基于C++的“书海”图书管理系统

需求说明书

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <14/10/2017> | <1.0> | 初步完成需求分析 | 张钦贤 |

**目录**

[1. 引言 3](#_Toc20941)

[1.1 系统概述 3](#_Toc2703)

[2. 功能模块需求分析 3](#_Toc17610)

[2.1 公共界面 3](#_Toc32135)

[2.2 学生用户界面 3](#_Toc7732)

[2.3 管理员界面 4](#_Toc6129)

[3. 实体说明及E-R图 4](#_Toc8735)

[3.1 实体说明 4](#_Toc15696)

[3.2 E-R图 6](#_Toc7870)

[4. 系统体系架构 6](#_Toc226)

[4.1 实体类库（Model） 6](#_Toc10463)

[4.2 数据访问层（DAL） 6](#_Toc24729)

[4.3 业务逻辑层（BLL） 6](#_Toc6533)

[4.4 表现层（UI） 6](#_Toc16613)

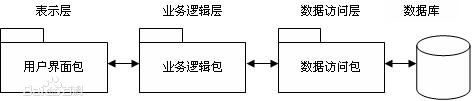
[5. 成员分工 6](#_Toc15629)

# 引言

## 系统概述

本项目为中国海洋大学“UML面向对象分析”这门课中的实践项目，要求以二进制文件为存储媒介，用C/C++开发一款图书管理系统，能够实现借书、还书、预约、罚款、对图书的增删改查等功能，并且能够处理10w级别的数据。

本项目采用三层体系结构：数据访问层（DAL）、业务逻辑层（BLL）、表示层（UIL）。开发人员可以只关注整个结构中的其中某一层，很容易的用新的实现来替换原有层次的实现。实现流程如下图所示：



# 功能模块需求分析

## 2.1 公共界面

**登录界面**

1. 此界面为程序开始界面。系统不需要注册用户，学生用户使用11位学号进行登录，初始密码为学号；系统管理员使用7位管理员账号登录，并提供有忘记密码找回功能，找回信息验证可以采取邮箱验证或者短信验证码验证。

## 2.2 学生用户界面

**个人信息界面**

1. 此界面展示学生用户的姓名、学号、专业、性别、邮箱、头像等个人基本信息，以及当前借阅、预约、超期的图书信息，还有当前欠款额度。
2. 图书信息显示书名、作者、出版社三项；点击当前借阅的图书可以进行还书操作，点击预约的图书可以进行借书、取消预约操作，点击超期的图书可以进行还书操作。

**个人信息维护**

1、此界面用于学生个人信息维护，可以修改个人邮箱和头像，可以修改密码，支付欠款。

**图书查询界面**

1. 此界面用于查询图书，查询条件可以是书名、作者、ISBN、出版社，分类、所有查询均为模糊查询。查询结果显示在列表中，列表具有分页功能。

**图书详情界面**

1. 在图书查询界面点击某一本图书，会跳到图书详情界面。该界面展示图书的书名、作者、ISBN、出版社、分类，在此页面可以进行借书、预约等操作。
2. 借书时限为30天，超期会处罚金，超期一天罚金0.1元，当欠款额度多于10元时，不可执行借书、预约操作；预约时限为10天，10天内不取，系统自动取消预约。

## 2.3 管理员界面

**个人信息界面**

1、此界面展示管理员的姓名、账号、邮箱、头像等个人基本信息，并可以进行修改，还能够进行密码修改，退出登录等操作。

**图书查询界面**

1. 同2.1中的“图书查询界面”。

**增加图书界面**

1. 管理员可以增加图书，增加图书时，需要输入图书的书名、作者、ISBN、出版社，并选择分类。

**图书详情操作界面**

1. 在图书查询界面点击某一本图书，会跳到图书详情操作界面。与2.1中的“图书详情界面”不同的是管理员可以在此页更新或者删除选中的图书。

**账号查询界面**

1、管理员可以查询学生账号，查询条件可以是学号、姓名、专业。查询为精确查询，查询结果显示在列表中，列表具有分页功能。

**增加账号界面**

1. 管理员可以增加学生账号，增加账号时，需要输入学号、姓名、专业、性别等信息。

**账号详情操作界面**

1. 在账号查询界面点击某一账号，会跳到账号详情操作界面，可以修改账号信息、重置密码、删除账号。

**分类管理界面**

1、此界面可以添加分类、删除分类、修改分类。

# 实体说明及E-R图

## 实体说明

本系统共有如下实体：

1. 书籍（book）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **字段名（英文）** | **字段名（中文）** | **字段类型** | **键** | **备注** |
| **1** | id | 编号 | INT | P | 自增，作为标识 |
| **2** | ISBN | 国际标准书号 | CHAR(20) |  |  |
| **3** | name | 图书名称 | CHAR(100) |  |  |
| **4** | author | 图书作者 | CHAR(100) |  |  |
| **5** | cover | 封面 | CHAR(100) |  | 采取路径的方式存储图片 |
| **6** | publishing | 出版社 | CHAR(100) |  |  |
| **7** | classifyId | 分类号 | CHAR(100) | FK | 作为外键同classify表相关联 |

1. 学生用户（student）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **字段名（英文）** | **字段名（中文）** | **字段类型** | **键** | **备注** |
| **1** | id | 编号 | INT | P | 自增，作为标识 |
| **2** | usercode | 学号 | CHAR(20) |  |  |
| **3** | username | 姓名 | CHAR(100) |  |  |
| **4** | sex | 性别 | CHAR(50) |  |  |
| **5** | icon | 头像 | CHAR(100) |  | 采取路径的方式存储图片 |
| **6** | mail | 邮箱 | CHAR(100) |  |  |
| **7** | forfeit | 罚金 | FLOAT |  |  |

1. 管理员用户（admin）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **字段名（英文）** | **字段名（中文）** | **字段类型** | **键** | **备注** |
| **1** | id | 编号 | INT | P | 自增，作为标识 |
| **2** | account | 账号 | CHAR(20) |  |  |
| **3** | mail | 邮箱 | CHAR(100) |  |  |

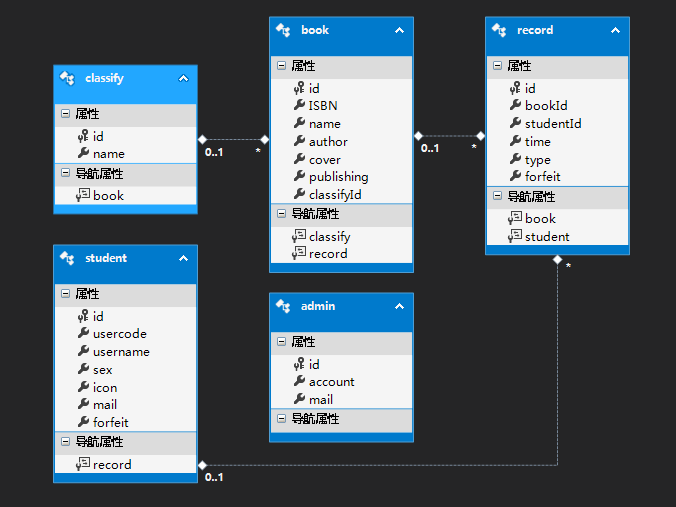
1. 借阅记录（record）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **字段名（英文）** | **字段名（中文）** | **字段类型** | **键** | **备注** |
| **1** | id | 编号 | INT | P | 自增，作为标识 |
| **2** | bookId | 所借图书id | INT | FK | |  | | --- | | 作为外键同book表相关联 | |
| **3** | studentId | 借书学生id | INT | FK | |  | | --- | | 作为外键同student表相关联 | |
| **4** | time | 借书时间 | CHAR(50) |  |  |
| **5** | type | 记录类型 | INT |  | 1-已借；2-预约；3-已归还 |
| **6** | forfeit | 超期罚金 | FLOAT |  |  |

1. 图书分类（classify）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **字段名（英文）** | **字段名（中文）** | **字段类型** | **键** | **备注** |
| **1** | id | 编号 | INT | P | 自增，作为标识 |
| **2** | name | 分类名称 | CHAR(50) |  |  |

## E-R图



# 系统体系架构

## 4.1 实体类库（Model）

将文件实体利用ORM原理，映射为C++中的类，类的成员变量为实体字段，类的方法为访问实体的get和set方法。

## 4.2 数据访问层（DAL）

存放对数据类的访问，即对文件数据的添加、删除、修改、更新等基本操作。将这四种操作统一封装公共类FileDB中，能够针对不同的实体，进行增删改查操作。

## 4.3 业务逻辑层（BLL）

调用DAL方法，访问Model，获取数据，并进行逻辑判断，数据处理等。比如判断用户名密码是否正确，判断罚金是否超出限额等。

## 4.4 表现层（UI）

显示数据，与用户进行交互，以取得良好的用户体验。本项目UI层采用C++QT开发。

# 成员分工

张钦贤：系统体系架构设计，实体类库（Model）设计，数据访问层（DAL）开发（查找数据、增加数据），成员分工，进度安排。

李策：数据访问层（DAL）开发（修改数据，删除数据），业务逻辑层（BLL）开发。

周尊康：UI设计，系统架构设计，QT界面开发。

王蕊：QT界面开发，业务逻辑层（BLL）开发，文档编写。

贲志强：QT界面开发，业务逻辑层（BLL）开发，系统测试。