**目录**

# 1 引言

## 1.1 编写目的

本文详细描述任务管理系统的需求，表述的需求信息要求明确、无二义性。开发方与软件使用者充分沟通需求，最终形成此文档。此文档是后续软件开发的依据。

## 1.2 背景

图书管理系统是一个文华学院与文华学院图书馆产学研合作项目，项目由文华学院教务处提出，由文华学院信息学部承担开发任务。

## 1.3 用户

# 图书管理系统的目前用户为文华学院图书馆，文华学院图书馆使用成功后可能会在武汉市各大高校推广。

# 2 任务概述

## 2.1目标

本项目的设计目标旨在方便[图书管理员](https://baike.baidu.com/item/%E5%9B%BE%E4%B9%A6%E7%AE%A1%E7%90%86%E5%91%98" \t "_blank)的操作，减少图书管理员的工作量并使其能更有效的管理书库中的图书，实现了传统的图书管理工作的信息化建设。

本系统的服务对象为图书馆流通部门的工作人员，用户界面友好，不需计算机专业的专门训练即可使用本系统。

## 2.2 系统的特点

图书管理系统的需求主要由文华学院教务处提出，因此本文档是与文华学院教务处交互后形成的需求定义，系统的功能和使用特点优先满足文化学院教务处的需求，若系统后续由于在各大高校全面推广而引入的新需求，则不在本文档考虑范围之内。

## 2.3 假定和约束

本文档经双方确认后，开发方依据本文档进行下阶段工作。若中途需求发生变更则文化学院教务处需及时告知开发方，若因文华学院教务处原因引入的需求变更造成开发方工作量的大幅增加，具体解决方案双方另行协商。若需求变更引入的工作量不大，开发方应尽量配合。

## 2.4 团队介绍

团队分工：

一个良好的团队，离不开团队成员每个人的努力，对成员的信任，对队长的信任，是一个良好的团队不可或缺的部分。

|  |  |  |
| --- | --- | --- |
| 姓名 | 角色 | 工作描述 |
| 谭昭俞 | 队长 | 分配成员任务，决策的实施 |
| 李秀昕 | 副队长 | 测试计划，ER图，完善系统架构设计 |
| 刘江川 | 组员 | 功能分析的四象限，完善需求规格说明书 |
| 周云乔 | 组员 | 任务分解WBS |
| 王浩 | 组员 | 系统架构设计 |
| 吴代乐 | 组员 | 思考用户痛点，描述使用场景 |

# 3 需求规定

## 3.1 需求分析

产品优点：减少人力成本和管理费用、良好的人机交互界面、提高信息的准确性和安全性、改进管理的服务、数据精准度：保证查询的查全率和查准率为100%，所在相应域中包含查询关键字的记录都能查到。系统响应时间：系统对绝大多数的响应时间在1——2秒内。适应性：满足运行环境在允许操作系统之间的安全转换和与其他应用软件的独立运行要求

## 3.2 用户需求分析：

能够对一定数量的读者进行相应的信息存储与管理

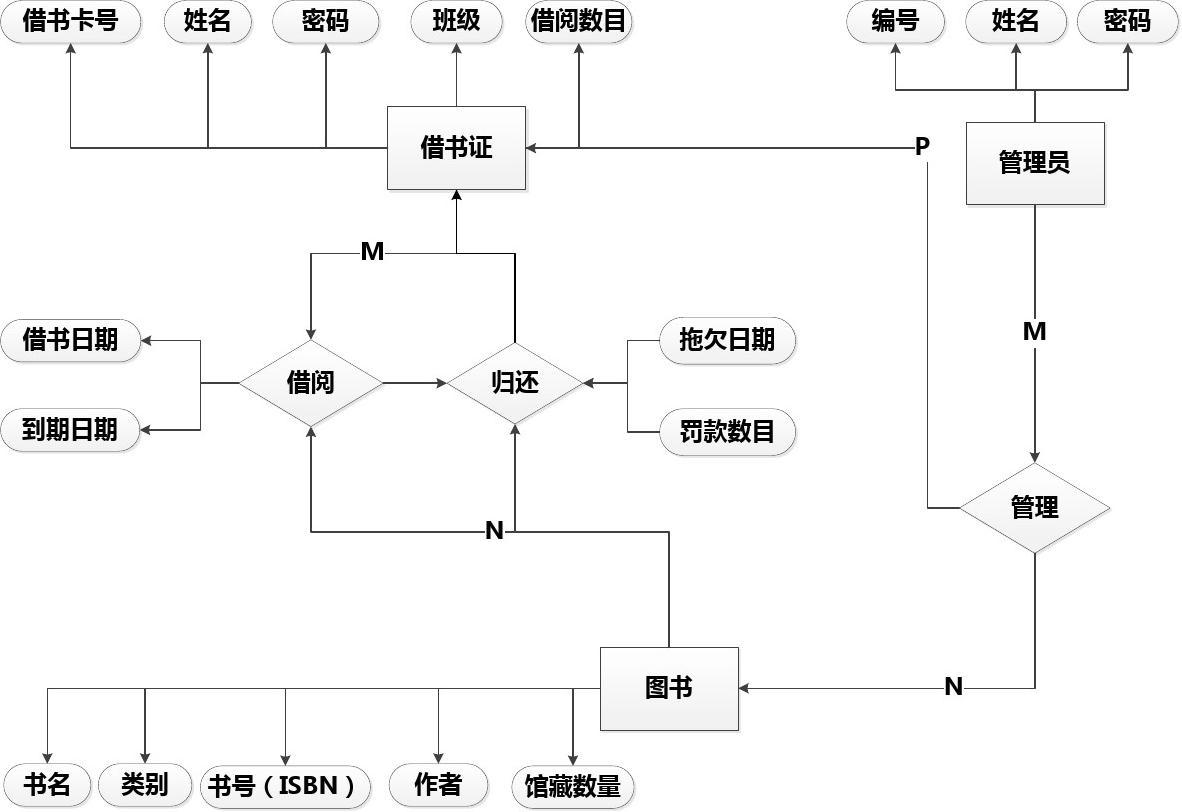
图书信息的多关键字检索查询.图书的出借、返还和资料统计

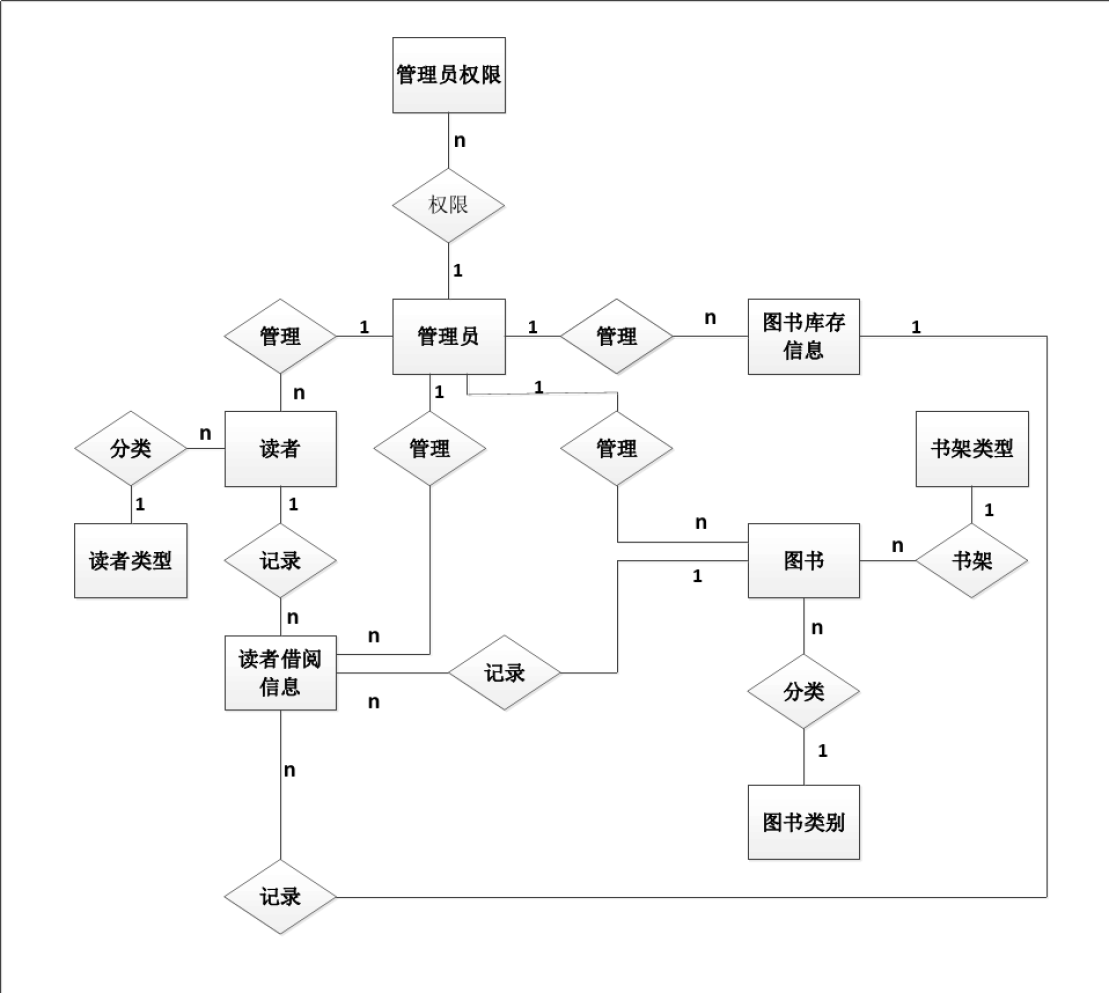
读者信息的登记、删除及修改。读者资料的统计与查询.能够提供一定的安全机制，提供数据信息授权访问。

能够存储一定数量的图书信息，并方便有效的进行相应的书籍数据操作和管理

## 3.3 任务处理流程需求

## 3.3.1 任务流程图：





# 4 详细功能需求

## 4.1 产品介绍

组成结构有四大部分：即信息源、信息处理器、信息用户、信息管理者。

相关专业知识：合运用了管理科学，系统科学，运筹学，统计学，计算机科学等学科的知识。

基础概念：一个由人、计算机等组成的能进行管理信息的收集、传递、加工、保存、维护和使用的系统

## 4.2 详细功能介绍

产品的主要功能为用户可自行借阅和归还图书等管理7个子功能及代码：

1. 借阅图书窗口；

<?php

namespace Admin\Controller;

use Base\BaseController;

use Admin\Model\BorrowModel;

final class Borrow extends BaseController{

public function index(){

$this->accessPage();

$this->smarty->display("Borrow/index.html");

}

1. 归还图书窗口；

use Admin\Model\BookModel;

final class Index extends BaseController{

public function index(){

$this->accessPage();

$userModel = new UserModel;

$userNum = $userModel->rowCount("admin != 1");

$bookModel = new BookModel;

$bookNum = $bookModel->rowCount();

$this->smarty->assign("userNum",$userNum);

$this->smarty->assign("bookNum",$bookNum);

$this->smarty->display("Index/index.html");

}

}

1. 个人借阅查询；

<?php

namespace Base;

use Tool\Db;

abstract class BaseModel{

protected $Db;

protected $table; public function \_\_construct(){

$this->Db = Db::getInstance();

}

public function fetchOne($where){

$sql = "SELECT \* FROM {$this->table} WHERE {$where}";

return $this->Db->query($sql);

}

public function fetchAll($where = "2>1"){

$sql = "SELECT \* FROM {$this->table} WHERE {$where}";

$result = $this->Db->query($sql);

if(count($result) == count($result,1) && !empty($result)){

$result = array($result);

}

return $result;

} public function rowCount($where = "2>1"){

$sql = "SELECT \* FROM {$this->table} WHERE {$where}";

return $this->Db->rowCount($sql);

}

1. 催还书目查询；

public function update($data,$where){

$update = "";

foreach($data as $key => $value){

$update .= "`{$key}`='{$value}',";

}

$update = rtrim($update,",");

$sql = "UPDATE {$this->table} SET {$update} WHERE {$where}";

return $this->Db->exec($sql);

}

public function delete($where){

$sql = "DELETE FROM {$this->table} WHERE {$where}";

return $this->Db->exec($sql);

}

}

1. 图书分布查询；

<?php

namespace Common\Controller;

use Base\BaseController;

use Common\Model\UserModel;

use Tool\Verify;

final class Login extends BaseController{

private function checkLogin(){

if(isset($\_SESSION['userId'])){

$p = $\_SESSION['admin'] ? "Admin" : "Home";

header("location:?p={$p}&c=Index&a=index");

die();

}

}

public function index(){

$this->checkLogin();

$this->smarty->display("login.html");

}

public function showVerify(){

new Verify;

}

public function login(){

header("Content-Type:application/json");

$rightCode = strtolower($\_SESSION['verifyCode']); $code = strtolower($\_POST['verify']);

$userId = htmlentities($\_POST['userId']);

$password = md5($\_POST['password']);

if($code != $rightCode){

$this->sendJsonMessage();

}

$userModel = new UserModel;

$where = "id='{$userId}' and pwd='{$password}'";

$result = $userModel->fetchOne($where);

if(!empty($result) && $result['status'] == 1){

$\_SESSION['userId'] = $userId;

$\_SESSION['admin'] = $result['admin'];

$\_SESSION['last\_login\_time'] = $result['last\_login\_time'];

1. 按关键字查询；

}else if(!empty($result) && $result['status'] == 0){

$message = array("message"=>""code"=>1);

}else{

$message = array("message"=>""code"=>1);

$\_SESSION = array();

session\_destroy();

}

echo json\_encode($message,JSON\_UNESCAPED\_UNICODE);

}

public function logout(){

$\_SESSION = array();

session\_destroy();

header("location:?p=Common&c=Login&a=index");

}

}

#### 4.2.1. 借阅图书窗口：

#### 在系统主窗口中选择“借阅图书”，进入“借阅图书”，使用时单击“借阅”按钮或直接按回车键，光标定位于“借阅人编码或姓名”项。

#### 4.2.2. 归还图书窗口：

#### 在系统主窗口中，选择“归还图书”，进入“归还图书”窗口。单击“还书”选项或直接按回车键，光标定位于“借阅人编码或姓名”项

**产品还带有信息查询功能**

#### 4.2.3. 个人借阅查询：

窗口将显示该人员的全部借阅信息。如果系统中无该借阅人或该借阅人未借阅过任何图书，窗口将不显示任何信息。

#### 4.2.4. 催还书目查询：

进入“催还书目查询”窗口，该窗口中显示所有已到期但尚未归还的书目信息。

图书分布查询：进入“图书分布查询”窗口，该窗口在初始状态下显示所有书目的摘要信息。

#### 4.2.5. 按关键字查询：

#### 窗口显示所有书目的的摘要信息。若查询结果只有一本书，系统将直接显示该书的具体信息。

系统模块

基本数据信息维护模块:

#### 用于实现对用户及书籍的管理

#### 4.2.7. 基本业务模块:

#### 用于实现用户和系统交互的一些操作，例如借还书、留言等。

#### 4.2.8.数据库管理模块:

#### 用于管理用户信息、书籍信息及留言信息等。在系统中，对于所有的书籍信息、用户信息及其他信息都需要统一管理。

#### 4.2.9. 基本业务模块:

信息查询模块: 用于实现查询书籍信息及借阅者的信息。

借书：处理借书业务

还书：处理还书业务

续借书籍：借阅者可以通过系统进行续借已经借阅的书籍，即延长借书时间

预约书籍：借阅者可以通过系统进行预约书籍

取消预约书籍：借阅者可以取消预约书籍。管理员也可以视情况进行取消预约书籍。

留言：借阅者可以通过留言反应问题。管理员需要审核留言，及通过留言解答借阅者的疑惑或是采纳建议及时给出回复

# 5 前景展望

***欠缺资源和信息整合：***

在许多图书馆，虽然有建设数字图书馆和纸质图书，但没有整合两个的信息。这显著降低了数字图书馆传达的信息的准确性和及时性，也容易引起读者得到错误的信息，降低图书馆操作的效率

***业务内容局限：***

目前的功能并没有办法有效地管理图书馆的所有资源，图书馆管理系统还应该进行更全面的业务发展，如电子文档，财务管理，资产管理，党员管理，努力满足现代图书馆服务的需求。

***图书馆管理核心不明确：***

一般的系统没有办法对图书馆的运行情况进行一个综合的评价，也没有办法获得读者对图书馆的体验的满意程度，缺少决策支持的信息源头，容易给图书馆的管理造成一定的信息获取阻碍，导致图书馆管理决策偏差。

我们以后要重点把握资源、服务、管理这三者的协调性和统一性。有效地提高图书馆管理系统的服务质量，并构建出顺应时代要求的图书馆。

利用数字化新技术科学地分析出各种数据，比如各图书种类的受欢迎程度和读者的阅读爱好等，然后对不同的读者做出不同的的推荐，借此提高图书馆的服务质量和工作效率。

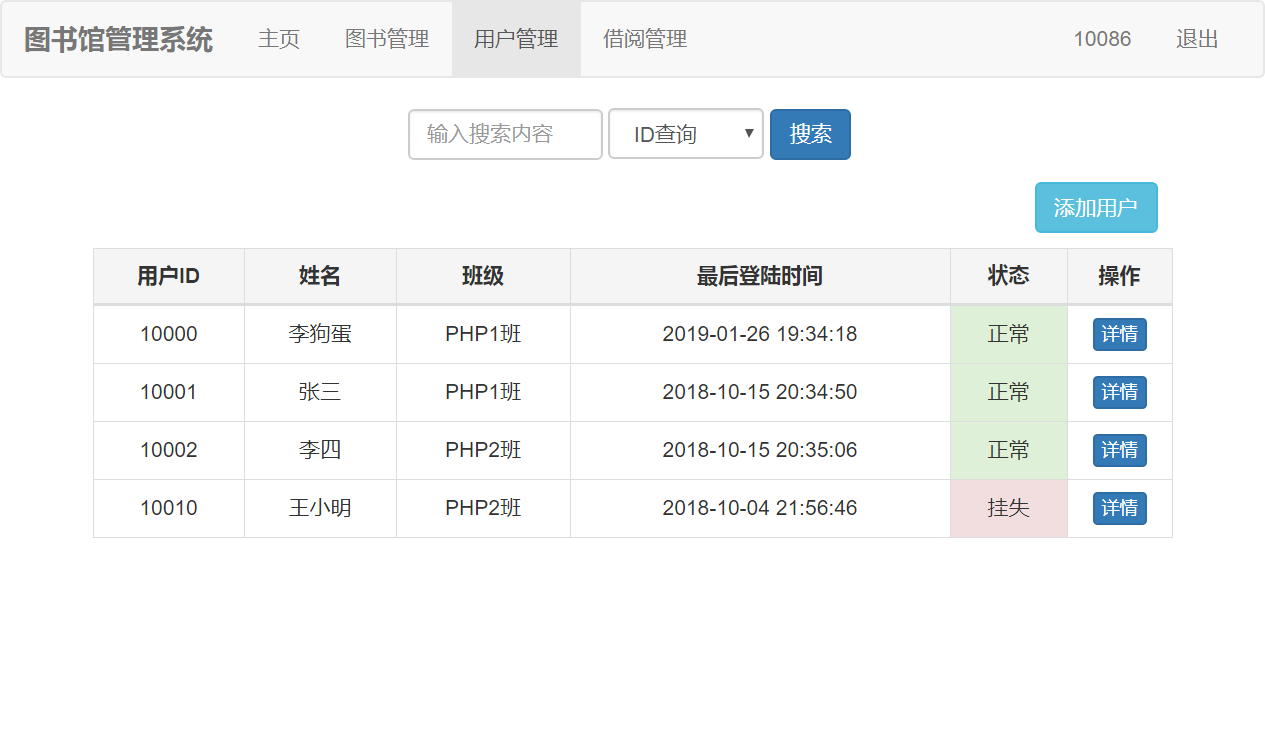
增强与各方面相互合作的理念，最大程度地和各个方面和谐、友好地进行合作，以此来保证各类资源都可以得到完全整合，只有这样才能快速地开展各类工作，获得高效的图书馆管理系统工作效率，并为读者提供更优质的服务。

# 6 软件截图





# IMG_256







# 7 实验总结

本次课程设计利用“4+1”视图建模方法进行“图书管理系统”的软件架构设计。架构内容以及设计流程为图书管理系统系统概述、需求分析、场景视图之用例图设计建模、逻辑视图之类图设计建模、发开视图设计建模、过程视图之交互图设计建模、物理视图之部署图设计建模。其中需求分析包括角色分析、系统需求分析以及系统功能模块图三个部分；开发视图设计建模包括包图设计建模和组件图设计建模。

本次课程设计的完成主要依赖于软件体系结构课程所学的相关知识，再加上适当的相关参考文献的阅读，才得以将整个软件架构课程设计很好的完成。整个课程设计的过程中，除了对“4+1”视图建模方法的具体使用以及对设计模式的理解，还非常考验对整个软件从需求的提出到整个架构建模实现的流程思路和处理逻辑。是在设计的过程中，加深了我对课堂所学知识的理解，通过实际的练习，熟练的掌握了整个软件架构设计的思路、流程以及方法，收获丰硕。