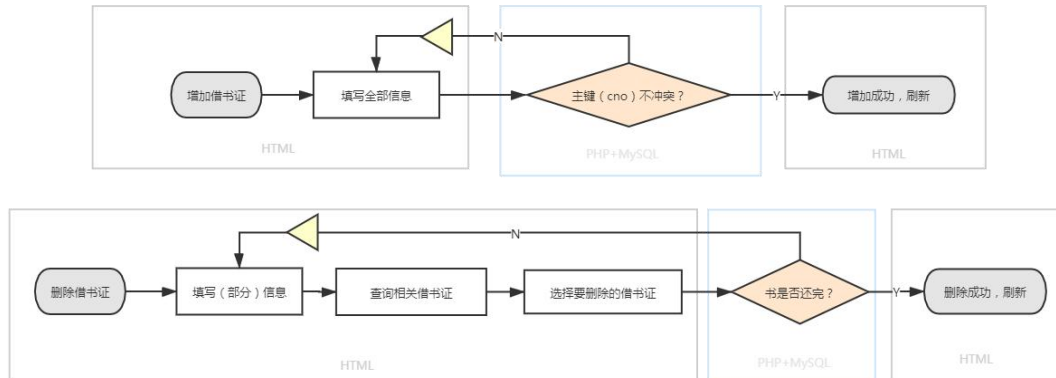
4.4.2 界面展示



借书统计

4.5 借书证管理

4.5.1 流程图



4.5.2 实现技术说明

1. 增加借书证时，需要考虑的就是借书证号不能重复，这一点通过设置 cno 为主键进行限制。

2. 删除借书证时，会同时删除借书记录，所以需要进行二次确认。在本系统中强制用户进行一次搜索，可以模糊搜索或部分条件搜索，在搜索结果中选择需要删除的元组。

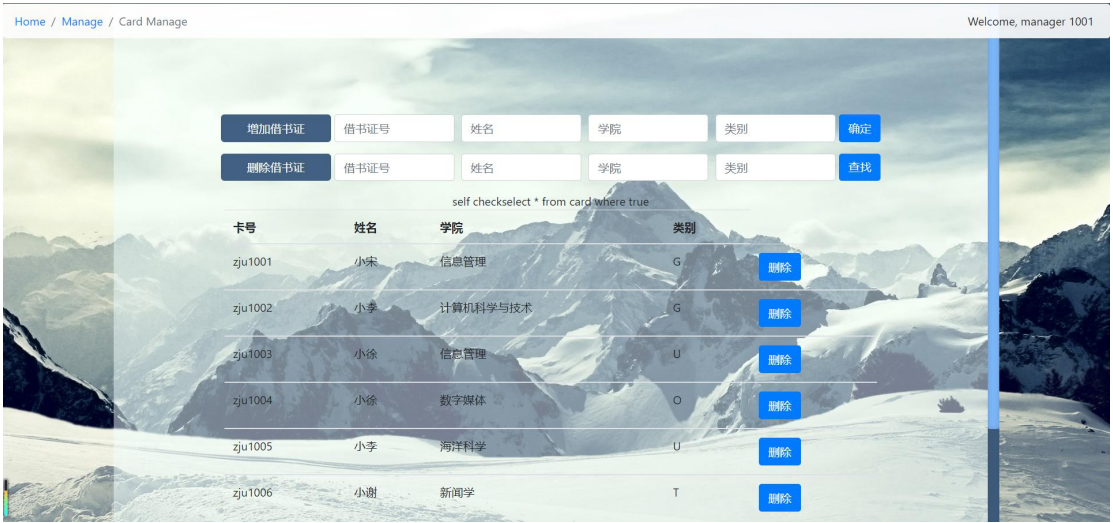
在删除条件中可以通过限制“未还书”为空集来约束未还书不能销卡，可以避免增加触发器，以及避免触发器带来的代价。

涉及到的 sql 语句有：

```
delete from card
where cno = '$my_cno' and not exists
    (select *from borrow
     where return_date is null and borrow.cno = '$my_cno');

insert into card values ('$cno','$name','$department','$type')
select * from card where true $where
-- where是根据选项生成的筛选条件
```


4.5.3 界面展示



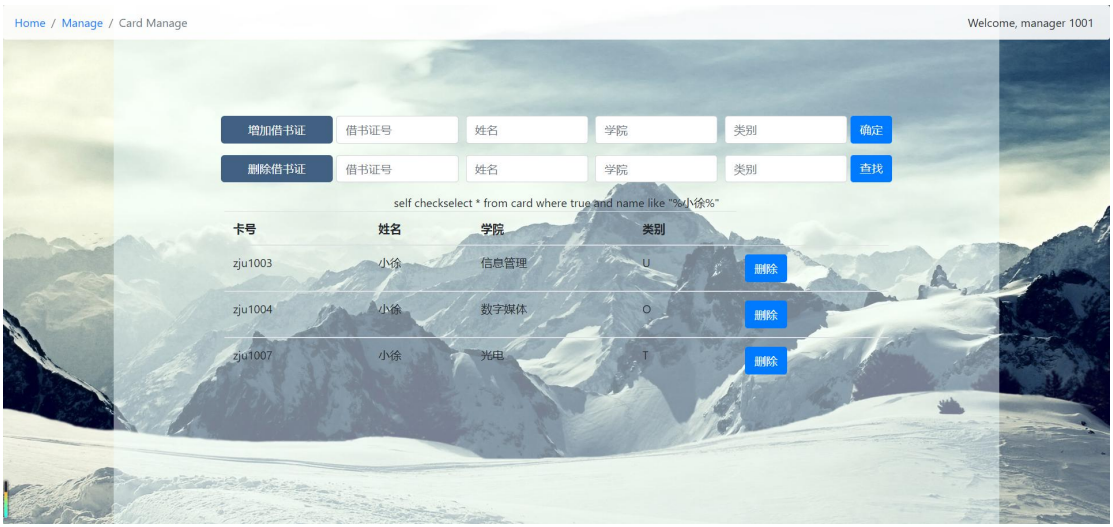
借书证管理界面



新增借书证成功



新增借书证失败



查找姓名为“小徐”的借书证



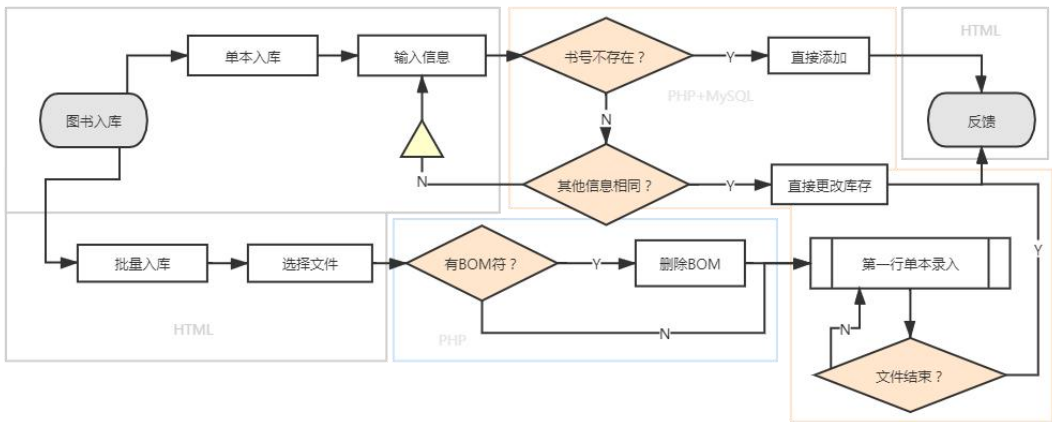
删除 zju1004 成功



删除 zju1003 失败, 还有书未归还

4.6 图书管理模块

4.6.1 流程图



4.6.2 实现技术说明

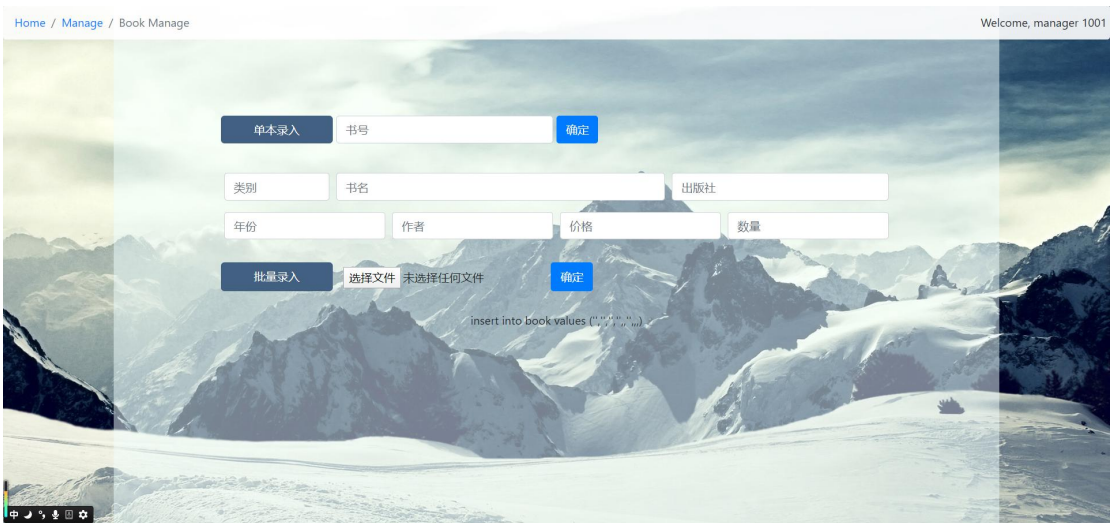
1. 单本入库时，信息必须填写完全，首先检查 bno 书号是否重复，如果重复则对比其他信息，如一致，说明是扩容操作，直接更改库存量和总藏书量；如其他信息不同，说明书号分配有误。
2. 批量入库时，目前仅支持 UTF-8 编码的 txt 格式。Windows 的 txt 格式可能带有 BOM，先统一去除再一行一行读取并进行单本入库操作。出错时，保留出错信息，继续读取。结束后统一显示更新信息。

涉及到的 sql 语句有：

```
select * from book where bno = '$bno' and title = '$title'
-- 用title来鉴别是否是扩容操作
update book set total = total + $total, stock = stock+$total where bno = '$bno'
-- 是扩容，更新数目

insert into book values
('$bno','$category','$title','$press',$year,'$author',$price,$total,$total)
```

4.6.3 界面展示



图书管理界面



单本更新成功



单本更新出错



批量导入选择文件



批量导入结果

分别展示：BOM 头去除、上传文件的基本信息、成功的反馈（成功在对应的 sql 下方）、出错的反馈。

4.7 图书查询模块

4.7.1 流程图



4.7.2 实现技术说明

1. 条件筛选部分。通过 html 传递的信息，不断完善 where 字段。以 title 为例说明：首先 html 端输入书名，并通过 ajax 将表单提交到后端。PHP 定义 \$title 接收表单内容，如果非空，说明存在该筛选条件，添加进 where 字段。最后查找时使用 true \$where，避免单独处理“and”。

```
html:
<p><label class="column_name">书名</label> <label>
  <input placeholder="包含" class="column_input" type="text" name="title"/></label></p>

PHP:
$title = $_REQUEST['title'];
if (!empty($title)) {
    $where .= " and title like \"${title}%\"";
}
$sql = "select * from book where true $where order by $order";
```

2. 排序部分，支持一次单击正序，再次单击倒序。点击表头时，set_order 会接收到不同的参数，从而对全局变量 my_option 进行更新；之后附加在表格序列化前端，通过 AJAX 传给 PHP 文件并进行排序和表头箭头的刷新。仍以 title 为例，展示关键代码：

```
PHP
if ($order == 'bno desc')
    echo "<th scope='col' onclick='set_order(1)'>编号</th>";
else
    echo "<th scope='col' onclick='set_order(1)'>编号</th>";

JS
var my_option;
set_order = function (opt) {
    switch (opt) {
        case 1:
            if (my_option !== 'bno desc')
                my_option = 'bno desc';
            else my_option = 'bno asc';
            break;
        case 2: ...
    } ...
}

AJAX
var my_data = $.param({'order': my_option}) + '&' + $('#condition').serialize();
$.ajax({
    ...
    data: my_data,
    success: function (result) {
        document.getElementById("search_result").innerHTML = result;
    },
    ...
});
```

4.7.3 界面展示

Home / Search

书名

包含

作者

包含

出版社

包含

类别

类别

年份区间

最低

最高

价格区间

最低

最高

查询

查询结果

select * from book where true order by title desc

编号	类别	名称	出版社	年份	作者	价格	总量	库存
kG1GQ	画册	鳞俾琢朕悱卦惑洮意市	博翼出版社	1831	独孤钙择	78.60	198	198
y16T0	体育	铤例担踟旁羊篇沛骀墙	郁太饶御都出版社	1960	基毋妨	80.26	297	297
OyC36	数理化参考书	陆聂连七多昂鄯	鄯赜冀冀曹出版社	1808	杭朵友	53.87	86	86
7a2p4	画册	贰偈拂播低亿辰尽驭基焚壤	契鑫出版社	1900	北宜高伙	30.98	285	285
00014	历史	计算机的发展	电子出版社	2004	俞乙子	60.00	5	0
0Y1WJ	小说	裹攀砌院逐狼八格滔膜判窟	施捺思出版社	1978	邵辰	89.43	96	96
q6xT5	计算机	膺硅棠强榷蕉宅幼夤频沐	穿梨姜掖奇出版社	1982	辜湖畔	9.82	76	76
7452V	历史	篇暖免波	够束饶塑出版社	1833	乔佑谓	72.18	249	249
00008	历史	程序的发展	电子出版社	2003	俞乙子	60.00	5	5

图书查询首页

Home / Search

书名

包含

作者

郭

出版社

包含

类别

计算机

年份区间

最低

最高

价格区间

最低

最高

查询

查询结果

select * from book where true and author like "%郭%" and category like "%计算机%" order by title asc

编号	类别	名称	出版社	年份	作者	价格	总量	库存
00013	计算机	CG从入门到放弃	黄河HUANGHE大学出版社	2016	郭扬州	9.80	5	3
00007	计算机	SQL Server 从入门到放弃	清水大学出版社	2006	郭扬州	9.80	5	3

复合查询、模糊查询示例

Home / Search

书名

包含

作者

包含

出版社

包含

类别

类别

年份区间

2001

最高

价格区间

最低

80

查询

查询结果

select * from book where true and year >= '2001' and price <= '80' order by title desc

编号	类别	名称	出版社	年份	作者	价格	总量	库存
00014	历史	计算机的发展	电子出版社	2004	俞乙子	60.00	5	0
00008	历史	程序的发展	电子出版社	2003	俞乙子	60.00	5	5
00002	计算机	程序员的我修养	电子工业出版社	2013	俞甲子	65.00	5	0
00005	英语	实用英文写作	高等教育出版社	2008	庞继贤	38.00	3	2
00003	教育	做新教育的行者	福建教育出版社	2002	高云鹏	25.00	3	2
00009	政治	做新政治的行者	福建出版社	2012	高山鹏	52.00	3	2
00015	政治	做新政治的先锋队	福建出版社	2012	高山鹏	52.00	2	2
00010	教育	做孩子眼中没本领的父母	工业出版社	2003	高山鹏	32.00	5	5
00004	教育	做孩子眼中没本领的父母	电子工业出版社	2013	高云鹏	23.00	5	5
00011	英语	不实用英文写作	高等教育出版社	2008	庞继贤	33.00	3	2

区间查询示例

查询结果 ook where true order by year desc			查询结果 ook where true order by year asc		
出版社 ▲	年份 ▼	作者 ▲	出版社 ▲	年份 ▲	作者 ▲
浙江大学出版社	2018	王丽丽	苦慢坐缺出版社	1805	卢澜
黄河HUANGHE大学出版社	2016	郭扬州	叫碌打感出版社	1806	嵇磬痊
电子工业出版社	2013	俞甲子	鲫锡冀氨曹出版社	1808	杭朵友
电子工业出版社	2013	高云鹏	素排啤嫌裂出版社	1820	令狐挪
福建出版社	2012	高山鹏	抛龛出版社	1831	独孤钙择
福建出版社	2012	高山鹏	够柬饶墮出版社	1833	乔佑谓
高等教育出版社	2008	庞继贤	卫杰先国岳出版社	1833	储廷厢

两种自定义排序效果

5、 特色设计

1. 安全性设计

在需要管理员权限的页面均增加了非法访问检查，具体请见 4.2.2

2. 可视化界面

在呈现形式上进行了初步的设计，提高了美感和可操作性

3. 基本的统计信息

在查询结束记录界面增加基本的统计，方便了年末总结等查询

4. 自定义排序

支持一次点击正序，再次点击倒序

5. 模糊查询

在借书证查询、图书查询两处都提供模糊查询。借以提高效率和便捷性

6. 完备的反馈

用户的每一步操作都有反馈，指出可能的错误或对成功操作进行确认，增加了使用舒适度