**浙江大学实验报告**

课程名称： 数据库系统原理 实验类型： JDBC编程

实验项目名称： 图书管理系统

学生姓名1： 专业1： 学号1：

学生姓名2： 专业2： 学号2：

指导老师： 陈岭

实验地点： 实验日期： 年 月 日

1. **实验目的和要求**

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. 实验要求

(1) 管理基本数据对象

(2) 实现基本功能模块

(3) 实现用户界面

完成一个交互式应用界面，可采用图形界面或字符界面。

(4) 数据库平台

SQL Server或MySQL

其中MySql详细信息请参见 [http://www.mysql.com](http://www.mysql.com/)

MySql APIs:

1. MySql ODBC 3.51
2. MySql JDBC 5.0
3. MySql PHP APIs

有关MySql的安装，请参看有关参考书。

(5) 开发工具

任选（如VC++, PHP, Java, Delphi, PowerBuilder等）

1. **实验内容和原理**

1. 数据对象管理

|  |  |
| --- | --- |
| 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 (教师 学生等) |
| 管理员 | 管理员ID, 密码, 姓名, 联系方式 |
| 借书记录 | 卡号, 借书证号 ,借期, 还期, 经手人 (管理员ID) |

四个数据对象分别要在数据库端和应用程序端控制。在数据库端可以分别以4张表的形式存在，在应用程序端以四个类的形式组织。通过JDBC接口，可以让数据在前后端的两种数据结构间互相传输。

2. 登录系统与用户管理

(1) 登录系统

将帐号组织为管理员和普通借书证用户，把所有帐号信息存储在数据库端中。验证登录权限的时候，从数据库端中扒取账户信息，通过与前端输入的账户信息对比控制登录状态。

(2) 用户管理

在客户端中实现普通的借书证用户的添加和删除控制以及借书证用户信息查询功能。可以通过JDBC实现对数据库中用户表的一些基本查询和操作来完成。

3. 图书入库/图书查询/借书/还书

|  |  |
| --- | --- |
| 模块名称 | 功能描述 |
| 图书入库 | 1. 单本入库 2. 批量入库 (方便最后测试)   图书信息存放在文件中, 每条图书信息为一行. 一行中的内容如下  ( 书号, 类别, 书名, 出版社, 年份, 作者, 价格, 数量 )  Note: 其中 年份、数量是整数类型; 价格是两位小数类型; 其余为字符串类型  Sample:  ( book\_no\_1, Computer Science, Computer Architecture, xxx, 2004, xxx, 90.00, 2 ) |
| 图书查询 | 要求可以对书的 类别, 书名, 出版社, 年份(年份区间), 作者, 价格(区间) 进行查询. 每条图书信息包括以下内容:  ( 书号, 类别, 书名, 出版社, 年份, 作者, 价格, 总藏书量, 库存 )  可选要求: 可以按用户指定属性对图书信息进行排序. (默认是书名) |
| 借书 | 1.输入借书证卡号  显示该借书证所有已借书籍 (返回, 格式同查询模块)  2.输入书号  如果该书还有库存，则借书成功，同时库存数减一。  否则输出该书无库存，且输出最近归还的时间。 |
| 还书 | 1.输入借书证卡号  显示该借书证所有已借书籍 (返回, 格式同查询模块)  2.输入书号  如果该书在已借书籍列表内, 则还书成功, 同时库存加一.  否则输出出错信息. |

4. 用户界面

通过常用的流行GUI设计手段，构建一个图形化的交互界面。应用程序处理前端的交互逻辑并通过JDBC访问数据库，最后将结果反馈给前端界面，完成交互。

1. **主要仪器设备**

MySQL数据库系统

Java8 SE

JavaFX + JavaFX Scene Builder

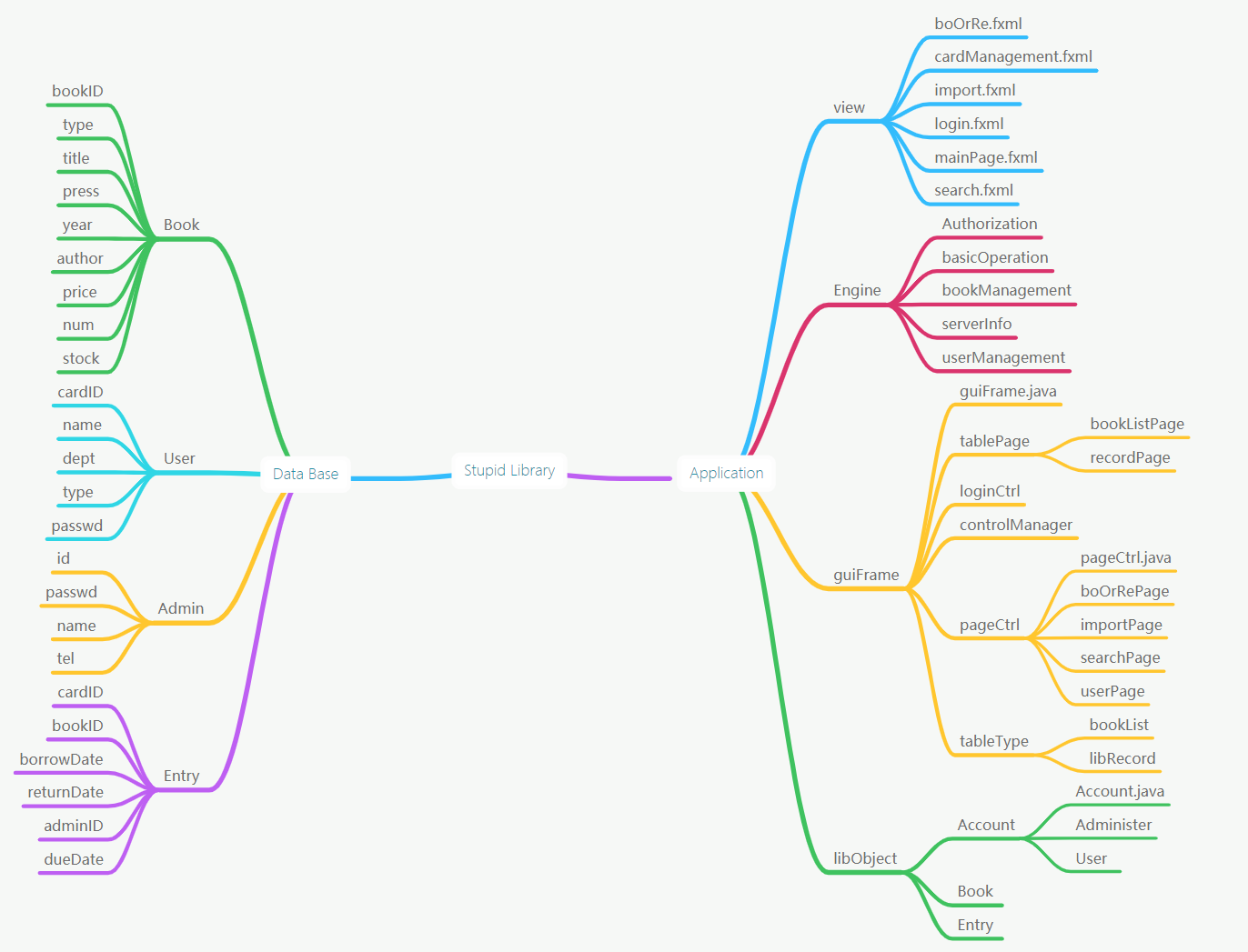
MySQL-JDBC-8.0.15

Mac OS / Windows 10

1. **操作方法与实验步骤**

1. 项目架构

(1)总览



如上所示，整个图书管理系统由数据库引擎后端与应用程序前端两部分组成。所有数据表都由Oracle MySQL 数据库引擎托管，由Java构建的应用程序端可以通过JDBC向数据库引擎发送各种SQL命令并从数据库系统中获得各种类型的返回数据。最后用户再通过GUI界面和应用程序端交互完成各种图书管理功能。

(2)数据库端

数据库端采用了Orcale MySQL系统。由于整个图书管理系统只要求对四类数据对象进行管理。所以整个后端的数据库系统中只有四张表。

数据表定义如下：

create table admin

(

id varchar(45),

passwd varchar(45),

name varchar(45),

tel varchar(45),

primary key(id)

)

create table Book

(

bookID varchar(45),

type varchar(45),

title varchar(45),

press varchar(45),

year int(11),

author varchar(45),

price decimal(18,2),

num int(11),

stock int(11),

primary key(bookID)

)

create table entry

(

cardID varchar(45),

bookID varchar(45),

borrowDate varchar(45),

returnDate varchar(45),

adminID varchar(45),

dueDate varchar(45),

primary key(cardID,bookID,borrowDate)

)

create table user

(

cardID varchar(45),

name varchar(45),

dept varchar(45),

type varchar(45),

primary key(cardID)

)

(3)应用程序端

整个应用程序端可以分为四个主要的大模块。分别是guiFrame，view，libObject，Engine。

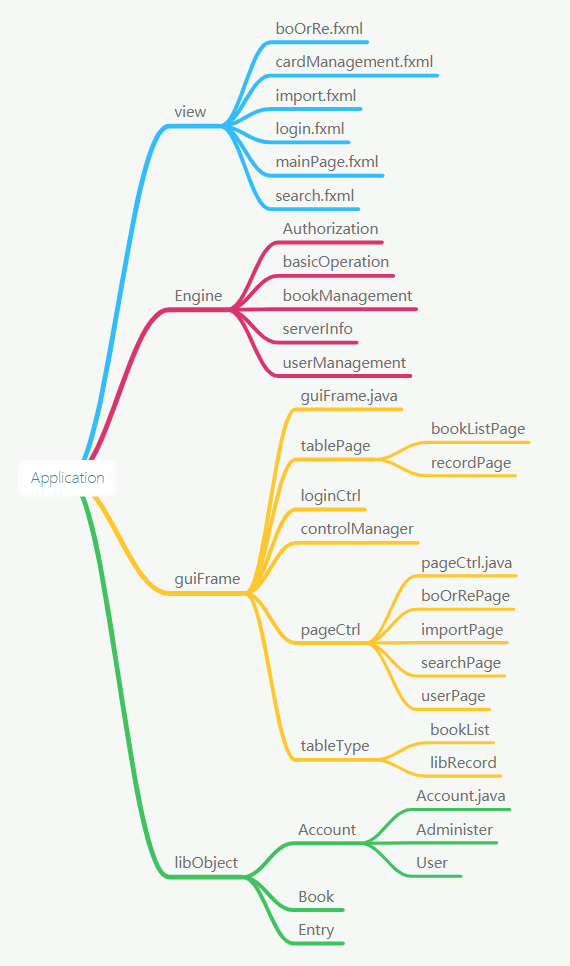
其中view主要是一些GUI的素材资源和前端布局的fxml文件，主要完成了GUI界面控件命名与布局，事件绑定等工作。整个应用程序主要做了6个静态的图形页面，分别用来管理登陆，主页欢迎页面，借还书，导入书籍，用户管理和书籍搜索。这部分内容都在JavaFX Scene builder 中通过可视化编辑的手段完成了。

libObject中定义了四个数据对象Book, Entry, Administer, User。它们以类的形式组织起来，作为图书管理系统中四个基本对象在应用端的数据结构。

guiFrame主要是前端各个页面交互事件的控制器。loginCtrl用来控制登陆界面的逻辑，pageCtrl用来控制各个功能界面的事务。controlManager负责界面切换，权限控制逻辑。tableType和tablePage主要用于将查询的结果表格进行呈现。

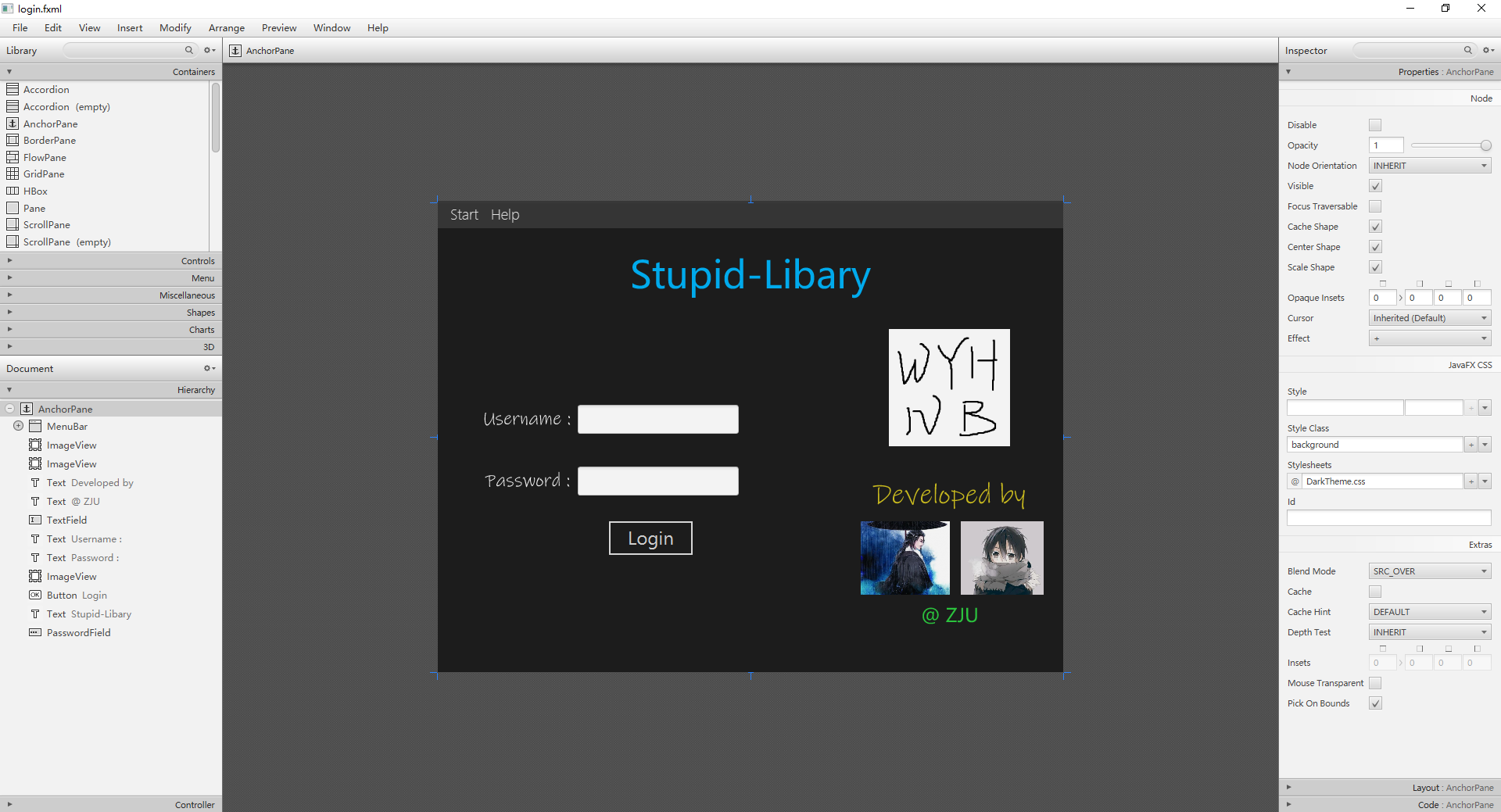
Engine模块是整个图书管理系统应用层与数据库系统层的信息交流核心。basicOperation中封装了基于JDBC的基本SQL命令操作。serverInfo封装了本地数据库服务器的信息。Authorization主要用来做权限控制，权限管理。userManagement和bookManagement分别负责用户与书籍的数据管理和交互。

应用程序端的基本架构逻辑图如下：

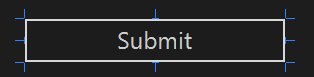


1. GUI设计

本项目的GUI设计完全基于JavaFX技术。

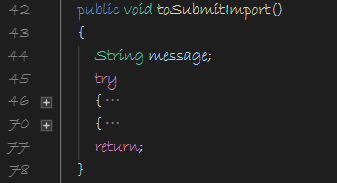


前端的布局和风格可以通过在JavaFX Scene Builder中可视化编辑，并加载流行的css定义表控制。上图是加载了VScode界面样式css表的登陆界面。









如上所示，我们在Scene Builder中编辑了前端图形后，给每一个需要事件响应的控件再绑定对应的监听器/控件ID等属性，自动生成.fxml文件。然后在.fxml文件中再绑定具体的逻辑控制模块。最后在逻辑控制模块中用代码实现事件响应。这也就是JavaFX的基本交互模式。

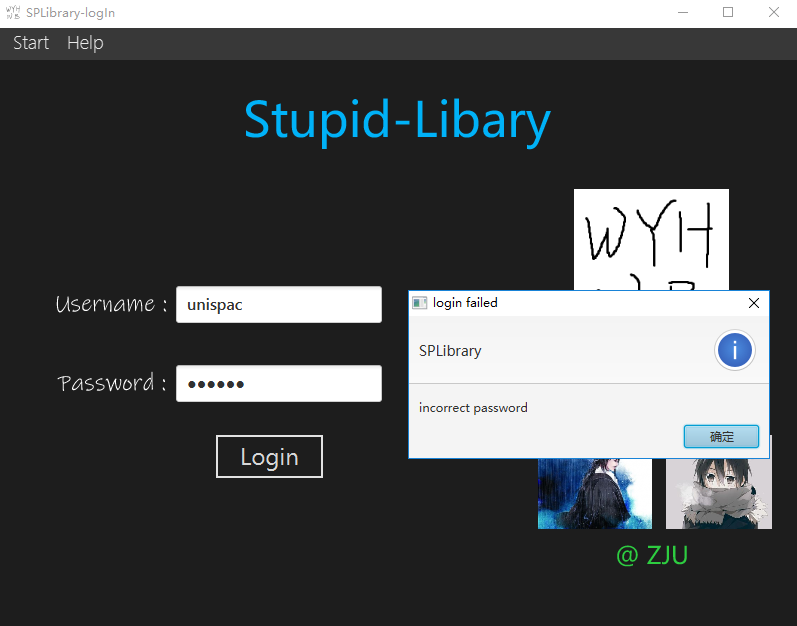
显然，用户的所有请求都是通过GUI界面提供的按钮，输入框等控件送到后台的。如上所述，这些请求会最先送到guiFrame模块的各个controller中，对于涉及到修改和查询数据库信息的请求，controller会调用Engine模块中实现的接口，Engine模块再通过JDBC和数据库交互。最后从数据库返回的信息又会经由Engine传回controller。一些需要反馈给用户的信息，则通过前端的弹窗对话框或者一个弹出的表格界面再呈现给用户。

1. Engine 设计
2. **实验结果与分析**

1. 账号与权限

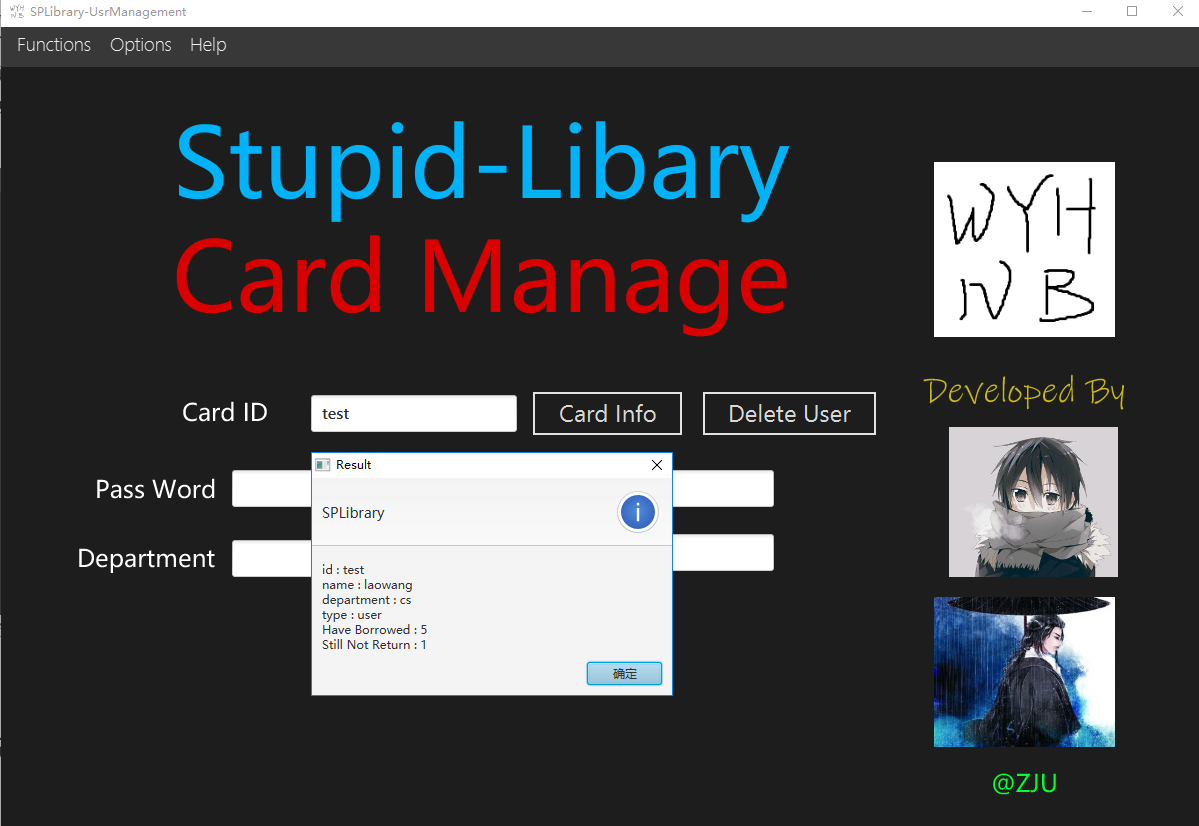
我们用两个账号分别来测试。

一个是管理员账号（ID : unispac，Passwd:996ICU）。另一个是普通的借书证账号（ID : test , Passwd:123456）。



当输入的密码与数据库中的信息不吻合时，登陆就会被阻止。

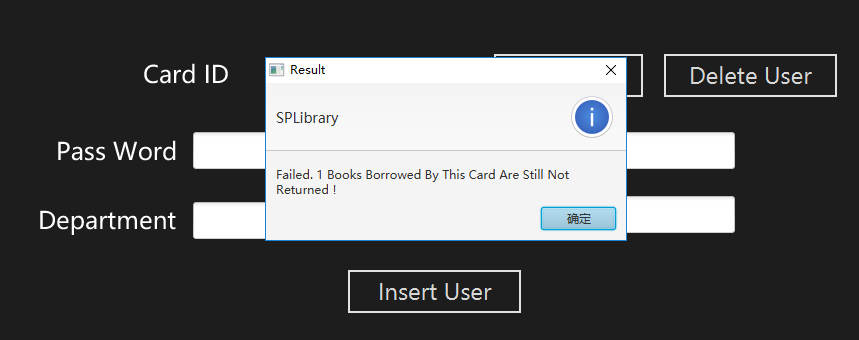
管理员unispac账号信息成功认证后，就能登陆进入系统，并且可以访问所有功能界面。



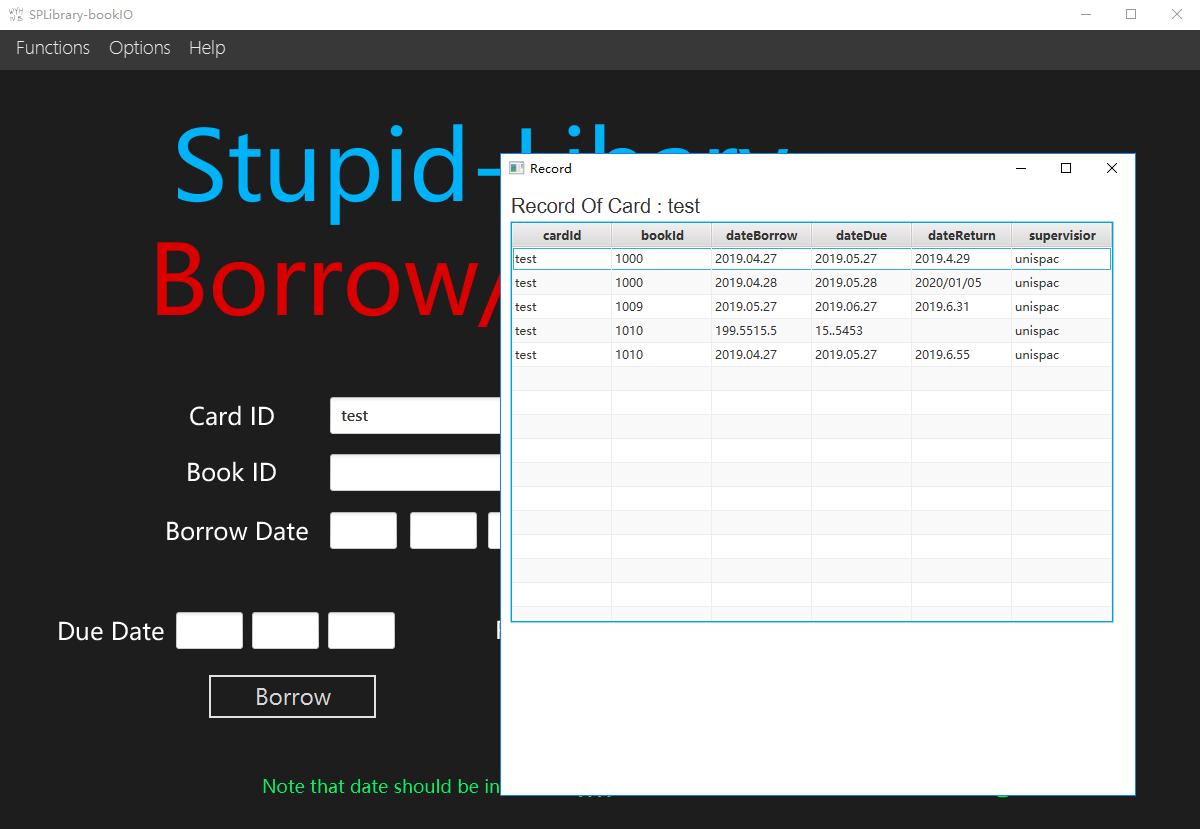
如图，unispac可以查询test账号的信息。弹出对话框，显示账户的所有信息，并且可以看到账户借书记录总共有多少条，是否还有书没有归还。可以看到test还有一本书没有归还。

尝试删除用户test：



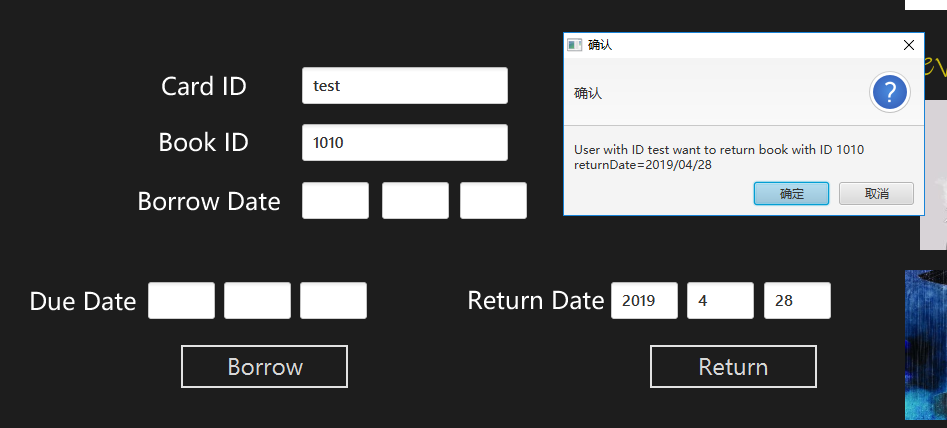


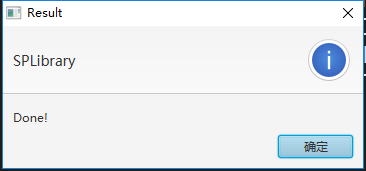
提示删除失败，因为用户还有书没有归还。

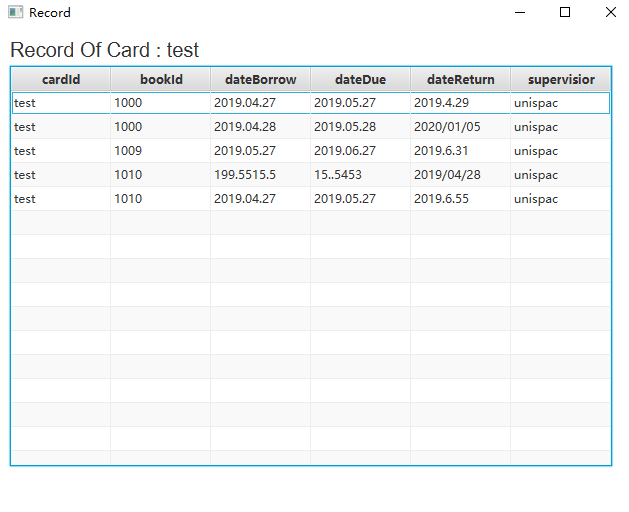


去借书页面查看用户test的借书记录，可以在表格中看到用户的五条借书记录，其中代码为1010的书还没有归还。

尝试还书：







1. **讨论、心得**