

**《面向对象程序课程设计》**

**课程实践报告**

项目名称 图书信息管理系统

系 部 计算机科学与工程系

班 级 2016级软件工程1班

小组名称 第2组

指导老师 林丁报

**计算机科学与工程系**

2019年10月17日

1. **实践项目名称**

图书信息管理系统

**二、实践目的及任务要求**

1、实践目的

课程设计围绕课程的整体教学目标进行，具有一定的综合性和应用性，只靠个别单元的知识一般不能实现课程设计的目标，需要对课程知识进行综合运用。要完成好课程设计，就要对课程知识进行系统学习、复习、总结、应用，达到在实践中学习知识、掌握知识、应用知识和培养能力的目的。

①通过课程设计，进一步熟悉掌握Java语言的基本知识点。

②通过课程设计，达到系统理解、综合运用课程知识的学习目标。

③通过课程设计，学会应用Java语言程序解诀实际问题的方法。

④通过课程设计，掌握程序的局部测试、调试方法，建立程序系统调试、测试的基本概念和思想，学会较大程序的系统测试、调试方法。

通过使用java编程语言实现小型程序项目，巩固对java语言理论知识--面向对象、swing、2d graphic的理解，提升代码编写、bug调试能力。

2、任务要求

要求：使用图形用户界面用数据库建立1或2个表。（不限使用哪种数据库）能连接数据库并实现图书管理相关功能。

**三、实践环境**

编程语言：JAVA。

开发环境：Eclipse，JDK。

**四、实践内容**

1、需求分析

（1）功能分析

（a）超级管理员用户：需要能够添加普通用户，能够对图书进行管理，能够查询用户借书记录；

（b）普通用户：需要能够搜索图书并执行借还书操作。

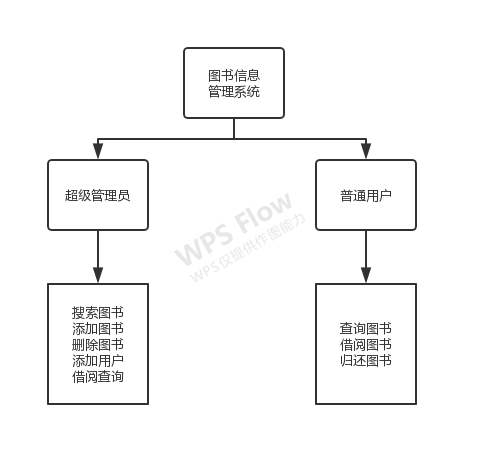


图4-1 功能结构图

（2）非功能性分析

（a）源代码已运行，版面清晰，语言流畅易懂，各模块的分析详细准确，具有极强的可移植性；

（b）代码规范，含丰富注释，具有易懂性。

（c）管理系统版面简洁，可操作性强，拥有丰富的界面交互功能，能够极大的方便用户与管理员完成图书管理的相关操作；

（d）除上述优点外，代码具有健壮性和可维护性。

2、系统设计

（1）登录界面：操作者输入对应的登录账户和密码，进入对应的操作界面。

（2）超级管理员界面：提供查询图书功能，在操作成功后显示查询页面；提供添加图书功能，在添加图书界面填写相关图书信息，添加成功则弹出提示“添加成功”，反之弹出提示“添加失败”；提供删除图书功能，在删除图书界面填写相关图书信息，添加成功则弹出提示“删除成功”，反之弹出提示“删除失败”；提供添加用户功能，在添加用户界面填写用户相关信息，添加成功则弹出提示“添加成功”，反之弹出提示“添加失败”；提供借阅查询功能，在新页面展示所有借阅记录，可根据条件进行相应的查询。

（3）普通用户界面：主页面展示用户个人信息；点击“我要借书”进入借阅面板，可根据条件查询所需图书，单击图书后点击“借书”，若借书成功则弹出提示“借书成功”，反之弹出“借书失败”；点击“我要还书”进入归还面板，单击图书后点击“还书”，若还书成功则弹出提示“还书成功”，反之弹出“还书失败”。

（4）退出系统：点击主界面右上角的叉号即可退出管理系统。

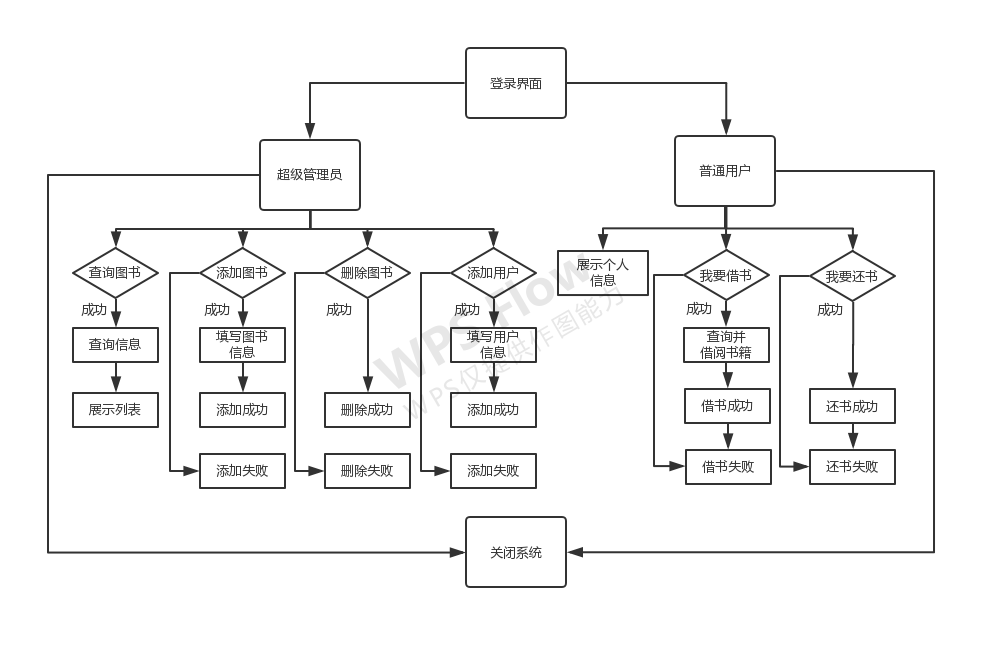


图4-2 总体流程图

3、系统实现

（1）登录验证

输入对应的账户密码后从数据库后台查询相应的用户信息，验证是否匹配，若不匹配则提示“用户名或密码错误”。图4-3为登录界面，图4-4为验证流程图。

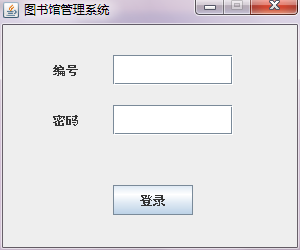


图4-3 登录界面

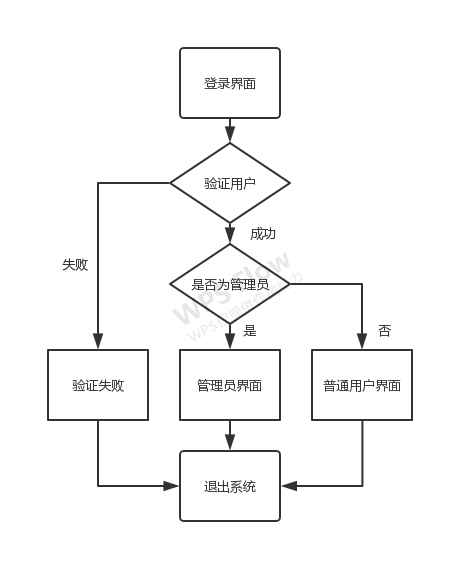


图4-4 随机数产生流程图

关键代码如下：

**public** **static** **int** checkUser(String id, String password) {

Connection conn = JDBCUtils.*getConnection*();

String sql = "select user\_pass from users where user\_id=?";

**try** {

PreparedStatement preps = conn.prepareStatement(sql);

preps.setString(1, id);

ResultSet rs = preps.executeQuery();

**if**(rs.next()) {

**if**(rs.getString(1).equals(password)) {// 验证通过

**if**(id.equals("admin")) {// 是管理员用户

**return** 2;

} **else** {// 是学生用户

**return** 1;

}

}

}

} **catch** (SQLException e) {

e.printStackTrace();

}

**return** 0;

}

（2）超级管理员界面

登录超级管理员界面，可执行图书和用户相关操作。



图4-5 超级管理员界面

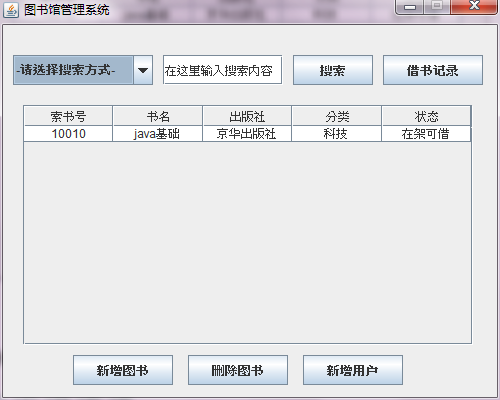


图4-6 搜索结果

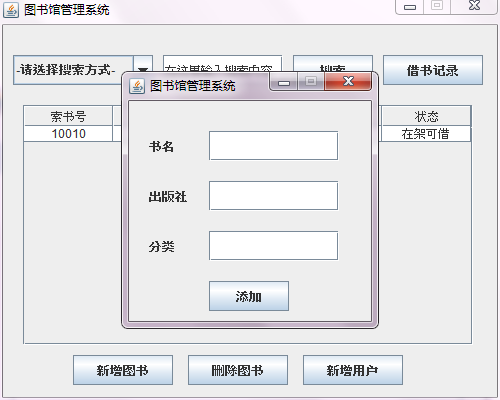


图4-7 添加图书

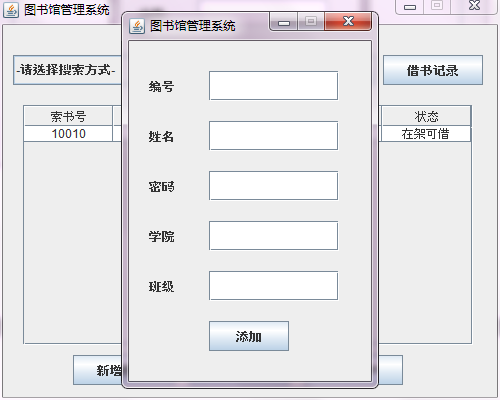


图4-8 新增用户



图4-9 借书记录

关键代码如下：

// 添加书籍

**public** **static** **boolean** addBook(String book\_name, String book\_publisher, String book\_classify) {

Connection conn = JDBCUtils.*getConnection*();

String sql = "insert into books (book\_name, book\_publisher, book\_classify, book\_status) "

+ "values (?, ?, ?, 1)";

**try** {

PreparedStatement preps = conn.prepareStatement(sql);

preps.setString(1, book\_name);

preps.setString(2, book\_publisher);

preps.setString(3, book\_classify);

**if**(!preps.execute()) {

JDBCUtils.*release*(conn, preps);

**return** **true**;

}

} **catch** (SQLException e) {

e.printStackTrace();

}

**return** **false**;

}

// 删除书籍

**public** **static** **boolean** deleteBook(**int** id) {

Connection conn = JDBCUtils.*getConnection*();

String sql = "delete from books where book\_id=?";

**try** {

PreparedStatement preps = conn.prepareStatement(sql);

preps.setInt(1, id);

**if**(!preps.execute()) {

JDBCUtils.*release*(conn, preps);

**return** **true**;

}

} **catch** (SQLException e) {

e.printStackTrace();

}

**return** **false**;

}

// 新增用户

**public** **static** **boolean** addUser(String id, String name, String pass, String depart, String clazz) {

Connection conn = JDBCUtils.*getConnection*();

String sql = "insert into users (user\_id, user\_name, user\_pass, user\_depart, user\_class) "

+ "values (?, ?, ?, ?, ?)";

**try** {

PreparedStatement preps = conn.prepareStatement(sql);

preps.setString(1, id);

preps.setString(2, name);

preps.setString(3, MD5Utils.*md5*(pass));

preps.setString(4, depart);

preps.setString(5, clazz);

**if**(!preps.execute()) {

JDBCUtils.*release*(conn, preps);

**return** **true**;

}

} **catch** (SQLException e) {

e.printStackTrace();

}

**return** **false**;

}

1. 普通用户界面

显示用户个人信息，可执行借阅、归还图书操作。

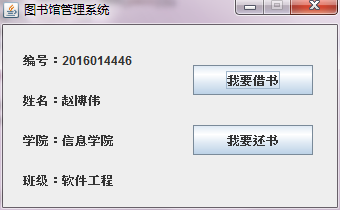


图4-10 普通用户界面



图4-11 借阅界面



图4-11 归还界面

关键代码如下：

// 完成借书操作

**public** **static** **boolean** borrowBook(**int** book\_id, String user\_id) {

Connection conn = JDBCUtils.*getConnection*();

Calendar cal = Calendar.*getInstance*();

**int** y = cal.get(Calendar.***YEAR***);

**int** m = cal.get(Calendar.***MONTH***);

**int** d = cal.get(Calendar.***DATE***);

**int** h = cal.get(Calendar.***HOUR\_OF\_DAY***);

**int** mi = cal.get(Calendar.***MINUTE***);

**int** s = cal.get(Calendar.***SECOND***);

String date = "" + y + "-" + m + "-" + d + "-" + h + "-" + mi + "-" + s;// 拼装时间

String sql = "insert into borrows (borrow\_id, user\_id, book\_id, borrow\_date, return\_date) VALUES (null, ?, ?, ?, '')";// 执行插入操作

**try** {

PreparedStatement preps = conn.prepareStatement(sql);

preps.setString(1, user\_id);// 设置对应的?的值

preps.setInt(2, book\_id);

preps.setString(3, date);

System.***out***.println(preps.toString());// 打印sql语句

preps.execute();// 执行插入

sql = "update books set book\_status=0 where book\_id=?";// 更新书籍状态

preps = conn.prepareStatement(sql);

preps.setInt(1, book\_id);

preps.executeUpdate();

**return** **true**;

} **catch** (SQLException e) {

e.printStackTrace();

}

**return** **false**;

}

// 完成还书操作

**public** **static** **boolean** returnBook(**int** book\_id, String user\_id) {

Connection conn = JDBCUtils.*getConnection*();

Calendar cal = Calendar.*getInstance*();

**int** y = cal.get(Calendar.***YEAR***);

**int** m = cal.get(Calendar.***MONTH***);

**int** d = cal.get(Calendar.***DATE***);

**int** h = cal.get(Calendar.***HOUR\_OF\_DAY***);

**int** mi = cal.get(Calendar.***MINUTE***);

**int** s = cal.get(Calendar.***SECOND***);

String date = "" + y + "-" + m + "-" + d + "-" + h + "-" + mi + "-" + s;

String sql = "update borrows set return\_date=? where book\_id=? and user\_id=? and return\_date=''";// 执行更新操作

**try** {

PreparedStatement preps = conn.prepareStatement(sql);

preps.setString(1, date);

preps.setInt(2, book\_id);

preps.setString(3, user\_id);

System.***out***.println(preps.toString());

preps.executeUpdate();

sql = "update books set book\_status=1 where book\_id=?";// 更新书籍状态

preps = conn.prepareStatement(sql);

preps.setInt(1, book\_id);

preps.executeUpdate();

**return** **true**;

} **catch** (SQLException e) {

e.printStackTrace();

}

**return** **false**;

}

// 查找当前用户所有未还图书

**public** **static** Object[][] searchUserAllUnreturnBook(String user\_id) {

Connection conn = JDBCUtils.*getConnection*();

String sql = "select borrow\_id, book\_id, borrow\_date from borrows where user\_id=? and return\_date=''";

**try** {

PreparedStatement preps = conn.prepareStatement(sql);

preps.setString(1, user\_id);

ResultSet rs = preps.executeQuery();

**return** RS2Object.*rs2Object2*(rs);

} **catch** (SQLException e) {

e.printStackTrace();

}

**return** **null**;

}

**五、实践总结**

通过选择图书信息管理系统这个题目，让我们掌握Java语言的用处，并且可以灵活的使用Eclipse软件。同时更加深入的了解图形化界面的编程过程，使我们更加熟练的掌握Java语言，代码的规范性和完整性有了很大提高，同时对相应代码的模块化也有了相当的认识。

在分析阶段，要进行面向对象的需求分析、系统设计，列出相应的需求文档、绘制系统设计模型图，这些工作很好的锻炼了我们的逻辑思维能力，同时，我们也学到了很多有关于软件管理方面的知识。在这次实践中，我们都有很大的收获，这为我们未来更好的学习打下了牢固的基础。

**六、评审意见及成绩**

评审意见及成绩详见课程实践评审表。