**上海大学通信工程学院2013学年秋季学期课外设计项目**

**探 究 实 验 报 告**

**智慧图书馆查询系统 20**

**▉第一节：选题背景**

我们小组维持成员组成，因为有了近一年一起做项目的经历，我们总结经验，一起再度开始本学期的探索。这次我们探究“智慧图书馆查询系统”有以下几个目标：

1、我们喜欢生活中能使用的项目，有一定的日常使用价值，此项目对于今后资料的整理和筛选有着极大的帮助作用。

2、我们对于本学期的《数据结构和算法基础》课程理论知识，尤其是数据库的建立、编辑、查询以及管理各环节，需要通过一个实践课题来帮助我们进一步巩固加深理解。

3、我们这次想把作品做的更加模块化和功能化，例如把查询系统做成管理员端口+用户端口；各功能互通查询等模块化的函数及算法。

4、我们这次不仅把作品当作课题研究，处于它属于实用性作品，因此想对其优化设计，并在今后更加长的过程中能够在提高精度和增强实用性等方面做到进一步优化，达到从学习向实践过度的目的。

综上所述，我们选择了：

**课外实验题目：《智慧图书馆查询系统》**

**▉第二节：设计概述**

**1、基本环境：**

* 语言-c++

原因：C++采用面向对象的思想；可以图书馆管理划分为多模块,便于管理。

* 系统-Linux

原因：能够在单片机以及ARM开发板上进行开发，便于投入实际应用，即在没有计算机系统支持下也可以将系统投入使用。

* 平台：Windows

原因：图形界面更易于操作，和实现人性化使用。

* 工具：Qt

原因：这个软件是1991年奇趣科技开发的一个跨平台的C++图形用户界面应用程序框架。它提供给应用程序开发者建立艺术级的图形用户界面所需的所有功能。

Qt很容易扩展，并且允许真正地组件编程。基本上，Qt 同 X Window 上的 Motif，Openwin，GTK 等图形界 面库和 Windows 平台上的 MFC，OWL，VCL，ATL 是同类型的东西。

优良的跨平台特性:

1）Qt支持下列操作系统: Microsoft Windows 95/98， Microsoft Windows NT， Linux， Solaris， SunOS， HP-UX， Digital UNIX (OSF/1， Tru64)， Irix， FreeBSD， BSD/OS， SCO， AIX， OS390，QNX 等等。

2）Qt 的良好封装机制使得 Qt 的模块化程度非常高，可重用性较好，对于用户开发来说是非常 方便的。 Qt 提供了一种称为 signals/slots 的安全类型来替代 callback，这使得各个元件 之间的协同工作变得十分简单，丰富的 API。

3）Qt 包括多达 250 个以上的 C++ 类，还提供基于模板的 collections， serialization， file， I/O device， directory management， date/time 类。甚至还包括正则表达式的处理功能。

4）支持 2D/3D 图形渲染，支持 OpenGL，大量的开发文档XML 支持等。

**2、基础功能概述：**

读者功能:

借书:1.查找到这本书

2.判断是否借完.判断读者借书总数是否超过规定的书

3.更新这本书的借出数

4.添加一条信息到我的书架

还书:1.将这条信息从我的书架删除

2.将书库中的书的借出数加

3.给这本书打分;

4.如果超时--则需还款才能将此书还掉;

我的书架:

显示我的书架

续借:

更改读者还书时间

管理员功能:

1.入库

2.出库

3.增加读者

4删除读者

5查找读者

**3、系统角色分配：**

本着“人性化”的角度，图书馆系统已角色分配，分为“读者”角色和“管理员”角色，每个角色需登录后方可实现自己的功能界面，以此为基础来进行自己权限范围内的功能操作。

**4、模糊查询与精确查询：**

本系统可以进行以书名为基础的模糊查询，输入信息越多，定位越准确；同时可以进行多信息的精确查询，具体到某本书及其包含的附带信息。

**5、动态评分功能、书目信息更新及系统时间设定：**

本系统中有3部分功能是动态实现的：

其一是基本功能中的书目借阅情况，能准确实时反映出每一本书是否有存货及存货数量等信息；

其二是拓展功能数目的评分信息，每次借阅归还一次书籍，都会在读者端口有个评分功能，每次的打分将加入累计，热门排行榜即根据总分来排序，将数目按照热门程度递减的顺序列出，供读者选择；

其三是借书时间限定，当借出一本书时候有一个系统时间记录，并开始累计计时，归还数目时候如果超出系统时间，会有自动提示进行缴纳超时付费部分。

此三项动态功能更好的完善了图书馆的实用性和人性化原则。

**6、系统智慧功能：**

1.登陆时通过判断账号的权限开启不同的功能;

2.借书时可以规定了借书数,判断读者是否有书未按时还,还可以得到读者的评分,帮助其他读者更好的了解最近哪些书热门;

3.还书时可以判断读者的还书时间是否超时,超时的话则需要罚款才能还书;

4.我的书架--随时可以看到自己所借书籍的信息;

5.管理员可以通过账号查询到读者的信息,包括借书的信息;

**7、图形界面功能升级：**

最初我们设计的查询系统是在调试窗口下运行的，而在后阶段的工程优化期，我们又决定将系统用图形界面实现，因此决定最后方案是用图形界面。

**▉第三节：管理员具体功能设置**

|  |  |  |
| --- | --- | --- |
| **序号** | **功能描述** | **注释** |
| 1 | 身份验证 | 需要密码用户名匹配 |
| 2 | 添加、编辑和删除数目 | 分别对数据库里的信息直接进行修改、添加和删除；并建立了程序和数据库的链接 |
| 3 | 读者信息添加和删除 | 可以由管理员端口进行读者管理 |
| 4 | 查询书目信息 | 更好的分类管理各类数目 |
| 5 |  |  |

**▉第四节：读者具体功能设置**

|  |  |  |
| --- | --- | --- |
| **序号** | **功能描述** | **注释** |
| 1 | 身份验证 | 需要密码用户名匹配 |
| 2 | 定位查询书目信息 | 可以精确查到某本书的某个状态及其相关信息。 |
| 3 | 模糊查询数目信息 | 对于未选定数目的读者可以进行模糊搜索，输入相关条件，系统会自动提取相关书单，供读者选择 |
| 4 | 借阅书目 | 书目信息自动进入读者“书架”列表，并有相关借书信息提示 |
| 5 | 归还书目 | 系统会自动计时借阅时间，如果超过规定借阅时间，会进缴纳超时费用的提示 |
| 6 | 续借数目 | 在未超时的情况下可以有读者端口续借该数目，并系统自动延长借阅时间 |
| 7 | 图书热门度实时排序查询 | 可以实时了解到图书馆各书目借阅量变化和热门程度的变化，更加方便查阅到最受用的信息 |
| 8 | 修改读者端口密码 | 便于保护读者隐私和账户安全，如果拓展到账户内存在费用，账户安全功能则更为重要 |
| 9 | 查看读者端口书架书目 | 便于读者清楚自己已经借阅的书目和信息 |
| 10 |  |  |

**▉第五节：探究过程中遇到的问题**

1、安装mysql问题:

在linux下安装mysql的时候是一件非常容易的事情.安装成功后,我们往mysql导入数据但是我们下的版本不支持sql语句导入,需要在linux系统中在终端下打入命令

Mysql –uroot –p123456 bookstore –local –infile=1 –e “load datelocal infile ‘/home/young/book.txt’” into table book;

2、乱码问题:

在导入中文的时候出现乱码,想到可能是mysql和linux编码的区别.经过查询发现mysql是gbk的编码,而linux是utf-8所以要将mysql 的编码形式改成utf-8的格式

3、连接mysql:

将程序如何连接数据是一件非常烦的事情,我们查阅书籍发现c语言和mysql有一些函数可以实现连接,于是就试了试,但是数据库并不认识这些函数,于是我们就把包含mysql驱动的函数和c语言函数的链接库添加在一起,这样编译器就可以知道这些函数是用来干什么的.

例如: g++ -L/usr/lib/mysql –lmysqlclient –o

使用数据库连接的函数: conn = mysql\_init(NULL); 初始化函数句柄

mysql\_real\_connect(conn,server,user,password,database,0,NULL,0)连接数据库

4、程序设计遇到的问题:

我们第一个遇到的问题就是怎样将在数据库语句中怎么添加变量,可以根据我们的选择来执行相应的步骤;

例如: 如果我们想通过id查询书籍,则数据库里面应该传入”select \* from book where id =1”; 但是id应该是一个变量如果用双引号括起来会变成字符串,为了解决这一问题,我们用了sprintf这个函数:用法如下:

Sprint(a,“select \* from book where id =%d”,id);

解决这一问题,我们的项目就可以真正的开始了.

接下来我们就要去学习mysql的 命令,通过查找,我们找到了我们需要的,例如: 查找 用”select … from… where id =…”;删除”delete from where id =…”

更新 “update book set… where …”等等.

5、借书还书时候如何判断所借书籍是否超时：

对于其他功能我们没遇到什么问题,但是对于借书,还书时候如何判断所借书籍是否超时,我们一开始不知道怎么解决.如果我们自己去计算时间的话,肯定会涉及到闰年这些运算,会导致程序庞大.正在这时,我们想到为什么不用系统本身的时间了,由于我们对qt有一定的了解,通过系统调用我们就获得了当前的借书时间,和还书时间,同时只要判断读者还书时的时间是否大于当前的时间。

**▉第六节：基本功能代码分析**

|  |
| --- |
| 【管理读者功能模块】  AddReader::~AddReader()  {delete ui;}  void AddReader::on\_YesButton\_clicked() //增加读者  {  QString userid = ui->userIdEdit->text().trimmed();  QString username =ui->unameEdit->text().trimmed();  QSqlQuery query;  query.prepare("insert into user values(?,'123456',?,0)");  query.addBindValue(userid);  query.addBindValue(username);  query.exec();  QMessageBox::warning(this, tr("Waring"),  tr("successful"),  QMessageBox::Yes);  this->close();  }  【用户登录功能模块】  Login::~Login()  {delete ui;}  void Login::on\_loginButton\_clicked()  {  QSqlQuery query;  query.exec("select \* from user");  int a=1;  while(query.next()){  if(ui->uname->text().trimmed()==query.value(0) && ui->upasswd->text().trimmed()==query.value(1)){  userid = query.value(0).toString();  userrank = query.value(3).toInt();  this->close();  ReaderView \*view = new ReaderView();  view->show();  a = 0;  }  }  if(a==1){  QMessageBox::warning(this, tr("Waring"),  tr("user name or password error!"),  QMessageBox::Yes);  ui->uname->clear();  ui->upasswd->clear();  ui->uname->setFocus();  }  }  void Login::on\_exitButton\_clicked()  {  QMessageBox qmsg;  qmsg.setText(QTime::currentTime().  toString("hh:mm:ss"));  qmsg.setStandardButtons(  QMessageBox::Yes|QMessageBox::No);  if(qmsg.exec()==QMessageBox::Yes){  this->close();  }  }  【图书借阅功能模块】  BorrowBook::BorrowBook(QWidget \*parent) :  QDialog(parent),  ui(new Ui::BorrowBook)  {  ui->setupUi(this);  }  BorrowBook::~BorrowBook()  {delete ui;}  void BorrowBook::on\_YES\_clicked()  {  //qDebug()<<userid;  //qDebug()<<userrank;  QSqlQuery query;  //query.prepare("select r");  //检查借书数量  query.prepare("select count(\*) from userbook where userid = ?");  query.addBindValue(userid);  query.exec();  query.next();  int c = query.value(0).toInt();  if(c>2){  QMessageBox::warning(this, tr("Waring"),  tr("over the book!"),  QMessageBox::Yes);  }else{  //判断是否有书超时 //更新这本书  int bookid,feeling,num,rent;  QString bookISBN;  QString bookname;  double price;  QString ISBN = ui->B\_ISBNEdit->text().trimmed();  if(!ISBN.isEmpty()){  query.prepare("select \* from book where ISBN = ?");  query.addBindValue(ISBN);  query.exec();  while(query.next()){  bookid = query.value(0).toInt();  bookISBN = query.value(1).toString();  bookname = query.value(2).toString();  feeling = query.value(7).toInt();  price = query.value(8).toDouble();  num = query.value(5).toInt();  rent = query.value(6).toInt();  }  if(num==rent){  QMessageBox::warning(this, tr("Waring"),  tr("NO the book!"),  QMessageBox::Yes);  }else{  // QString borrowdate = QDateTime::currentDateTime().date().toString();  QString borrowtime = QDateTime::currentDateTime().time().toString();  QString returntime = QDateTime::currentDateTime().time().addSecs(10).toString();  qDebug()<<"returntime"<<returntime; //测试  rent = rent +1;  query.prepare("update book set rent = ? where ISBN = ?");  query.addBindValue(rent);  query.addBindValue(ISBN);  query.exec();  //插入这本书籍  query.prepare("insert into userbook values(?,?,?,?,?,?,?,?)");  query.addBindValue(userid);  query.addBindValue(bookid);  query.addBindValue(bookISBN);  query.addBindValue(bookname);  query.addBindValue(feeling);  query.addBindValue(price);  query.addBindValue(borrowtime);  query.addBindValue(returntime);  query.exec();  QMessageBox::warning(this, tr("Waring"),  tr("successful!"),  QMessageBox::Yes);  this->close();  }  }else{  QMessageBox::warning(this,tr("waring"),tr("NO ISBN"),QMessageBox::Yes);  }  void BorrowBook::on\_NO\_clicked()  {this->close();}  【续借功能模块】  ContinueBook::~ContinueBook()  {delete ui;}  void ContinueBook::on\_C\_Yes\_clicked()  {  QString returntime = QDateTime::currentDateTime().time().addSecs(3600).toString();  QString ISBN = ui->C\_ISBNedit->text().trimmed();  QSqlQuery query;  // query.prepare("select returndate from userbook where bookISBN =?");  //query.addBindValue(ISBN);  //query.exec();  //query.next();  //QString returndate = query.value(0).toString();  query.prepare("update userbook set returndate = ? where bookISBN=? ");  query.addBindValue(returntime);  query.addBindValue(ISBN);  query.exec();  QMessageBox::warning(this, tr("Waring"),  tr("successful"),  QMessageBox::Yes);  }  void ContinueBook::on\_C\_NO\_clicked()  {this->close();}  【超时付费功能模块】  PaidMoney::PaidMoney(QString money,QWidget \*parent) :  QDialog(parent),  ui(new Ui::PaidMoney)  {  ui->setupUi(this);  ui->MoneyEdit->setText(money);  }  PaidMoney::~PaidMoney()  {delete ui;}  void PaidMoney::on\_YesButton\_clicked()  {this->close();}  【还书功能模块】  Returnbook::~Returnbook()  {delete ui;}  void Returnbook::on\_Yes\_clicked()  {  QString ISBN = ui->R\_ISBNedit->text().trimmed();  QSqlQuery query;  int overtime=0;  int rent =0;  QString returndate;  query.prepare("select returndate from userbook where bookISBN=?");  query.addBindValue(ISBN);  query.exec();  while(query.next()){  returndate = query.value(0).toString();  }  QString s = "2013-10-22 " + returndate;  query.prepare("select strftime('%s','now','localtime')-(select strftime('%s',?))");  //query.prepare("select julianday('now')\*86400 -julianday('23:18:00')\*86400)");  query.addBindValue(s);  query.exec();  while(query.next()){  overtime = query.value(0).toInt();  qDebug()<<"overtime"<<overtime;  }  if(overtime<=0){  //删除这本书  query.prepare("delete from userbook where bookISBN = ?");  query.addBindValue(ISBN);  query.exec();  //更新这本书  query.prepare("select feeling from book where ISBN=?");  query.addBindValue(ISBN);  query.exec();  query.next();  int feeling = query.value(0).toInt();  bool c = ui->check1->checkState();  if(c){  feeling = feeling+1;  }  c = ui->check1->checkState();  if(c){  feeling = feeling+2;  }  c = ui->check1->checkState();  if(c){  feeling = feeling+3;  }  query.prepare("select rent from book where ISBN = ?");  query.addBindValue(ISBN);  query.exec();  while(query.next()){  rent = query.value(0).toInt();  }  rent = rent -1;  query.prepare("update book set rent = ?,feeling =? where ISBN = ?");  query.addBindValue(rent);  query.addBindValue(feeling);  query.addBindValue(ISBN);  query.exec();  QMessageBox::warning(this, tr("Waring"),  tr(" return successful!"),  QMessageBox::Yes);  this->close();  }else{  query.prepare("delete from userbook where bookISBN = ?");  query.addBindValue(ISBN); //删除读者书架的书  query.exec();  query.prepare("select rent from book where ISBN = ?");  query.addBindValue(ISBN);  query.exec();  while(query.next()){  rent = query.value(0).toInt();  }  rent = rent -1;  query.prepare("update book set rent = ? where ISBN = ?");  query.addBindValue(rent);  query.addBindValue(ISBN);  query.exec();  QMessageBox::warning(this, tr("Waring"),  tr("over time!"),  QMessageBox::Yes);  double money = 0.01\*overtime;  QString s = QString::number(money,'f',0);  PaidMoney \*p = new PaidMoney(s);  p->show();  this->close();  }  void Returnbook::on\_No\_clicked()  {this->close();}  [全部基本功能代码完毕] |

**▉第七节：制作进程**

第一周：选定项目，并进行预览和评估，招纳组员，组成团队并上报。

第二周：第一次讨论会，分析项目规划和大致方向，比如每个模块要实现的查询或编辑功能等，技术人员开始准备研究每一部分。

第三周：技术人员开始编写组框架代码；记录人员负责数据备置与中期报告拟写，并在初步草稿的基础上进行完善计划和补充功能。

第四周：召开中期会议，交流制作进程，在完成基本功能的前提下，我们进一步规划更加较为人性和实际应用的功能，来对我们的项目进行润色。

第五周：测试经过完善后的程序系统，并咨询身边同学，听取他们的建议和想法，结合自己项目，对自己的系统再进行修改和完善。

第六周：测试人员开始对程序进行测试，寻找不同情况下的系统输入，检查系统的反应，并最终将出现的问题反映给技术组人员，做最后的调整工作。此时有了一个巨大的尝试，我们准备改成“图形界面”操作方式，这无疑是个巨大的挑战，因为可能要大改所有代码。

第七周：“图形界面”查询系统探究实验成功，完成项目全部内容，编写最终课题报告，小组共同开展总结会议，对项目涉及的总体思路、构架、具体算法、原理等方面进行梳理总结，共同最后调试程序。

第八周：小组仔细探究程序较为复杂部分的原理和代码含义；全体组员准备答辩验收时自己负责的展示部分；周六参加项目验收答辩。

**▉第八节：组员分工情况**

|  |  |
| --- | --- |
| 人员 | 负责部分 |
| 卿杨春 | 作为组长，非常认真负责的把握项目的进展程度，提供重要的技术支持；并主要负责几个技术性较强的模块，结合测试人员不断的进行系统修正和细节完善；负责最后部分图形界面的主要设计；能在一定程度上完成共同讨论出来的功能构想。 |
| 严 星 | 负责查询系统功能建设基本构想，结合小组讨论的结果，最终整合功能要求和信息，给出一个较为完善的功能列表和框图，让技术人员进一步实现。 |
| 刘 晶 | 负责程序代码的编写，同技术组一起进行功能代码段的构建，完成出现的难题，并且预测可能出现的不同状况，在编写代码时予以实现考虑；提出了“图形界面”的设计思想和最终探究决策；参与最终报告的编写。 |
| 曹伯乐 | 负责具体功能的构建和细化、具体化每一部分的功能，并结合人性化以及实际性的角度进行协调；负责中期阶段程序的调试过程，寻找功能漏洞，并模拟不同情况，和技术人人员进行交流；模拟图形界面所编写中期报告、最终报告。 |
| 牛丽莹 | 负责程序维护和测试，不断的进行各种功能的尝试，以及各种反馈结果的记录分析，及时和技术组人员进行交流，达到最终优化项目的目的 |

**▉第九节：结束语**

项目进程马上就截函数了，看着自己的查询系统，基本功能也已经实现，还有不少拓展功能，挺有成就感的。我们最后要做的是不断的进行测试看看是否有漏洞，并且逐步添加一些更加优化、更加人性的嵌入功能；仔细读懂一些较为复杂的原理和程序细节；找一些测试人员来参与试运转，争取更加完美和操作简单清晰。

我想说，在整个过程中，我们十分关注我们学到了什么，无论是数据结构原理也好，系统知识也好，还是工程技巧也罢，我们都很用心的将其总结归纳，并记录下来，作为这个项目真正对我们的价值。

我想以一句话作为结束语：我相信8个星期以来一起奋斗的每一个队友，也相信我们项目团队的每一个人，我们享受过程，在通向成功的道路上走着；如果感到累，是因为我们正在走上坡路！

[全文 完]

日 期：2013年10月20日