**第一章** **项目概述**

(修改时间2017年11月14日14:54)

**1.1** **目录**

**第一章：**[**项目概述**](#项目概述)

1.1 [目录](#目录)

1.2 [项目背景与目的](#项目背景与目的)

1.3[项目的范围和目标](#项目的范围和目标)

[系统流程图](#系统流程图)

[基本功能](#基本功能)

[特色功能](#特色功能)

**第二章：**[**项目估算**](#项目估算)

2.1[数据结构 & 接口：](#数据结构和接口问题)

|  |  |  |  |
| --- | --- | --- | --- |
| [**第一阶段第一部分**](#第一阶段第一部分) |  |  |  |
|  | [**BookManagementIOHeader.h**](#BookManagementIOHeader) | [**删除信息**](#删除信息部分) | [**查询书籍信息**](#查询书籍信息) |
|  | [**BookManagementIO.cpp**](#BookManagementIOcpp) | [**分类控制部分**](#分类部分) | [**查询用户信息**](#查询用户信息) |
|  | [**修改书籍信息**](#修改书籍信息部分) | [**修改用户信息**](#修改借书人信息) |  |
| [**第一阶段第二部分**](#第一阶段第二部分) |  |  |  |
|  | [**IdGenerator.h**](#IdGeneratorh) | [**LogUtil.h**](#LogUtil) | [**Book(CLASS)**](#Book) |
|  |  |  |  |
|  |  |  |  |

2.2 [使用的评估技术](#使用的评估技术)

2.3 [工作量、成本、时间估算](#工作量、成本、时间估算)

**第三章：**[**进度评估**](#风险评估)

**第四章：**[**项目进度计划**](#项目进度计划)

**第五章：**[**细则要求问题**](#细则要求问题)

**第六章：**[**软件配置**](#软件配置)

**第七章：**[**人员组织**](#人员组织)

**1.2** **项目背景与目的**

图书馆管理系统是图书馆管理工作中不可缺少的部分，它对图书馆的管理者和使用者来说都十分重要。而近年来快速发展的计算机软件技术为方便快捷地进行图书管理提供了条件。借助计算机强大的信息处理和存储的能力，我们可以方便地存储大量图书和借书人的信息，并快速地对这些信息进行查找、修改、删除、增添和管理。极大地方便了图书的管理者和使用者——使用计算机及图书管理软件对图书信息进行管理，有手工管理所无法比拟的优点，例如：检索迅速、查找方便、易修改、可靠性高、存储量大、数据处理快捷、保密性高、寿命长、成本低、便于打印等。这些优点是图书馆的科学化、正规化管理的重要条件。因此，开发这样的管理软件是很有必要的事。

本系统的开发宗旨是实现图书馆管理的系统化、规范化和自动化，达成图书，人员资料集中、统一管理的目标。

**1.3****项目的范围和目标**

**1.3.1 范围描述**

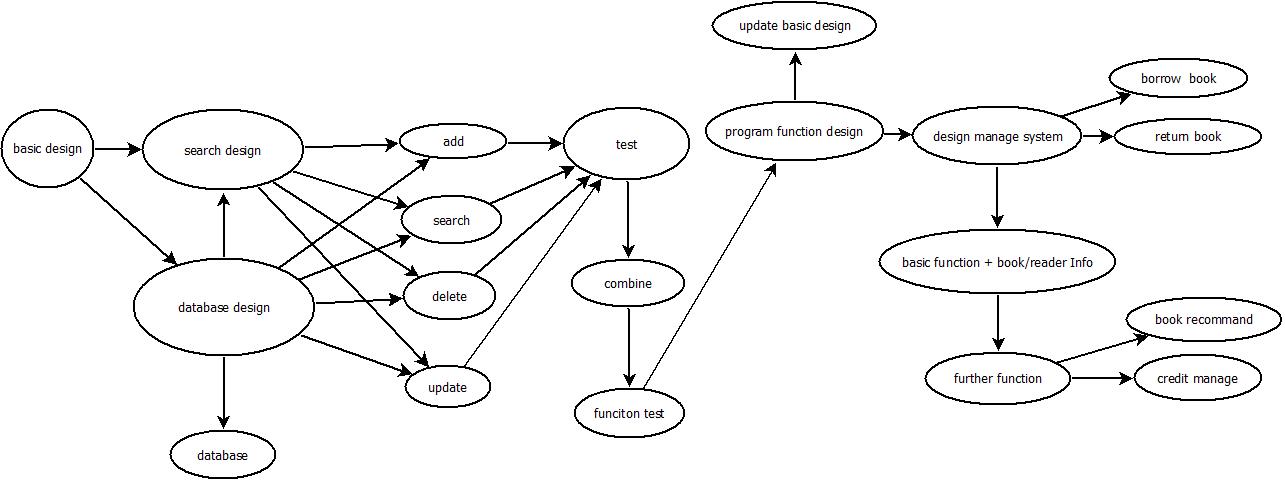
满足基本图书馆结束者以及管理员使用的小型图书管理工作。

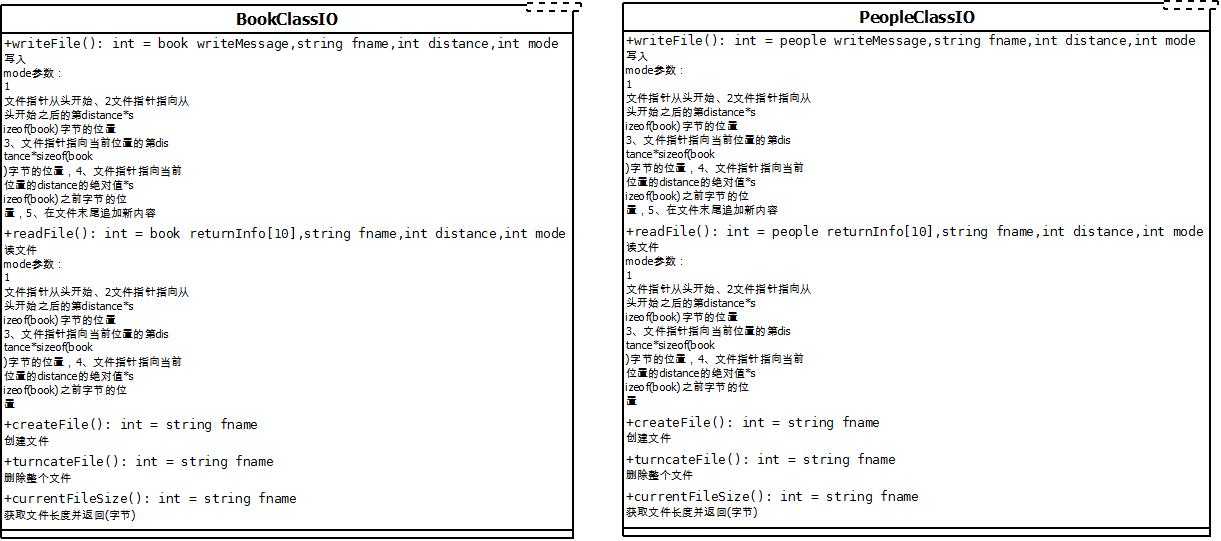
**1.3.2 主要功能**

（1）概述

本管理系统分为两大部分：图书信息管理和借书人信息管理。这两大部分分别使用二进制文件存贮图书信息和借书人信息。同时，每一部分分为三层：数据层（用于存储数据）、传输层（用于读取数据并将其发送到需要该数据的部分）和应用层（处理传输层传递的数据，将要实现的各种数据）。

（2）系统流程图





（3）功能描述

**基本功能：**

1.以二进制文件存储图书以及借书人信息

2.书籍和借书卡信息的增删改查

3.有借书和还书操作

备注：书籍信息至少为10万，借书人至少500个，要求不得直接将数据全部载入内存。

**特色功能（中后期视实际工程完成情况再考虑是否可实现以及保留）：**

1：提供个人近来借书历史记录

2：根据历史记录可以推荐类似书

3：图书馆财务管理系统与相关功能（包括赔偿制度）

4：客户端和服务端分开，借书人和图书管理员用的不是一个端，

但是数据都是相同文件里的

5: 一些特殊数据需要用鲜亮的颜色来显示 比如库存缺少数字则为红色，

或者库存充足则绿色

6: 保存本(类似于本校的保存本制度)

7：账户等级（普通用户I，普通用户II，图书管理员，超级管理员）

8：隐藏在账号后的信用评级（权限高低影响借书时间和数量）

9：图书类型分类。（便于第2条的实现）

10：热门书籍排行榜以及热门书籍推荐（根据借出次数）

11：新书推荐（根据书本添加时间）

12：评分功能以及评分排行。（类似豆瓣评分）

13：收藏书籍功能。

14：书籍名称首字母索引。

**1.3.4 技术约束**

条件：c/c++、使用二进制文件读写实现数据读写

限制：硬件内存大小、大数据量的搜索时间、时间安排

**第二章** **项目估算**

**2.1****数据结构&接口问题：**

**第一阶段第一部分**

**BookManagementIOHeader.h：**

**（丛爽）**

**typedef struct{**

**unsigned int bookId;//identify number for books**

**unsigned int peopleId;//indentify number for readers**

**int bookClass;//** **bookClass为1-22，分别对应22个分类**

**float price;//价格**

**char bookName[45];//书名,最大44个字节，22个汉字**

**char publisher[25];//出版社,24个字节，12个汉字**

**char author[25];//作者，24个字节，12个汉字**

**char publishDate[5];//出版年份，4个字节，只有年份，例如2009**

**char borrowTime[11];//time of lending the book,year xxxx month xx date xx hour xx**

**char borrowFlag;//借出标志位**

**char isReservedFlag;//预定标志位**

**char isPreservedFlag;//保存本标志位**

**char isLated;//是否逾期**

**char isDeleted;//whether the record is delated(false as default)**

**char isRealDeleted;//是否真正删除**

**}book;**

**typedef struct{**

**unsigned int peopleId;//借书人编号**

**unsigned int[10] recentBorrowId;//最近借阅书籍编号**

**unsigned int[5] booksBorrowed;//已经借阅的书籍**

**char password[10];//password for the user**

**int level;//借书人级别**

**int credit;//信用点**

**char peopleName[20];//people's name**

**char isDeleted;//whether the record is delated(false as default)**

**char isRealDeleted;//是否真正删除**

**}people;**

**分类部分：**

**（颜晓书）**

**typedef enum bookClassification{MaoDeng,PhilosophyReligion,Social,PoliticalLaw,Military,Economic,Literature\_PE,Language,**

**Literature,Art,HistoryGeography,Nature,MathChemistry,Astronomy,Biology,Medicine,Agriculture,Industry,Transportation,Aerospace,** **EnvironmentSafety,Comprehensive}BookType;**

/\*依次对应为马列主义、毛泽东思想、邓小平理论,哲学宗教,社会科学总论,政治法律,军事,经济,文化、科学、教育、体育,语言、文字,文学,艺术,历史、地理,自然科学总论,数理科学与化学,天文学、地球科学,生物科学,医药、卫生,农业科学,工业技术,交通运输,航空、航天,环境安全科学,综合性图书\*/

**string getBookValue(BookType num){**

**switch(num){**

**case MaoDeng:**

**return "马列主义、毛泽东思想、邓小平理论";**

**case PhilosophyReligion:**

**return "哲学宗教";**

**case Social:**

**return "社会科学总论";**

**case PoliticalLaw:**

**return "政治法律";**

**case Military:**

**return "军事";**

**case Economic:**

**return "经济";**

**case Literature\_PE:**

**return "文化、科学、教育、体育";**

**case Language:**

**return "语言、文字";**

**case Literature:**

**return "文学";**

**case Art:**

**return "艺术";**

**case HistoryGeography:**

**return "历史、地理";**

**case Nature:**

**return "自然科学总论";**

**case MathChemistry:**

**return "数理科学与化学";**

**case Astronomy:**

**return "天文学、地球科学";**

**case Biology:**

**return "生物科学";**

**case Medicine:**

**return "医药、卫生";**

**case Agriculture:**

**return "农业科学";**

**case Industry:**

**return "工业技术";**

**case Transportation:**

**return "交通运输";**

**case Aerospace:**

**return "航空、航天";**

**case EnvironmentSafety:**

**return "环境安全科学";**

**case Comprehensive:**

**return "综合性图书";**

**default:**

**break;**

**}**

**}**

**当前功能：**

**BookManagementIO.cpp：**

**（丛爽）**

**class BookClassIO{//this class is used for reading or writing datas of books.**

**private:**

**public:**

**int writeFile(book writeMessage,string fname,int distance,int mode);**

**int readFile(book returnInfo[10],string fname,int distance,int mode);**

**int currentFileSize(string fname);**

**int createFile(string fname);**

**int turncateFile(string fname);**

**};**

**class ReaderClassIO{//this class is used for tansfering data from database for readers**

**private:**

**public:**

**int writeFile(people writeMessage,string fname,int distance,int mode);**

**int readFile(people returnInfo[10],string fname,int distance,int mode);**

**int currentFileSize(string fname);**

**int createFile(string fname);**

**int turncateFile(string fname);**

**};**

**NEW：**

**unsigned int hashOperation(char \*arr,int arrlength,int hashLength)//进行哈希string getFilePath(string fname)//获取同名文件路径(之前以带路径的方式存储过)**

**hashOperation函数正在检测**

\* writeFile()mode参数：

1 文件指针从头开始、2文件指针指向从头开始之后的第distance\*sizeof(book)字节的位置 3、文件指针指向当前位置的第distance\*sizeof(book)字节的位置，4、文件指针指向当前位置的distance的绝对值\*sizeof(book)之前字节的位置，5、在文件末尾追加新内容

(当前问题：发生截断)

\* readFile()mode参数：

1 文件指针从头开始、2文件指针指向从头开始之后的第distance\*sizeof(book)字节的位置 3、文件指针指向当前位置的第distance\*sizeof(book)字节的位置，4、文件指针指向当前位置的distance的绝对值\*sizeof(book)之前字节的位置

\* readFile()函数中返回的数组，如果已经读完文件，将填充为:unsigned int/int类型填充0,char类型填充'#'

以上函数返回值为一int型数据：

-6 文件存在、 -5 mode非法、-4 文件已经读完、-3 路径不存在、 -2 文件不存在、 -1 文件打开失败、0 成功 、1 读失败、 2 写失败、 3文件名为空 、4 文件删除失败 、 5 文件创建失败 、 6 文件被占用

**查询书籍信息：**

**（张文政）**

**void searchAllBooks();**//无条件展示所有书籍

**void searchBookById(unsigned int bookId);**//按书号查询书籍信息

**void searchBookByPeopleId(unsigned int peopleId);**//按借书人序号查询书籍信息

**void searchBookByClass(int bookClass);**//按书籍类别查询书籍信息

**void searchBookByPrice(float price1,float price2);**//按价格 区间查询书籍信息

**void searchBookByName(char name[20])**;//按名字查询书籍信息

**void searchBookByPublisher(char publisher[20]);**//按出版社查询书籍信息

**void searchBookByAuthor(char author[8]);**//按作者查询书籍信息

**void searchBookByPublishDate(char publishDate[8]);**//按出版日期查询书籍信息

**void searchBookByBorrowTime(char borrowTime[10]);**//按借出日期查询书籍信息

**int searchInstanceOfBook(unsigned int bookId);**//查询书号来返回某本书在文件里的distance

**查询用户信息：**

**（张文政）**

**void searchAllPeople();**//无条件展示所有用户信息

**void searchPeopleById(unsigned int peopleId);**//通过用户号查询用户信息

**void searchPeopleByRecentBorrowId(unsigned int recentBorrowId[10]);**//暂时不考虑

**void searchPeopleByRank(int rank);**//通过用户等级查询用户信息

**void searchPeopleByCredit(int credit);**//通过信用等级查询用户信息

**int searchInstanceOfPeople(unsigned int peopleId);**//查询用户号来返回某用户在文件里的distance

**删除信息部分：**

**（徐云哲）**

//根据书籍编号（bookId）

**//void falseDeleteBookbyId(string filename,book \*b,unsigned int bookId)**

//根据书籍分类（bookClass）

**void falseDeleteBookbyClass(string filename,book \*b, int bookClass)**

//根据书籍价格区间（price）

**void falseDeleteBookbyPrice(string filename,book \*b, float price1,float price2)**

//根据书籍名称（bookName）

**void falseDeleteBookbyName(string filename,book \*b,char \*name)**

//根据书籍出版社（publisher）

**void falseDeleteBookbyPublisher(string filename,book \*b, char \*publisher)**

//根据书籍作者（author）

**void falseDeleteBookbyAuthor(string filename,book \*b, char \*author)**

//根据书籍出版年份（publishDate）

**void falseDeleteBookbyPublishDate(string filename,book \*b, char \*publishDate)**

//根据书籍借出时间（borrowTime）

**void falseDeleteBookbyBorrowTime(string filename,book \*b, char \*borrowTime)**

//各类删除调用

**void falseDeleteBook(string fname,book \*b)**

//----------------------------------------------真删除-------------------------------

//根据书籍编号（bookId）

**//void trueDeleteBookbyId(string filename,book \*b,unsigned int bookId)**

//根据书籍编号恢复（bookId）

**void recoverDeleteBookbyId(string filename,book \*b,unsigned int bookId)**

**修改书籍信息部分：**

(颜晓书)

总体框架：

**bool updateBook\_(string filename,book \*Book,"要更新的信息" )**

1：修改书籍编号（bookId）

**函数：bool updateBookId(string filename,book \*Book,int BookId);**

2：修改借书人编号（peopleId）

**函数：bool updateBookPeopleId(string filename,book \*Book,int PeopleId);**

3：修改书籍分类（bookClass）

**函数：bool updateBookClass(string filename,book \*Book,int PeopleId);**

4：修改书籍价格（price）

**函数：bool updateBookPrice(string filename,book \*Book,float Price);**

5：修改书籍名称（name）

**函数：bool updateBookName(string filename,book \*Book,char \*Name);**

6：修改书籍出版社（publisher）

**函数：bool updateBookPublisher(string filename,book \*Book,char \*Publisher);**

7：修改书籍作者（author）

**函数：bool updateBookAuthor(string filename,book \*Book,char \*Author);**

8：修改书籍出版年份（publishDate）

**函数：bool updateBookPublishDate(string filename,book \*Book,char \*PublishDate);**

9：修改书籍借出时间（borrowTime）

**函数：bool updateBookBorrowTime(string filename,book \*Book,char \*BorrowTime);**

10：修改借出标志位（borrowFlag）

**函数：bool updateBookBorrowFlag(string filename,book \*Book,char borrowFlag);**

11：修改预定标志位（isReservedFlag）

**函数：bool updateBookReservedFlag(string filename,book \*Book,char ReservedFlag);**

12：修改保存本标志位（isPreservedFlag）

**函数：bool updateBookPreservedFlag(string filename,book \*Book , char PreservedFlag);**

13：修改逾期标志位（isLated）

**函数：bool updateBookLatedFlag(string filename,book \*Book,char LatedFlag);**

14：修改删除标志位（isDeleted）

**函数：bool updateBookDeletedFlag(string filename,book \*Book,char DeletedFlag);**

15:修改真删除标志位（isRealDeleted）

**函数：bool updateBookRealDeletedFlag(string filename,book \*Book,bool RealDeletedFlag);**

**修改借书人信息部分：**

(颜晓书)

总体框架：

**bool updatePeople\_(string filename,people \*People,"要更新的信息" )**

1,修改借书人编号（peopleId）

**函数：bool updatePeopleId(string filename, people \*People,int PeopleId);**

2,修改最近借阅书籍编号（recentBorrowId）

**函数：bool updatePeopleRecentBorrowId(string filename,**

**people \*People,int \*RecentBorrowId);**

3,修改已经借阅的书籍编号（booksBorrowed）

**函数：bool updatePeopleBooksBorrowed(string filename,**

**people \*People,int \*BooksBorrowed);**

4,修改用户密码（password）

**函数：bool updatePeoplePassword(string filename, people \*People,int \*Password);**

5,修改借书人级别（level）

**函数：bool updatePeopleLevel(string filename, people \*People,int Level);**

6,修改借书人信用点（credit）

**函数：bool updatePeopleCredit(string filename, people \*People,int Credit);**

7，修改借书人姓名（peopleName）

**函数：bool updatePeopleName(string filename, people \*People,char \*PeopleName);**

8,修改该用户删除标志位（isDeleted）

**函数：bool updatePeopleDeleted(string filename, people \*People,char IsDeleted);**

9, 修改该用户真删除标志位（isRealDeleted）

**函数：bool updatePeopleRealDeleted(string filename, people \*People,char IsRealDeleted);**

**第一阶段第二部分：**

**IdGenerator.h：**

**（丛爽）**

**int hashOperation(char \*arr,int length)** // this function is used to deal with the three string and change them into an id which is easy to identify

**bool charMatch(char \*arr\_O,char \*arr\_T,int length)** //compare two strings, and find out whether their length are the same value which match length(int)

**bool bookMatch(book obj,book test)** //this function is used to detect whether the books are identical

**int getIntDigit(unsigned int num)** //get how many digits the number has

**unsigned int intCut(unsigned int obj,int begPosition,int endPosition)**

//intCut是将unsigned int型数据按begPosition和endPosition

//this function is used to get specific part of an int data, begPosition < endPosition,begPosition is where it begin to cut

**int collideDetection(unsigned int value\_O, unsigned int value\_T,book testObject,book test)**//this function is used to detect hash collision and correct it

**unsigned int BookIdGenerater(char \*bookName,char \*publisher,char \*author,char \*publishDate,int bookClass,int curBookNum)**

//r是产生book的id使用，参数为bookName[]书名publisher[]出版商,author[]作者,publishDate[]出版日期,bookClass书的种类, curBookNum当前书的数量

//use this function to generate hash value

**LogUtil.h**

**（丛爽）**

**class IOLOG{**

**private:**

**public:**

**int stLog(std::string info)**

**{**

**/\*this function is used to write log\*/**

**std::ofstream outFile;**

**std::string statusLog = "log.txt";**

**outFile.open(statusLog.data(),std::ios::app);**

**if(!outFile) return -1;**

**outFile << info;**

**outFile.close();**

**return 0;**

**}**

**int sendMail();//send email**

**};**

**Book类**

**（张文政）**

**typedef enum bookClassification{MaoDeng,PhilosophyReligion,Social,PoliticalLaw,Military,Economic,Literature\_PE,Language,**

**Literature,Art,HistoryGeography,Nature,MathChemistry,Astronomy,Biology,Medicine,Agriculture,Industry,Transportation,**

**Aerospace,Environmental,Safety,Comprehensive}BookType;**

**class Book//图书行为操作类**

**{**

**public:**

//构造函数

**Book(unsigned int bookIdt,unsigned int peopleIdt,int bookClasst,float pricet,char bookNamet[45],**

**char publishert[25],char authort[25],char publishDatet[5],char borrowTimet[11],char borrowFlagt,**

**char isReservedFlagt,char isPreservedFlagt,char isLatedt,char isDeletedt,char isRealDeletedt);**

//复制函数

**Book(Book &t);**

//展示函数

**void displayBook();**

// unsigned int bookId;//identify number for books

**int setbookId(unsigned int bookId);**

**unsigned int getbookId();**

// unsigned int peopleId;//indentify number for readers

**int setpeopleId(unsigned int peopleId);**

**int getpeopleId();**

// int bookClass; //书籍分类,bookClass为1-22，分别对应22个分类,对应枚举变量bookClass-1

**int setbookClass(int bookClass);**

**int getbookClass();**

// float price;//价格

**int setprice(float price);**

**float getprice();**

// char bookName[45];//书名,最大44个字节，22个汉字

**int setbookName(char bookName[45]);**

**string getbookName();**

// char publisher[25];//出版社,24个字节，12个汉字

**int setpublisher(char publisher[25]);**

**string getpublisher();**

// char author[25];//作者，24个字节，12个汉字

**int setauthor(char author[25]);**

**string getauthor();**

// char publishDate[5];//出版年份，4个字节，只有年份，例如2009

**int setpublishDate(char publishDate[5]);**

**string getpublishDate();**

// char borrowTime[11];//time of lending the book,year xxxx month xx date xx hour xx

**int setborrowTime(char borrowTime[11]);**

**string getborrowTime();**

// char borrowFlag;//借出标志位

**int setborrowFlag(char borrowFlag);**

**char getborrowFlag();**

// char isReservedFlag;//预定标志位

**int setisReservedFlag(char isReservedFlag);**

**char getisReservedFlag();**

// char isPreservedFlag;//保存本标志位

**int setisPreservedFlag(char isPreservedFlag);**

**char getisPreservedFlag();**

// char isLated;//是否逾期

**int setisLated(char isLated);**

**char getisLated();**

// char isDeleted;//whether the record is delated(false as default)

**int setisDeleted(char isDeleted);**

**char getisDeleted();**

// char isRealDeleted;//是否真正删除

**int setisRealDeleted(char isRealDeleted);**

**char getisRealDeleted();**

// string getBookValue(BookType num){

// switch(num){

// case MaoDeng:

// return "马列主义、毛泽东思想、邓小平理论";

// case PhilosophyReligion:

// return "哲学宗教";

// case Social:

// return "社会科学总论";

// case PoliticalLaw:

// return "政治法律";

// case Military:

// return "军事";

// case Economic:

// return "经济";

// case Literature\_PE:

// return "文化、科学、教育、体育";

// case Language:

// return "语言、文字";

// case Literature:

// return "文学";

// case Art:

// return "艺术";

// case HistoryGeography:

// return "历史、地理";

// case Nature:

// return "自然科学总论";

// case MathChemistry:

// return "数理科学与化学";

// case Astronomy:

// return "天文学、地球科学";

// case Biology:

// return "生物科学";

// case Medicine:

// return "医药、卫生";

// case Agriculture:

// return "农业科学";

// case Industry:

// return "工业技术";

// case Transportation:

// return "交通运输";

// case Aerospace:

// return "航空、航天";

// case EnvironmentSafety:

// return "环境安全科学";

// case Comprehensive:

// return "综合性图书";

// default:

// return "获取信息出错";

// break;

// }

//

// }

**private:**

**unsigned int bookId;**

**unsigned int peopleId;**

**int bookClass;**

**float price;**

**char bookName[45];**

**char publisher[25];**

**char author[25];**

**char publishDate[5];**

**char borrowTime[11];**

**char borrowFlag;**

**char isReservedFlag;**

**char isPreservedFlag;**

**char isLated;**

**char isDeleted;**

**char isRealDeleted;**

**};**

**2.2** **使用的评估技术**

暂略

**2.3** **工作量、成本、时间估算**

成本：人力资源5人

时间：总时间为十周，截止至2017年12月15日（周五），首期完工尽量控制在八周以内。

工作量：小规模作业系统开发的代码量。

**第三章 进度评估**

过程中的不确定因素导致进度落后（需要将具体任务时间安排限定在8周之内完成，剩下的近两周时间用来以备不时之需）

**第四章** **项目进度计划**

**4.1 项目任务分解**

**前期准备工作：**

**2017/09/29** ——开发团队人员的确定

**2017/09/29**—— 确立队长以及文案负责人员，确定了“不抛弃、不放弃、组内所有成员都将参与到程序代码开发工作之中”的分工原则，讨论大体的工程任务方向，探讨图书管理系统所能具有的小组特色功能。

**2017/09/30**——小组成员共同前往学校图书馆，对学校的图书馆管理系统进行实地的仔细观察并结合组内将要共同完成的图书管理系统的情况进行可行性上的查漏补缺，同时拍摄多张校图书馆系统的记录照片来协助之后的策划书的调整安排。

**2017/10/07~10**——首次团队成员线下见面讨论，初步确定一些开发思想方向以及后续的开发任务分配，整理成第一版图书馆系统软件计划书。

**【后期进度有额外的附录文档进行记录】**

**第一阶段：构建具有初步功能的命令行图书馆管理系统。**

**（2017年10月11日到11月10日）**

1：输入输出与文件处理工作程序编程

包括初步的命令行测试界面设计

（计划时间：第一周，初步验收与连接测试时间：2017年10月20日）

(实际接收时间：2017年10月30日)

2：图书增删改查 与 账号的增删改查程序编程

（计划时间：第一周，初步验收与连接测试时间：2017年10月20日）

3：借书还书功能添加的程序编程

（计划时间：第二周，初步验收与连接测试时间：2017年10月27日）

4：具有（使用者与管理者的）用户程序编程

（计划时间：第二周，初步验收与连接测试时间：2017年10月27日）

5：形成可视化界面

（计划时间：第三、四周，初步验收与连接测试时间：2017年11月10日）

**第二阶段：特色功能的添加**

**（2017年11月11日到12月01日）**

具体时间安排以及分工待第一阶段完成后再按实际情况考量与讨论设计。

（详情请看第二阶段的图书馆系统软件计划书）

**第三阶段：系统的漏洞测试以及完善工作**

**（2017年12月02日到12月15日）**

具体时间安排以及分工待第二阶段完成后再按实际情况考量与讨论设计。

（详情请看第二阶段的图书馆系统软件计划书）

**第五章** **细则要求问题**

1：建议每位成员需要有自己的工作日志，用于记录在整一个开发过程自己所承担任务以及编程设计之中做的变动及具体日期。（word or txt or else格式都OK）

2：编程过程中命名规则需要符合多数的计算机程序开发规范

3：尽量多写注释帮助代码连接时查找问题以及互相调试////起码每个函数里要有一句一上注释

4：编程过程中多沟通\*3（表强调）。//比如数据结构、数据类型应该怎么样更合适，是没法一开始就直接空想出最准确的方式，需要边写边发现，慢慢讨论调整。

5：做出接口（或者其他代码要素）的版本更新时，需要直接在原接口上直接改，文字声明也需要另外写出来，比如“将people 中的rank改名level”，需要直接在原来的struc里面也更改，再在下方做文字更新说明，方便其他人员的直接更改。（10/16添加）

6：建议每个成员在开发时将其他成员提交的资料按名字以及负责任务部分分文件夹放，按时间排序，方便使用。

**第六章** **软件配置**

IDE: codeblocks

可视化：QT（初步计划，未定）

**第七章** **人员组织**

**第一阶段：构建具有初步功能的命令行图书馆管理系统。**

**第一部分：**

1：输入输出与文件处理工作程序编程

**：丛爽**

2：图书增删改查 与 账号的增删改查程序编程

**增：刘真**

**删：徐云哲**

**改：颜晓书**

**查：张文政**

**数据截取：燕晨放**

**第二部分：**（待第一阶段第一部分完成后视个人情况再分配）

3：借书还书功能添加的程序编程

4：具有（使用者与管理者的）用户程序编程

**：丛爽**

**：刘真**

**：徐云哲**

**：颜晓书**

**：张文政**

**：燕晨放**

**第三部分：**（待第一阶段第二部分完成后视个人情况再分配）

5：形成可视化界面

**第二阶段：特色功能的添加**

具体时间安排以及分工待第一阶段完成后再按实际情况考量与讨论设计。

**第三阶段：系统的漏洞测试以及完善工作**

具体时间安排以及分工待第二阶段完成后再按实际情况考量与讨论设计。