

**图书管理系统**

**测试文档**

**PK项目成员：**

**褚辉**

**张锐**

**2015年5月**

目录

[引言 2](#_Toc420582352)

[1.1编写目的 2](#_Toc420582353)

[1.2 背景 2](#_Toc420582354)

[1.3定义 2](#_Toc420582355)

[1.4 参考资料 2](#_Toc420582356)

[1. 系统需求概述 3](#_Toc420582357)

[2.1业务目标 3](#_Toc420582358)

[2.2运行环境 3](#_Toc420582359)

[2.3功能需求 3](#_Toc420582360)

[2.4 非功能需求 3](#_Toc420582361)

[3. 测试计划 3](#_Toc420582362)

[3.1 测试机构及人员 3](#_Toc420582363)

[3.2 测试方法 4](#_Toc420582364)

[3.3 测试的功能范围 4](#_Toc420582365)

[1. 能够图书管理。 4](#_Toc420582366)

[2.能够借阅管理 4](#_Toc420582367)

[3.能够账户管理 4](#_Toc420582368)

[**4.单元测试报告** 4](#_Toc420582369)

[4.1 图书管理模块 4](#_Toc420582370)

[4.1.1描述 4](#_Toc420582371)

[4.1.2测试方法 4](#_Toc420582372)

[4.1.3测试用例 4](#_Toc420582373)

[4.1.4测试结果截图与分析 5](#_Toc420582374)

[4.2借阅管理模块 7](#_Toc420582375)

[4.2.1描述 7](#_Toc420582376)

[4.2.2测试方法 8](#_Toc420582377)

[4.2.3测试用例 8](#_Toc420582378)

[4.2.4测试结果截图与分析 8](#_Toc420582379)

[4.3 账户管理 9](#_Toc420582380)

[4.3.1描述 9](#_Toc420582381)

[4.3.2测试方法 9](#_Toc420582382)

[4.3.3测试用例 9](#_Toc420582383)

[4.3.4测试结果截图分析 9](#_Toc420582384)

[1. 集成测试报告 11](#_Toc420582385)

[5.1集成测试描述 11](#_Toc420582386)

[5.2 集成测试方法设计 11](#_Toc420582387)

[5.3 测试内容 11](#_Toc420582388)

[5.4 测试结果分析 12](#_Toc420582389)

[6. 评价 12](#_Toc420582390)

[6.1 软件能力 12](#_Toc420582391)

[6.2 限制和缺陷 12](#_Toc420582392)

[6.3 测试结论 12](#_Toc420582393)

# 引言

## 1.1编写目的

检验系统基本功能的实现情况,和容错性。尽最大努力，发现代码以及逻辑上的尽可能多的错误。查找出导致错误的原因，并最终改正这些错误。尽量保证系统正确，流畅的运行。

## 1.2 背景

本软件分为多个模块，每个模块完成后都会进行一次小型的、较为严谨的测试来发现并且改正错误。在整个工程完成之后，再对整个系统进行一次详细完整的测试。

## 1.3定义

系统共分图书管理、借阅管理、账户管理、图书查看等模块。首先读这些模块分别进行测试，最后进行总体测试。

## 1.4 参考资料

《软件工程（第二版）》 张海潘

《面向对象设计UML实践（第二版）》  (英)Mark Priestley著

《UML 2.0和统一过程》 [美]Jim Arlow lla Neustadt著

# 系统需求概述

## 2.1业务目标

系统共分图书管理、借阅管理、账户管理、图书查看等模块。

## 2.2运行环境

Windows xp以上版本系统，Windows 7为最佳

SQL Server 2005 SP2以上版本

JAVA虚拟机：JDK1.4或更高版本

MDAC版本在2.6以上

## 2.3功能需求

图书管理、借阅管理、账户管理、图书查看。

在界面上实现图书的增删改查。

## 2.4 非功能需求

系统应该占有较小内存，运行流畅，无重大bug，内部算法简单高效，外观简洁漂亮，让用户使用时感觉容易上手 ，感觉顺心愉悦。系统要求具有可重用性，系统的组件需要可以被重复使用。

# 3. 测试计划

## 3.1 测试机构及人员

表3-1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **测试内容** | **数据库** | **中间层** | **用户界面** | **集成测试** |
| 测试人 | 张锐 | 全体组员 | 全体组员 | 褚辉 |

## 3.2 测试方法

1.通过对界面的操作验证功能是否完整、正确。

2.通过编写测试类来验证内部函数是否正确工作，数据能否正常存储。

## 3.3 测试的功能范围

## 能够图书管理。

## 2.能够借阅管理

## 3.能够账户管理

**4.单元测试报告**

## 4.1 图书管理模块

### 4.1.1描述

在主界面上进行书籍的增删改查。

### 4.1.2测试方法

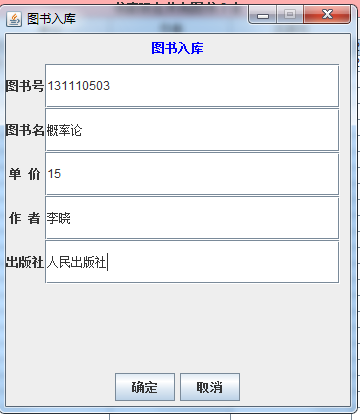
通过操作界面，进行测试。

### 4.1.3测试用例

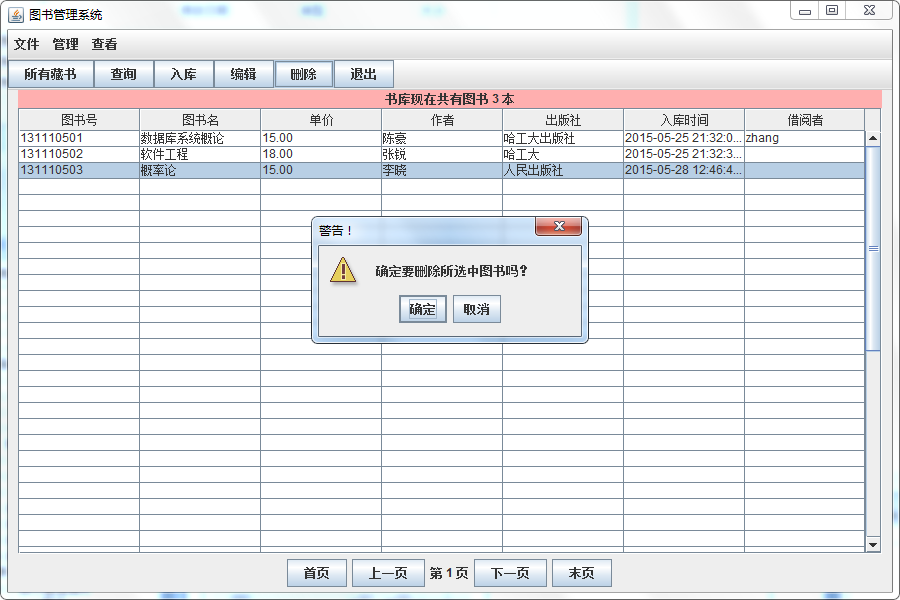
1. 选择图书入库。
2. 选择删除图书。
3. 编辑图书。
4. 查询图书。

### 4.1.4测试结果截图与分析

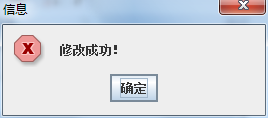
1. 图书入库



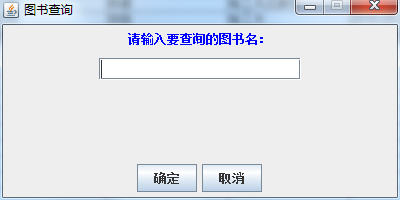
1. 图书删除。



3.图书编辑



4.图书查询





## 4.2借阅管理模块

### 4.2.1描述

查找书籍，输入读者号，借阅成功。

输入书号。还书成功。

### 4.2.2测试方法

通过操作界面，进行借阅，还书。

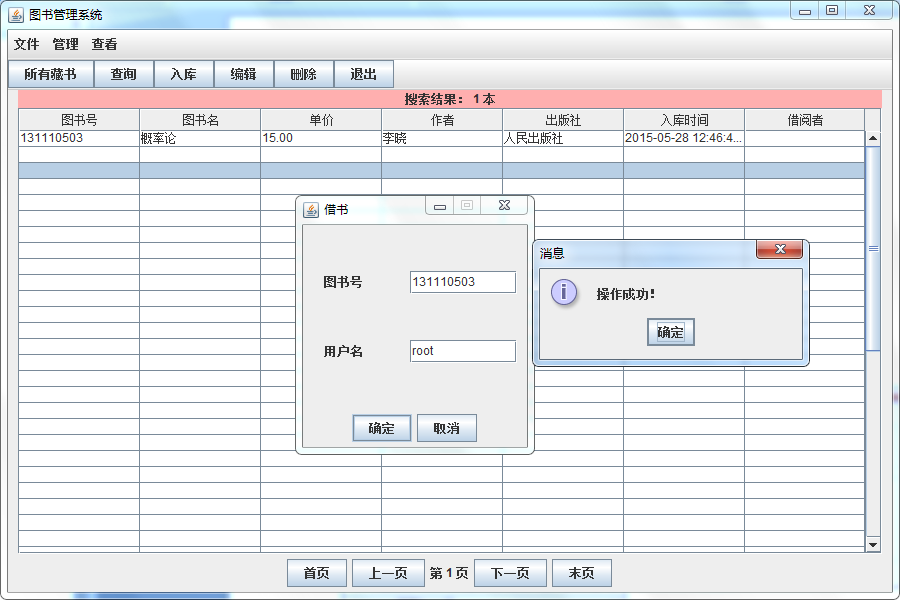
### 4.2.3测试用例

选择借阅功能，进行借阅。

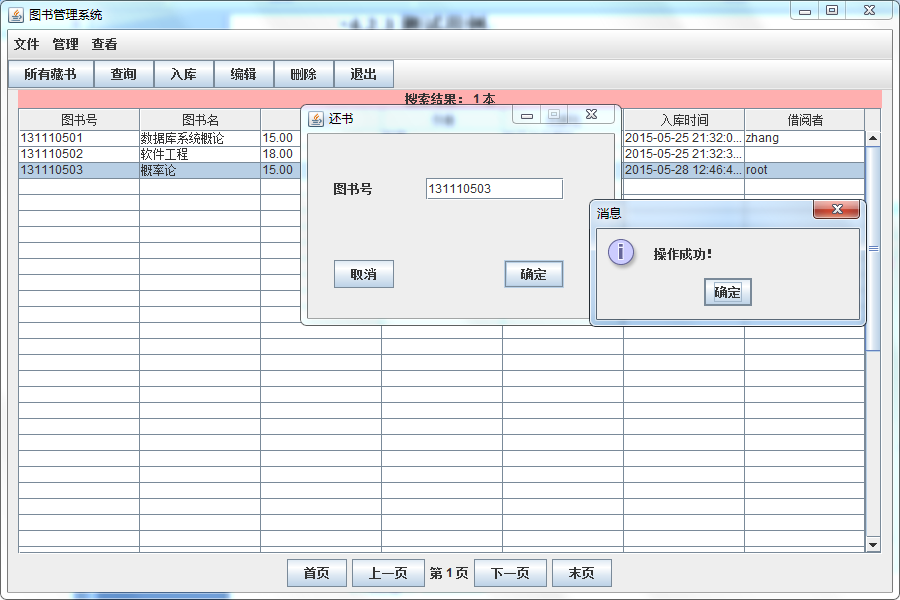
选择还书，进行归还。

### 4.2.4测试结果截图与分析

1. 将书号131110503的图书借给root用户



2.归还图书131110503



## 4.3 账户管理

### 4.3.1描述

能够增加新的读者和管理员

### 4.3.2测试方法

通过账户管理增加账户

### 4.3.3测试用例

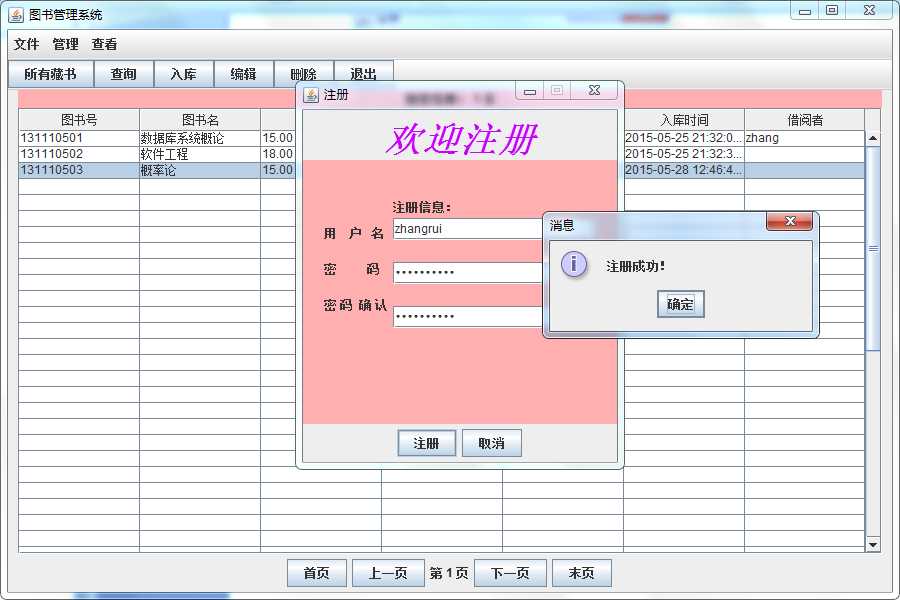
通过账户管理增加账户

### 4.3.4测试结果截图分析

1. 注册读者



2.增加管理员



# 集成测试报告

# 5.1集成测试描述

集成测试是对系统的综合测试，主要是对界面的测试。

## 5.2 集成测试方法设计

集成测试采用自顶向下的测试方法：界面—>中间层—>数据处理层。

## 5.3 测试内容

**评价指标**：

正确性：是/否

操作性：优/良/差

结果如下：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **测试单元** | **包含功能** | **实际结果** | **正确性** | **操作性** |
| 图书管理 | 图书的增删改查 | 操作成功 | 是 | 优 |
| 借阅管理 | 借书、还书 | 操作成功 | 是 | 优 |
| 账户管理 | 账户的增加删除 | 操作成功 | 是 | 优 |

## 5.4 测试结果分析

统集成测试系统总体功能稳定，未出现重大的功能上的问题，但需要修改一些细节的地方，提高系统的容错性、可操作性和通过改进界面提高用户的用户体验。

# 6. 评价

## 6.1 软件能力

|  |  |  |
| --- | --- | --- |
| **功能及性能** | **要求** | **测试结果** |
| 基本功能 | 图书管理、借阅管理、账户管理。 | 上述功能全部实现，测试通过。 |
| 附加功能 | 无 | 无 |
| 容错性 | 具有良好的容错性，输入错误，以及操作错误时给出警告。 | 容错性一般 |
| 可操作性 | 操作简单，易于掌握。 | 功能齐全，操作简便 |
| 效率与安全 | 响应迅速，代码安全高效 | 响应迅速，代码较为合理 |

## 6.2 限制和缺陷

1.功能相对较简单。

2.不能批量导入图书，导致距离实用还有一段距离。

3.界面美化较少，图标不够美观，

## 6.3 测试结论

就项目的总体功能和外观上都已满足需求分析的要求，界面友好，运行流畅无bug的目的，并且拥有相当的容错性，保障了数据的安全性。