

图书管理系统

个人信息

课程名称：	数据库系统概念
姓名：	任青
学号：	3110000258
专业：	计算机科学与技术
日期：	2013 年 4 月 26 日

一、概况

1.1 综述

图书管理系统是通过数据库系统来实现的图书数字化管理的系统，系统包含了图书的查询检索、管理员对图书、借书证（用户）的添加删除修改管理以及对图书的借还图书等功能。通过实践开发图书管理系统，加深对数据库系统原理及应用的理解，亲自动手完成数据库的 E-R 模型设计，规划数据库的关系模型，并通过编程对整套系统加以实现

1.2 开发设计环境

数据库系统	MySQL Server 5.5
数据库运行环境	Windows Server 2003 64 bits Enterprise Edition
应用开发环境	Qt 4.7.4 with MinGW
Qt 与数据库通讯方式	ODBC / MySQL ODBC 5.2w Driver 32 bits
应用界面	基于 GUI 的图形化界面
开发测试环境	Windows 7 64 bits Ultimate

确保在上述环境下，应用可正常开发设计、测试运行。

1.3 实验基本要求

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)

模块名称	功能描述
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	<p>1. 单本入库</p> <p>2. 批量入库 (方便 最后测试)</p> <p>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)</p> <p>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</p> <p>Sample :</p> <p>(book_no_1, Computer Science, Computer Architecture, xxx, 2004, xxx, 90.00, 2)</p>
图书查询	<p>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. 返回最多前五十条符合条件的图书信息, 每条图书信息包括以下内容:</p> <p>(书号, 类别, 书名, 出版社, 年份, 作者, 价格, 总藏书量, 库存)</p> <p>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)</p>
借书	<p>1.输入借书证卡号</p> <p>显示该借书证所有已借书籍 (返回, 格式同查询模块)</p> <p>2.输入书号</p> <p>如果该书还有库存, 则借书成功, 同时库存数减一。</p> <p>否则输出该书无库存, 且输出最近归还的时间。</p>

还书	<p>1.输入借书证卡号</p> <p>显示该借书证所有已借书籍 (返回, 格式同查询模块)</p> <p>2.输入书号</p> <p>如果该书在已借书籍列表内, 则还书成功, 同时库存加一.</p> <p>否则输出出错信息.</p>
借书证管理	增加或删除一个借书证.

二、系统架构

2.1 整体构架

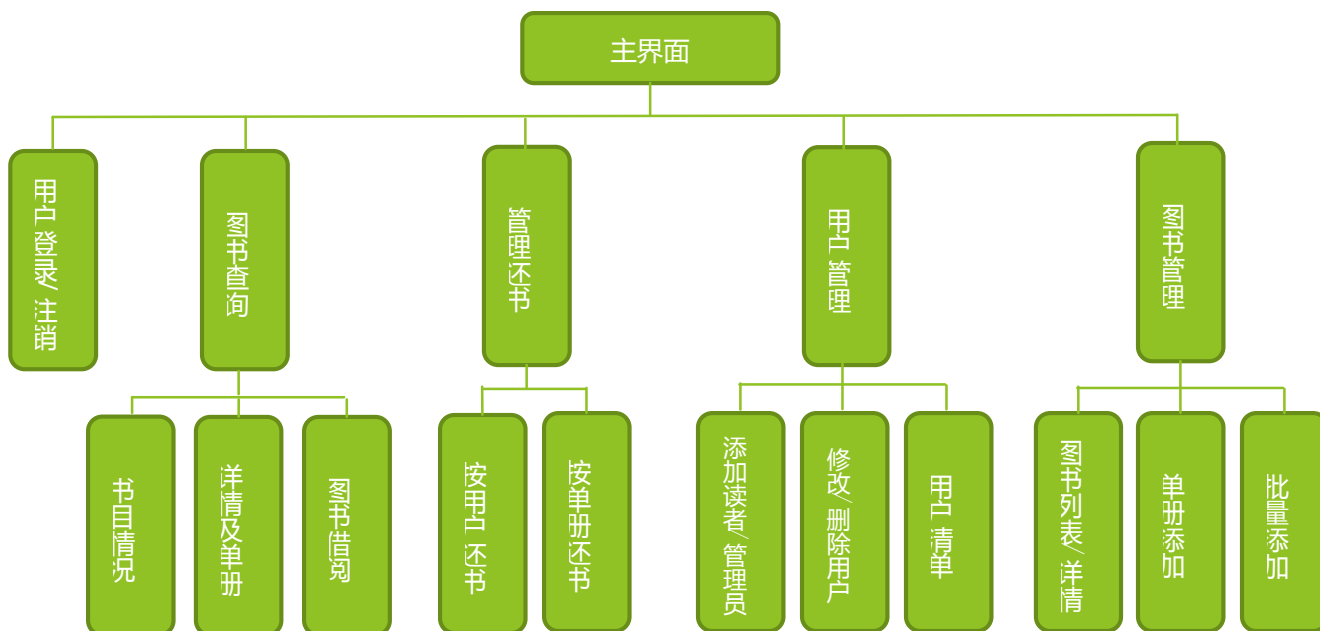
求是图书管理系统在开发的时候采用了 Qt 4.7 的环境，编译器使用 MinGW 进行编译，整体系统可以分为两个部分：用户交互界面（可以视作为 View）、后端处理模型（可以视作为 Model 与 Control 的集合体）。

用户交互界面：这一部分是通过 Qt Creator 的 UI 设计器设计完成，并将用户界面封装为一个类的形式在后端主程序当中对类进行实例化。对于用户 UI 界面与后端主程序通讯的方式，我们可以使用 Qt 的信号槽机制（SIGNAL & SLOT）来实现，对于 UI 界面中的每一个交互操作都会形成一个 SIGNAL，并通过 connect 触发主程序的对应 SLOT 来执行。通过基于 GUI 的 UI 界面可以增强对用户的操作友好性，相比网页应用操作更为流畅便捷，相对于 Console 应用更加直观易用。

后端处理模型：这里负责与数据库的通讯、界面交互以及后端数据处理等工作。在数据库通讯方面我们采用了 QODBC 链接 MySQL ODBC 驱动，在默认情况下 Windows 系统没有安装，需要手动安装 32 位版本的 MySQL ODBC 5.2 Driver，具体下载地址：<http://dev.mysql.com/downloads/connector/odbc/>

界面交互时，我们根据用户的不同操作，来链接数据库获取相应信息并经过对数据的加工处理后返回给用户。

2.2 功能图



三、数据库表设计

为了方便迁移以及便于多个 app 同时共用同一个数据库的情况，我们在所有表的名称上加上前缀 qlms_

数据库编码统一使用：utf-8

数据库表共分为四部分：

- qlms_book [用于储存图书信息]、
- qlms_book_item [用于储存单册图书信息]、
- qlms_record [用于储存借阅记录]、
- qlms_user [用于储存用户信息包含管理员以及普通读者]

3.1 qlms_book

字段名	类型	长度	默认值	允许空	属性	说明
isbn	Int	~			Primary	
Title	Char	50				
Type	Char	50				
Pub_Press	Char	50				
Pub_Year	Int	~				
Author	Char	20				
Price	Double	~				
Num_Total	Int	~				

3.2 qlms_book_item

字段名	类型	长度	默认值	允许空	属性	说明
Id	Int	~			Primary Auto_Incremennt	图书单册编号
Isbn	Int	~			References qlms_book	
Status	Int	~				0 表借出 1 表在馆
Location	Char	20				图书位置

3.3 qlms_record

字段名	类型	长度	默认值	允许空	属性	说明
Uuid	Int	~			Primary Auto_Incremennt	记录唯一 ID

Id	Int	~			References qlms_book_item	单册编号
Stuid	Int	~			References qlms_user	
Status	Int	~				0 表示未还 1 表示已还
Time_Borrow	Timestamp	~	当前时间			借书时间
Time_Return	Timestamp	~	NULL	Yes		还书时间

3.4 qlms_user

字段名	类型	长度	默认值	允许空	属性	说明
Stuid	Int	~			Primary Key	学工号
Password	Char	32				密码，明文
Name	Char	20				姓名
Num_borrowed	Int	~				当前借书量
Num_limit	Int	~				借阅上限
Department	Char	20				所在单位
IsAdmin	Tinyint	~				0 读者 1 管理员

四、详细流程

4.1 用户登入、登出及权限控制

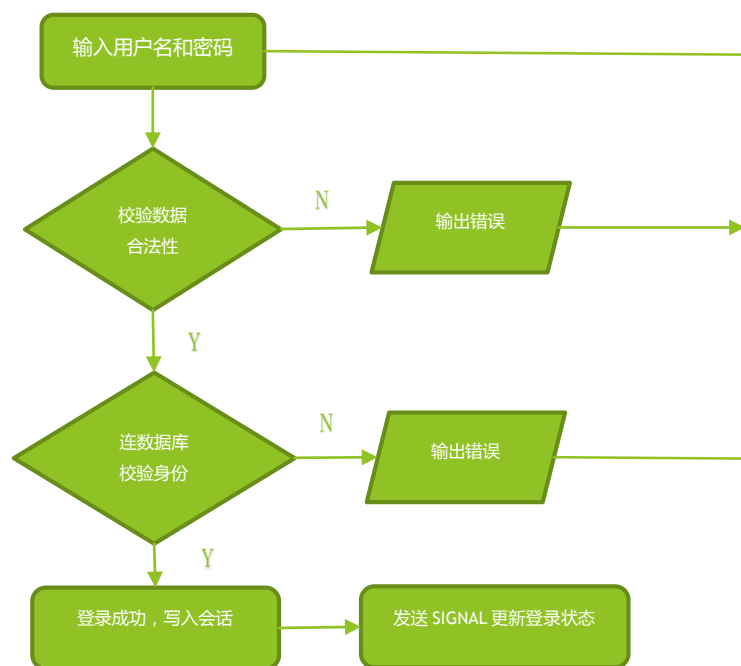
说明

为了区分不同用户的操作权限，图书管理系统需要有权限控制模块，来保障用户不会越权操作。

实现技术

将用户的登录状态、权限、学号以及姓名等信息封装至 QLMS_CLASS 类当中，在 main.cpp 中对 QLMS_CLASS 进行实例化，在其它模块或窗体的.cpp 文件中通过 extern 方式访问 QLMS_CLASS 的实例，来实现对用户当前会话信息的记录操作。

用户登录：用户在输入用户名与密码后通过登录按钮触发后端系统进行校验，首先程序会对用户输入信息的合法性进行校验，在通过校验后会向数据库系统发送 Query 请求，以 stuid 作为索引条件查询 qlms_user 表，若记录存在则比对密码，若不存在则通过 QMessageBox 来返回错误提示信息。在登录成功的情况下，会把用户的当前信息（stuid、isadmin、name、num_limit、num_total 等）信息写入到 QLMS_CLASS 的会话信息当中。并通过 Qt 的 SIGNAL-SLOT 机制调用 on_signal_change_login_status() 函数，根据权限的不同在主窗口内启用相应权限的按钮。

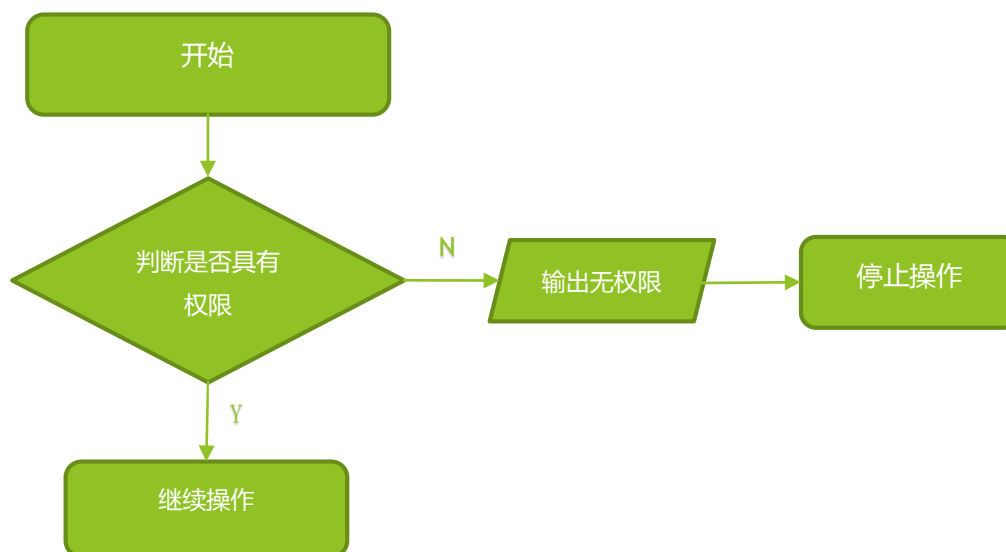


用户登出：在用户点击登出按钮之后会清空 QLMS_CLASS 实例当中的用户相关信息，并通过 Qt 的 SIGNAL-SLOT 机制调用 on_signal_change_login_status()函数来调整当前进行的权限，屏蔽没有权限操作的按钮。

权限校验：

对于判断当前用户是否登录可以直接调用封装在 QLMS_CLASS 的函数 check_isUserLogin()，函数会检查 isLogin，并返回其值。若 check_isUserLogin()返回为 false 的话，则拒绝用户进行需要用户登录才能进行的操作，并强制要求登录。

对于判断当前用户是否具有管理员权限，可以直接调用 QLMS_CLASS 的函数 check_isAdmin()，函数会检查 isAdmin，并返回其值。若 check_isAdmin()返回为 false 的话，则拒绝用户进行需要管理员权限的功能。



功能截图

登录界面：



对权限限制：

从左至右分别是通过管理员、读者及尚未登录的权限树



4.2 图书查询

说明

图书查询功能是图书管理系统的基本功能，通过本功能可以让用户方便的去检索本系统内的所有图书信息。为了更好的对用户友好，我们引入互联网搜索引擎的单搜索框的设计理念，通过一个文本框让用户来检索所有的图书信息（包含 ISBN、图书名称、作者、出版社等），在用户点击具体的搜索结果之后，可以返回本图书的具体图书信息，并返回本图书的所有单册图书在馆及借阅信息。

实现技术

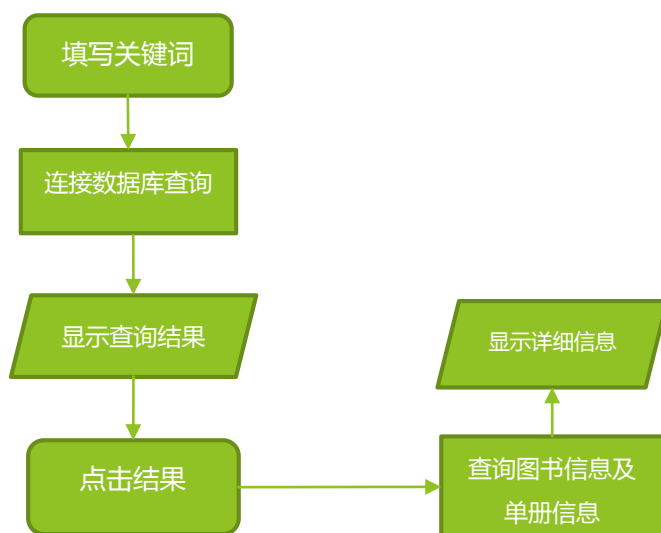
图书检索

当用户填写完关键词并按搜索按钮后，通过在 `book_dialog.cpp` 中的连接数据库并执行 `SELECT` 语句来检索图书，为了实现通过关键词检索多个不同属性，我们在 `WHERE` 语句中对多个进行 `LIKE %KEYWORD%`，并将结果 `OR` 起来以实现检索。通过 `ORDER BY isbn` 的方式，我们将图书按照 ISBN 号从小到达排序，通过 `LIMIT 0,50` 语句我们来实现只查询前 50 条有效记录。

当用户在查询窗口内点击任意单行查询结果之后，我们继续通过 Qt 的 `SIGNAL-SLOT` 机制，在查询窗口发送带有 `isbn` 号的 `SIGNAL` 信号，以调用 `book_item` 窗口，窗口在收到加载信号后会查询图书的详细信息，并查询图书的单册信息及借阅状态，最终显示给读者。

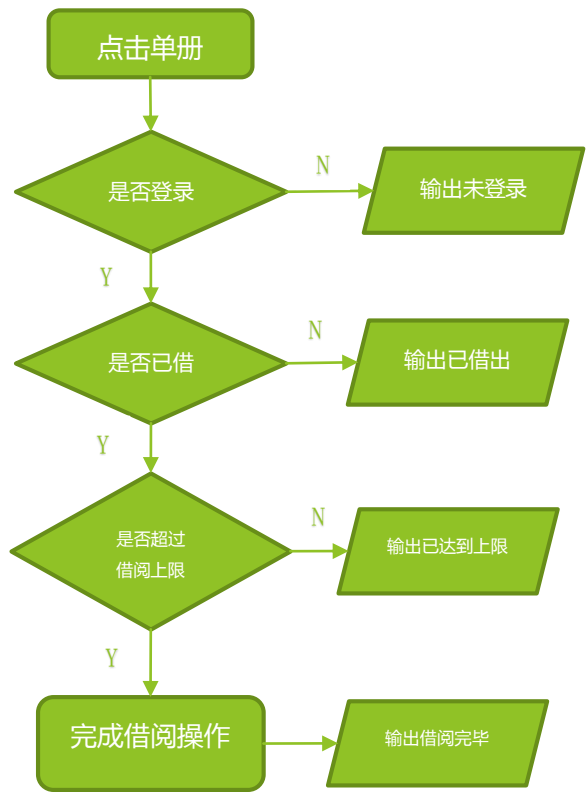
查询单册状态的 SQL 语句：

```
SELECT title, isbn, type, pub_press, pub_year, author, price, num_total FROM qlms_book WHERE isbn = ?
```



图书借阅

当用户在单图书的详情页面，点击单册图书的状态的时候，则可以对图书进行借阅。若图书当前不在馆则返回图书不可借阅，若用户没有登录则不允许借阅，若用户已经打到个人图书借阅上限[即 num_borrowed >= num_limit]的时候，会返回用户借阅图书数量超过上限不允许继续借阅。在所有检测都没有问题之后，会授权用户完成图书借阅操作，这个时候会向 qlms_record 表增加新的借阅记录，并修改 qlms_book_item 单册图书状态和 qlms_user 中当前用户的图书借阅数量。



功能截图
搜索全部内容

图书查询 - 求是图书管理系统

图书查询

关键词:

图书清单

图书ISBN	图书名称	作者	馆藏数量	出版社
2	西窗法雨	刘星	3	院系分馆
21	Computer...	xxx	2	xxx
31	Computer...	xxx	6	xxx
32	编译原理	xxx	9	xxx
33	人工智能1	xxx	2	xxx
34	计算机网络1	xxx	3	xxx
1111	中国文学...	张少康	2	西溪流通书库

以“1”作为关键词搜索

图书查询 - 求是图书管理系统

图书查询

关键词: 1

图书清单

图书ISBN	图书名称	作者	馆藏数量	出版社
21	Computer...	xxx	2	xxx
31	Computer...	xxx	6	xxx
33	人工智能1	xxx	2	xxx
34	计算机网络1	xxx	3	xxx
1111	中国文学...	张少康	2	西溪流通书库

查看图书详情及借阅



4.3 管理还书

说明

还书功能是图书管理系统的一项基本功能，管理员登录之后可以进行还书操作，为了方便操作共有两种还书方式：1）根据用户的学号显示所有他的借阅情况并点击后还书 2）根据单册的图书 ID 来进行还书操作

实现技术

针对学号还书

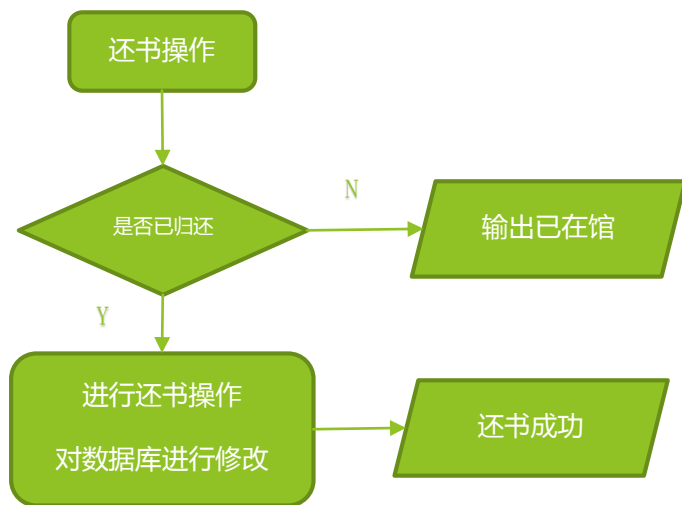
用户在输入学号之后，点击搜索按钮后，主程序会向数据库发送查询请求，查询该用户的所有借阅清单，并以 QTreeView 的方式呈现出来。当管理员单击某一条借阅状态的时候，主程序会取出当前记录的 id，并调用针对单册图书 ID 还书函数进行还书操作。

SQL 查询语句：

```
SELECT qlms_record.id, qlms_book.isbn, qlms_book.title, qlms_record.status, qlms_record.time_borrow,
qlms_record.time_return FROM qlms_record LEFT JOIN qlms_book_item ON qlms_book_item.id = qlms_record.id
LEFT JOIN qlms_book ON qlms_book.isbn = qlms_book_item.isbn WHERE qlms_record.stuid = %1 ORDER BY
qlms_record.status, qlms_record.uuid DESC
```

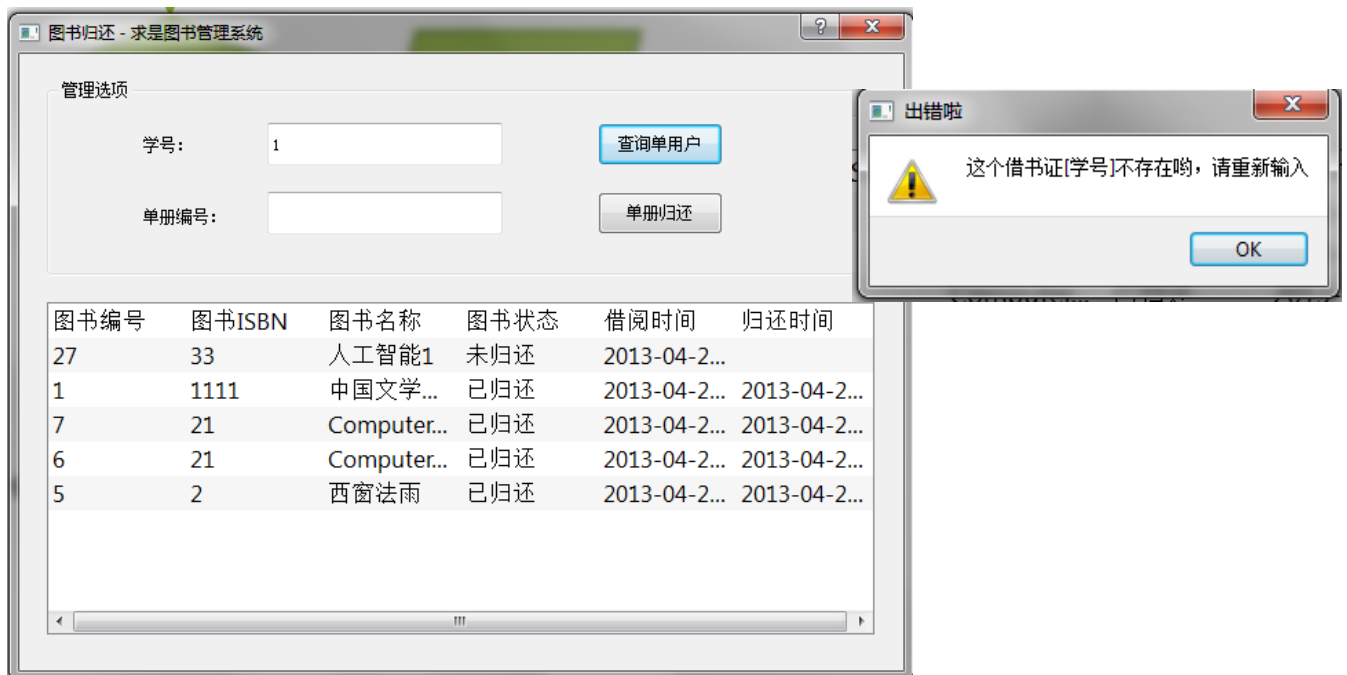
针对单册还书

首先主程序会请求数据库检查，查询当前单册图书的状态，若已经归还则不允许重复操作，否则会将该册图书状态修改为已经归还，并同时修改用户的 num_borrowed 数量以及 qlms_record 的归还日期。

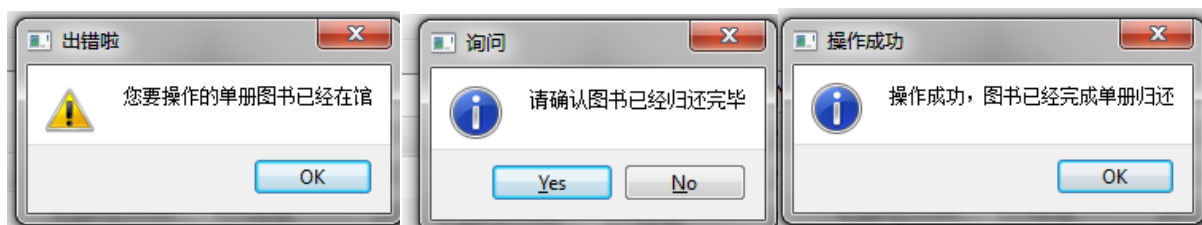


功能截图

查询单用户借阅情况



具体还书操作



4.4 用户管理

说明

用户模块是图书管理系统的核心部分，它用来决定当前用户的操作权限，为了更加人性化，我们将读者与管理员都视作用户，但是不同角色拥有着不同的操作权限，比如读者可以查询图书、借阅图书，但是涉及到还书、管理图书的等操作就需要以管理员身份才有权限进行操作。

实现技术

用户列表

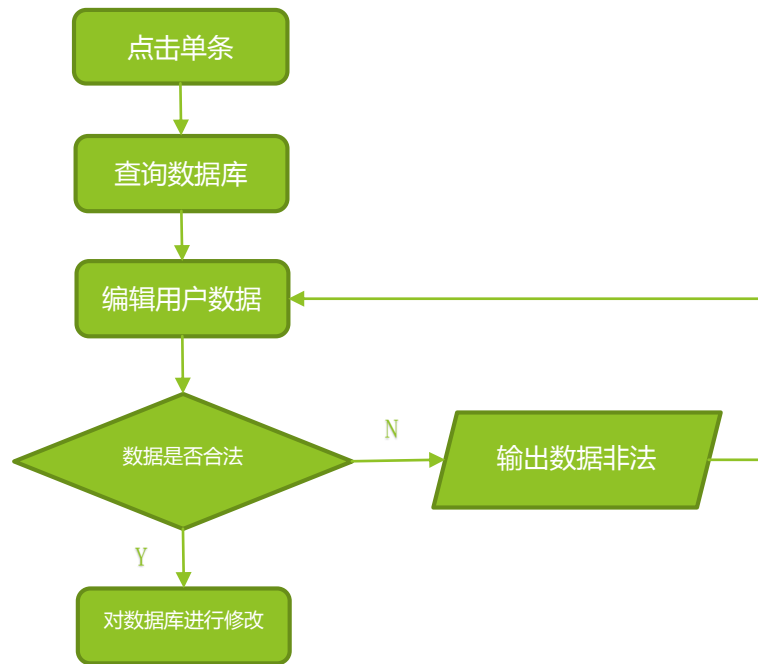
当管理员点击用户管理按钮之后，主程序会向数据库服务器发送查询 qlms_user 表的请求，并将所有用户数据在处理完毕之后显示出来。

SQL 语句：

```
SELECT stuid, name, department, num_borrowed, isadmin FROM qlms_user ORDER BY stuid;
```

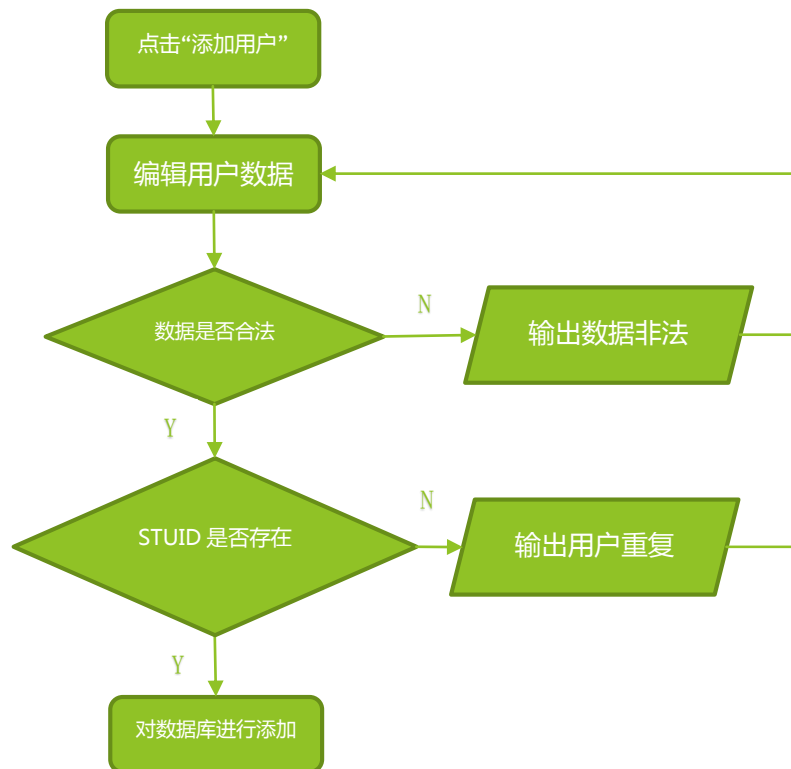
修改用户

当管理员在用户列表页面点击任何一个用户行的时候，会获取当前行的用户的 STUID，并以此 STUID 到数据库中查询当前用户的数据，并将获取到的数据写入到 UI 实例中以呈现给用户。当用户在 UI 窗口完成对用户数据的修改操作之后，点击确定后首先会对用户填写的数据进行校验，之后会通过执行 UPDATE 命令的方式修改用户信息。



添加用户

当用户点击“添加按钮”之后，会调用出相应的添加用户窗口，当用户填写完毕所有资料之后，点击确定之后，主程序首先会检查用户填写的信息的合法有效性，之后会连接数据库检查该 STUID 是否已经存在，若存在则不允许重复添加，在经过所有检查操作之后会将用户的数据插入到数据库中，至此完成了用户添加操作。



功能截图

用户列表



修改用户



用户管理 - 求是图书管理系统

操作选项

用户管理 添加用户

管理用户

学号: 1 管理员

姓名: 管理员

密码: ●

部门: 图书馆

借书上限: 100

修改 删除

添加用户



用户管理 - 求是图书管理系统

操作选项

用户管理 添加用户

添加新用户

学号:

姓名:

密码:

部门:

借书上限: 10

普通用户

创建

4.5 管理图书

说明

图书管理功能可以查询当前图书馆的数目清单，并且可以进行单册图书添加以及批量图书添加操作。

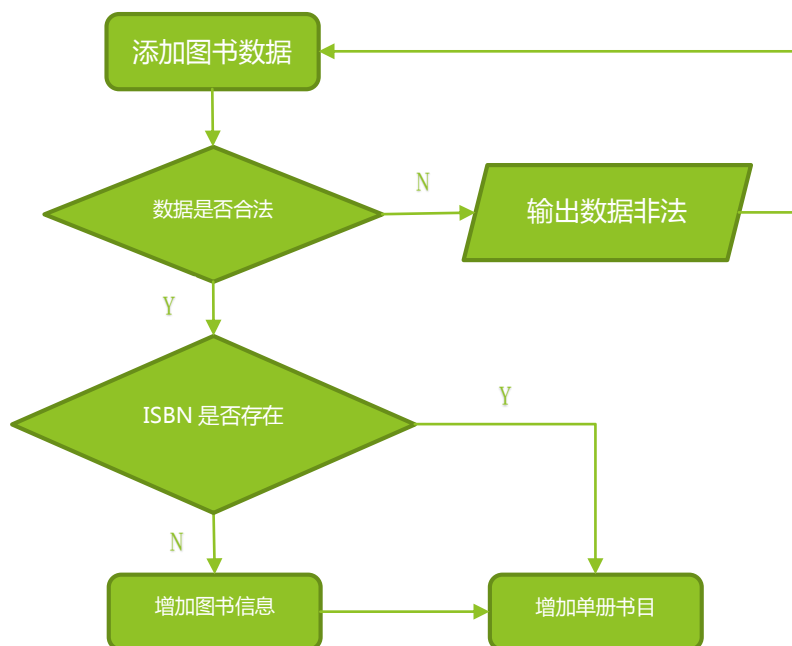
实现技术

图书列表

在后端程序里面通过 ODBC 连接数据库，并通过 SELECT 语句将所有的图书信息查询出来，经过程序处理之后将数据写入到 QListView 当中。

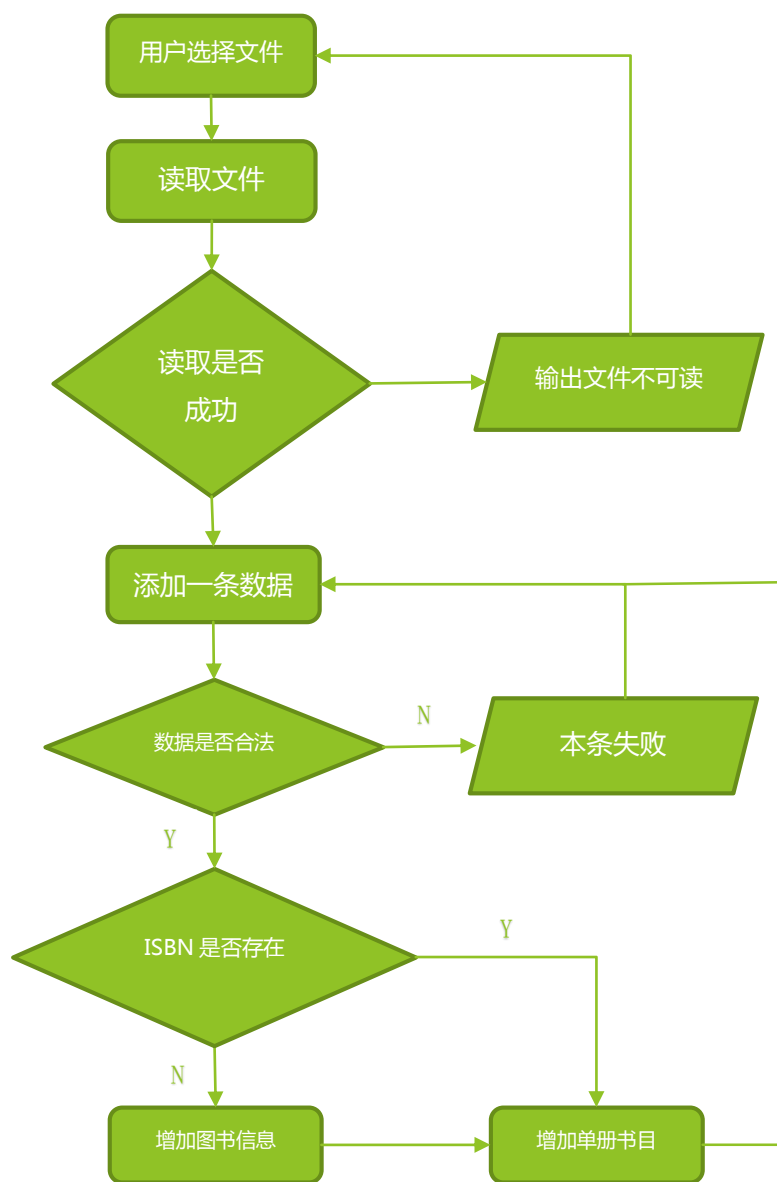
单册图书添加

当用户点击“单册添加”按钮之后，会调用出添加图书的窗体，在用户填写完毕图书信息之后，主程序会对用书输入的用户信息进行校验，在合法的情况下会通过 ODBC 查询数据库，判断这个 ISBN 号是否存在，如果存在那么则只对此 ISBN 增加单册图书信息，若 ISBN 不存在，则插入新的图书信息并插入单册图书信息。



批量图书添加

当用户点击“批量添加”按钮之后，会调用 QFileDialog 控件让用户选择批量添加的文件的.txt，之后主程序会读取文件，并对文件进行解析至对应 item，并调用单册图书添加函数来完成图书的添加工作。



功能截图
图书列表

图书管理 - 求是图书管理系统

操作选项

图书列表

单册添加

批量添加

图书列表

ISBN	图书名称	出版社	作者	数量
2	西窗法雨	院系分馆	刘星	3
21	Computer...	xxx	xxx	2
31	Computer...	xxx	xxx	6
32	编译原理	xxx	xxx	9
33	人工智能1	xxx	xxx	2
34	计算机网络1	xxx	xxx	3
1111	中国文学...	西溪流通...	张少康	2

单册图书添加

图书管理 - 求是图书管理系统

操作选项

图书列表

单册添加

批量添加

单册图书添加

ISBN:

图书名称:

作者:

图书类型:

单价:

出版社:

年份:

位置:

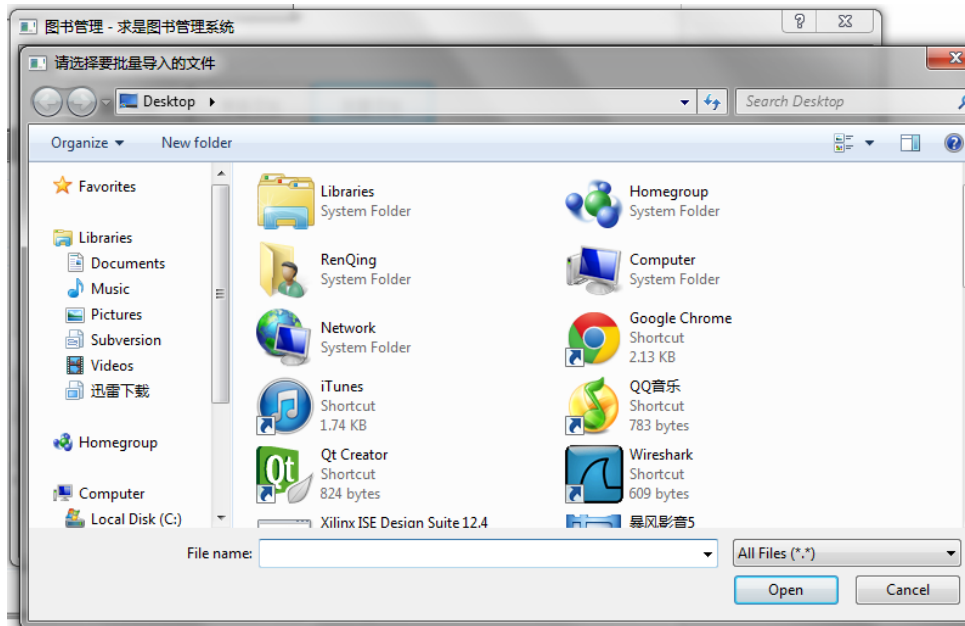
数量:

1

添加

清空

批量图书添加



五、总结

图书管理系统是一个比较实用、常见的系统。我们设计的系统达到了要求的基本功能，并且在这个基础上加入了读者、管理员一体管理、借阅图书上限、单册图书状态等功能。在完成这次大程的过程中，我熟悉并能较熟练地使用 Qt4 进行开发，并且能够熟练地使用 SQL 语言对数据库进行增删改查，收获良多。