大连东软信息学院

**毕业设计（论文）**

**论文题目**：**基于MVC的\*\*系统的**

**设计与实现**

|  |  |
| --- | --- |
| **学 院：** | 高等职业技术学院 |
| **专 业：** | 软件技术系 |
| **学生姓名：** | 张邦宸 |
| **学生学号：** | 20107310106 |
| **指导教师：** |  |
| **导师职称：** |  |
| **完成日期：** |  |

大连东软信息学院

**Dalian Neusoft University of Information**

基于MVC的\*\*系统的

设计与实现

# 摘 要

随着社会节奏在不断的加快，人们也从生活中体会到了各种各样的压力，其中毕业生就业压力最为明显。按目前的状况分析，普遍存在着招聘单位与求职者的一些关键信息的流失或信息的过时，无法达到信息的及时更新和共享，而职位中介机构面临的主要问题就是工作效率不高。

为了解决目前存在的问题，本课题在分析了毕业生求职应聘现状以及对现有的毕业生求职应聘管理系统进行研究比较的基础上，针对毕业生求职应聘的特殊性，设计研发了一套基于MVC的毕业生求职应聘管理系统。本系统的集成开发环境是MyEclipse，使用MySQL作为数据库管理系统，Web服务器采用Tomcat，运用MVC思想，采用Struts2框架技术实现。为了充分利用网络信息化的优势，招聘系统采用B/S结构。系统具有招聘职位管理、职位搜索、投递简历、新闻资讯、留言模块管理等功能，初步完成求职招聘管理系统的要求。招聘信息的录入全部在网上进行，招聘过程中所涉及的数据都在招聘信息库中进行传递和维护，因此数据的填写、更新都不需要以纸质方式进行处理，而是直接通过网络进行传递。

本系统的实现能够极大的提高求职应聘机构的工作效率，并达到求职信息和岗位需求信息的及时更新和实时共享。系统运行稳定，并且有良好的易用性，能够充分满足求职应聘者和中介机构的需求。

**关键词**：求职应聘，MVC，Struts2，B/S

说明：以上文字必须修改为自己的毕设系统的相关摘要文字，不要出现雷同！建议采用三段式写法：第一段描述背景与意义，文字占摘要总体文字比例不超过30%，一般3-4行为宜；第二段阐述论文主要完成的工作以及所使用的技术与工具，是摘要的主体，文字占摘要总体文字比例50%左右；第三段，描述结论，对效果进行评价，总结优缺点及改进方向，文字比例20%左右，3-4行为宜。中文摘要语句通顺，与论文正文内容相符；摘要中不要使用“将来时”语气。

**Design and Implementation of \*\* System based on MVC**

# Abstract

With moving increased competition of the operator, the new business demand is emerging constantly, as society continues to accelerate the rhythm, people experience from life to a variety of pressures, including pressure on employment of graduates is most obvious. An analysis of the current work status, the loss of some key information on the prevalence of candidate units and job seekers, and a common problem is the agency working efficiency is not high.

This paper analyzed the status of job applicants and graduates of existing management systems graduate research job candidates on the basis of comparison for graduate job candidates particularity, design and development of a graduate job candidates based on MVC Management systems. The system development environment is MyEclipse, the background using MySQL as the database management system, web server using Tomcat, use the MVC pattern to develop ideas, using Struts2 framework technology. In order to fully utilize the advantages of network information, recruitment system uses B/S structure. System with Job management, job search, resume, news, message module management functions, initially completed the required job recruitment management system. All entry jobs online, the data involved in the recruitment process are carried out in the recruitment repository circulation, maintenance, and therefore the data fill in paper form are not required to update processing, passing directly through the network.

The system can greatly enhance work efficiency of job-introduction agency. It can timely update and share the career information and position requirements information. The system runs stability and has good usability.  It can fully meet the applicant and the job-introduction agency demand.

**Key words:** Job candidates, MVC, Struts2, B/S

说明：与中文摘要对应，无明显语法错误，坚决杜绝直接使用软件翻译的情况！

目 录

摘 要 I

Abstract II

第1章　项目概述 1

1.1 问题描述 1

1.2 项目目标 1

1.3 项目适用范围 1

1.4 项目应遵守的规范与标准（可选） 1

1.5 涉众 1

第2章　需求分析 2

2.1 业务需求（可选） 2

2.1.1 企业组织 2

2.1.2 业务流程 2

2.1.3 业务对象 2

2.2 功能性需求 4

2.2.1 用例概述 4

2.2.2 用例描述 5

2.3 非功能性需求 7

2.4 项目约束（可选） 7

2.5 运行环境 7

第3章　系统设计 9

3.1 技术路线 9

3.2 软件体系结构设计 9

3.2.1 功能结构 9

3.2.2 系统架构 10

3.3 外部接口（可选） 11

3.3.1 硬件接口 11

3.3.2 软件接口 11

3.3.3 其它接口 11

3.4 数据库设计 11

3.4.1 概念结构设计 11

3.4.2 表的详细设计 12

3.5 对象设计 12

3.5.1 借阅用例对象设计 12

3.5.2 \*\*用例对象设计 15

3.6 界面设计 15

3.6.1 界面设计规范 15

3.6.2 界面布局设计 15

3.7 \*\*设计（可选） 16

第4章　系统实现 17

4.1 核心功能实现 17

4.1.1 系统主界面 17

4.1.2 \*\*界面 18

4.2 关键技术难点与解决方案 18

4.3 关键算法（可选） 18

4.4 编码规范 18

第5章　系统测试 19

5.1 功能测试 19

5.2 性能测试（可选） 20

5.3 测试总结 20

第6章　结论与展望 22

参考文献 23

致 谢 24

# 第1章　项目概述

## 1.1 问题描述

在用户移动本地存储空间成本昂贵的大背景下，网络云存储成为了用户首选的移动存储方式，可以满足用户的工作、学习、生活各类需求。

## 1.2 项目目标

网络存储系统的开发目标是按照信息管理系统的管理和开发方法，用我们所掌握的技术进行系统开发。运用现代快速移动网络技术实现随时随地存储读取信息，避开了移动存储成本高，存储移动性弱的缺点。

## 1.3 项目适用范围

家庭办公，家庭存储，个人存储。

## 1.4 项目应遵守的规范与标准

YD/T 2391-2011

## 1.5 涉众

0级用户：管理人员（Administrator）管理审核人员，和云存储用户的数据。

1级用户：云存储用户（NormalUser）日常办公，生活存储。

2级用户：游客（Tourist），仅有浏览权限。

服务器维护人员，运行维护人员。

# 第2章　需求分析

## ~~2.1 业务需求（可选）~~

### 2.1.1 企业组织

说明：给出企业的组织架构图。

### 2.1.2 业务流程

说明：描述当前的业务流程，可用活动图表示。

完成一项任务所需要经历的各部门（或人员）之间进行协作的步骤。对于复杂的业务逻辑进行图形化建模有利于对业务之间的关系进行理解。例如国际合作项目的总体业务流程。而对于一些简单的信息展示项目来讲，可能不存在业务流程。例如，主要针对资料整理和展示的英文学习网站。

举例：

国际合作项目的总体业务流程如图2.1所示。



图2.1 总体业务流程图

### 2.1.3 业务对象

说明：从业务流程中提取业务对象，可画领域模型。

举例：

领域模型如图2.2所示。



图2.2 领域模型

## 2.2 功能性需求

### 2.2.1 用例概述

在图2.3高层用例图中抽取了满足用户基本业务需求的基本用例。这些用例可以从用户服务、审核服务、系统管理三个方面完成整个网络存储系统的正常运作。

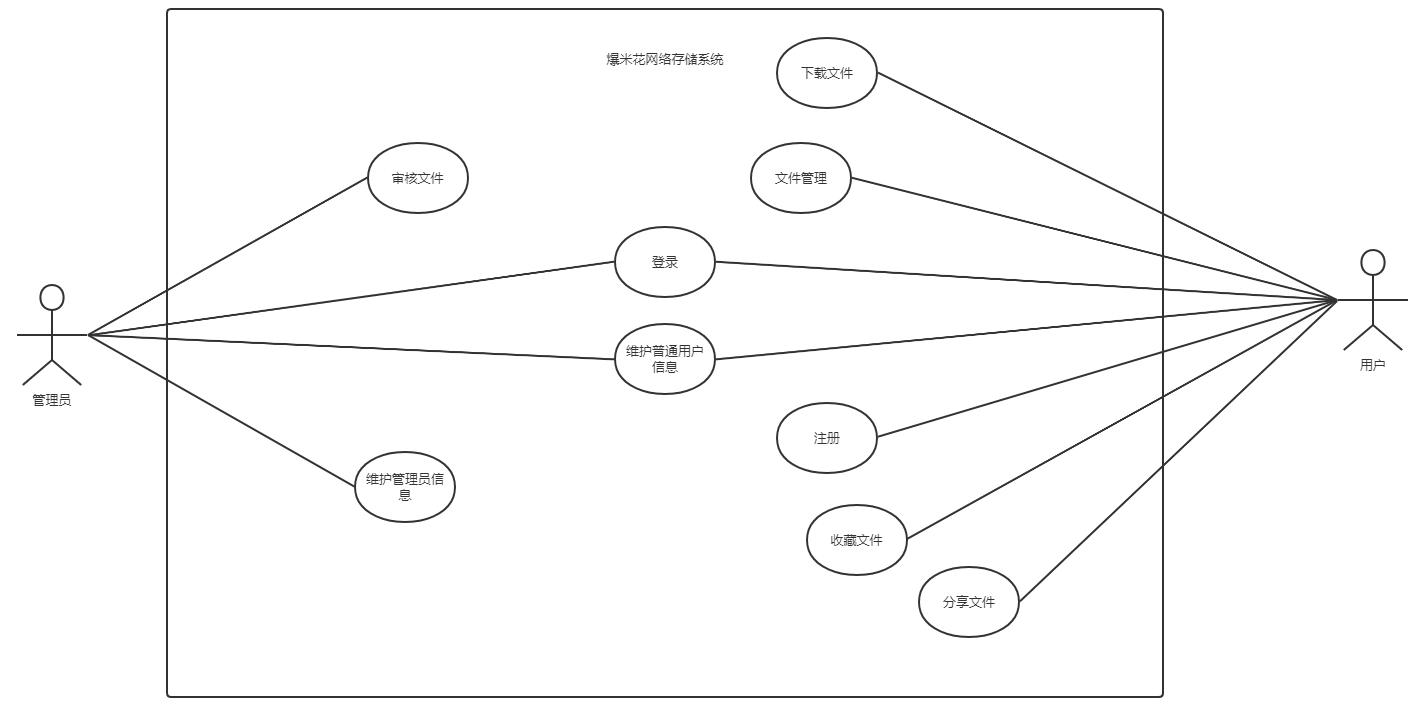


图2.3 高层用例图

用例摘要描述如表2.1所示。

表2.1 用例摘要描述

|  |  |  |
| --- | --- | --- |
| **用例标识（UC）** | **用例名称** | **摘要描述** |
| 0 | 登录 | 使用完整系统前的身份验证。 |
| 1.1 | 下载文件 | 普通用户可以从系统下载自己的文件 |
| 1.2 | 文件管理 | 普通用户可以对系统中的文件进行上传，替换，查看和删除操作 |
| 1.3 | 维护普通用户信息（普通用户） | 普通用户可以填写补充、修改、删除和查看个人信息 |
| 1.4 | 注册 | 普通用户使用完整系统前的信息注册 |
| 1.5 | 收藏文件 | 普通用户可以对系统中的个人文件进行标注 |
| 1.6 | 分享文件 | 普通用户可以把自己的个人文件通过生成链接分享给站外用户 |
| 2.1 | 维护普通用户信息（管理员） | 管理员可以管理普通用户账号，定期清理长时间不活跃用户数据。 |

续表2.1 用例摘要描述

|  |  |  |
| --- | --- | --- |
| 2.2 | 维护管理员信息 | 管理员可以填写补充、修改、删除和查看信息。 |
| 2.3 | 审核文件 | 管理员负责审核普通用户上传的文件和文件备注内容。 |

### 2.2.2 用例描述

说明：如果存在对用例的细化，则可在此列出细化用例图。

举例：

（1）管理文件用例

用户可以通过管理文件用例对图书信息进行维护，包括对文件的上传，删除，更新和查看。管理文件用例图如图3.4所示。

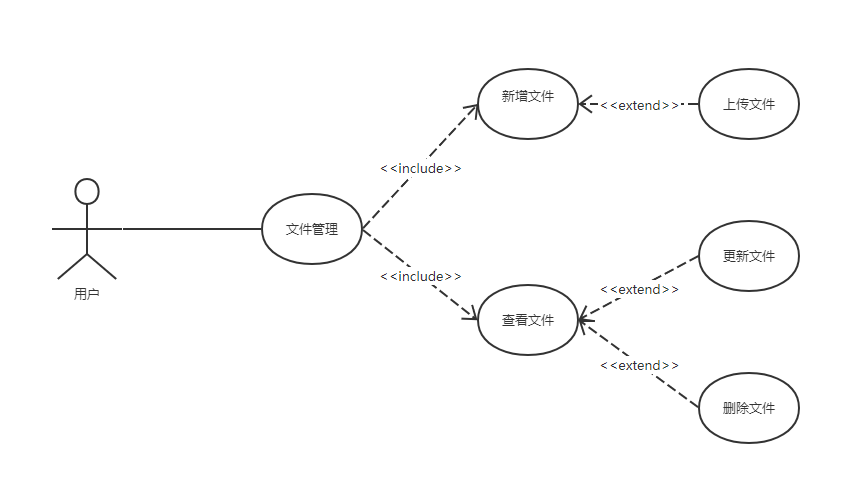


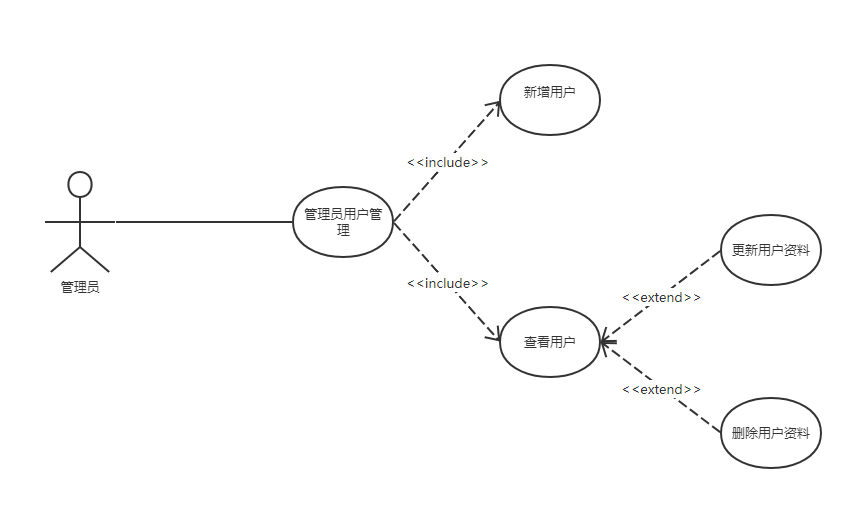
图2.4 文件管理用例图

文件管理用例描述如表2.2所示。

表2.2 文件管理用例

|  |  |
| --- | --- |
| **用例标识** | UC2.3.1 |
| **用例名称** | 新增文件 |
| **参与者** | 普通用户 |
| **前置条件** | 已经以普通用户身份登录。 |
| **后置条件** | 在用户文件表中添加一条数据并在服务器中新增文件 |
| **用例概述** | 用户向服务器上上传一个文件 |
| **基本事件流** | 1. 用户在用户文件管理页面选择需要上传至的目录。 2. 系统跳转至该目录。 3. 用户点击用户文件管理页面的“上传文件”按钮。 4. 系统显示上传界面。 5. 用户选择所需上传的文件，并提交上传请求。 6. 系统根据文件名确认没有重复文件，自动生成文件编号、编辑时间、操作人和存放目录存入用户文件表中。 7. 系统显示文件上传成功。 |
| **备选事件流** | 2a. 若系统不存在该目录则新建目录，执行完毕转入步骤3  6a. 若存在重复文件提示用户是否替换该文件，执行完毕转入步骤7 |
| **备注** | 文件编号生成规则：时间戳+用户名 |

（2）业务员信息管理用例



## 2.3 非功能性需求

说明：可用文字描述。包括：性能、并发、兼容性、安全性

举例：

正确性：不允许出现业务错误。

健壮性：在系统出现故障时，系统必须提供相应的数据保护措施，不能丢失数据，并提供重新恢复正常工作的机制。

性能：系统处理业务时间正常业务量的情况下不超过2秒，繁忙业务情况下最迟时间不可超过5秒。

安全性：读者信息具有保密性，读者信息只能由图书馆内部人员使用。

兼容性：可运行在Windows10以上操作系统中。

## ~~2.4 项目约束（可选）~~

说明：为了保证项目的顺利实施而保证一些环境或条件已经满足。

例如，做网上交易系统，则需保证已经获得第三方支付平台的接口。

## 2.5 运行环境

说明：给出本软件系统运行所需的基本软/硬件环境、使用本软件系统典型用户的设备分布图及设备上相应软件配置。软件环境包括：操作系统、数据库、以及其它支撑软件；硬件环境包括：主机类型、网络类型、存储器容量、其它特殊设备。

举例：

本软件系统运行所需的硬件环境如下。

（1）主机类型

应用服务器：SUN E220，单CPU，1G RAM

数据库服务器：SUN E220，单CPU，1G RAM

工作站：LEGEND PC，256M RAM

（2）网络类型

以太网

（3）存贮器容量

应用服务器：20G以上

数据库服务器：100G以上

（4）其他特殊设备

打印机：HP 6L

（5）设备分布图，如图2.5所示。



图2.5 网络拓扑图

本软件系统运行所需的软件环境如下。

（1）操作系统

应用服务器：Windows NT Server 2003以上版本

工作站：windows XP以上

（2）数据库管理系统

BEA Oracle Enterprise 9i

# 第3章　系统设计

## 3.1 技术路线

说明：给出本软件系统在实现过程中所采用的关键技术。对比较新的技术给予详细的阐述及必要的说明，在可能的情况下给出相应的技术参考资料。可包括：描述硬件、软件开发环境。

## 3.2 软件体系结构设计

### 3.2.1 功能结构

根据需求调研结果确定本系统主要包括以下功能模块，如图3.1所示。

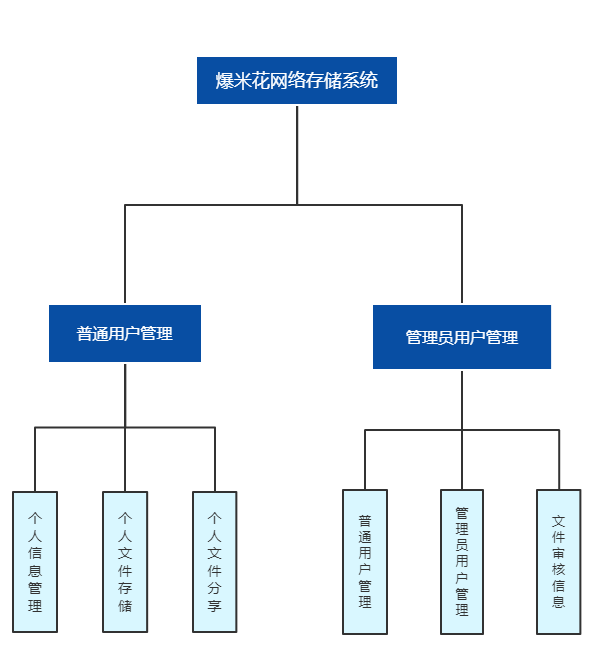


图3.1系统功能结构图

### 3.2.2 系统架构

说明：给出系统架构图，并进行描述。

系统架构图如图3.2所示。

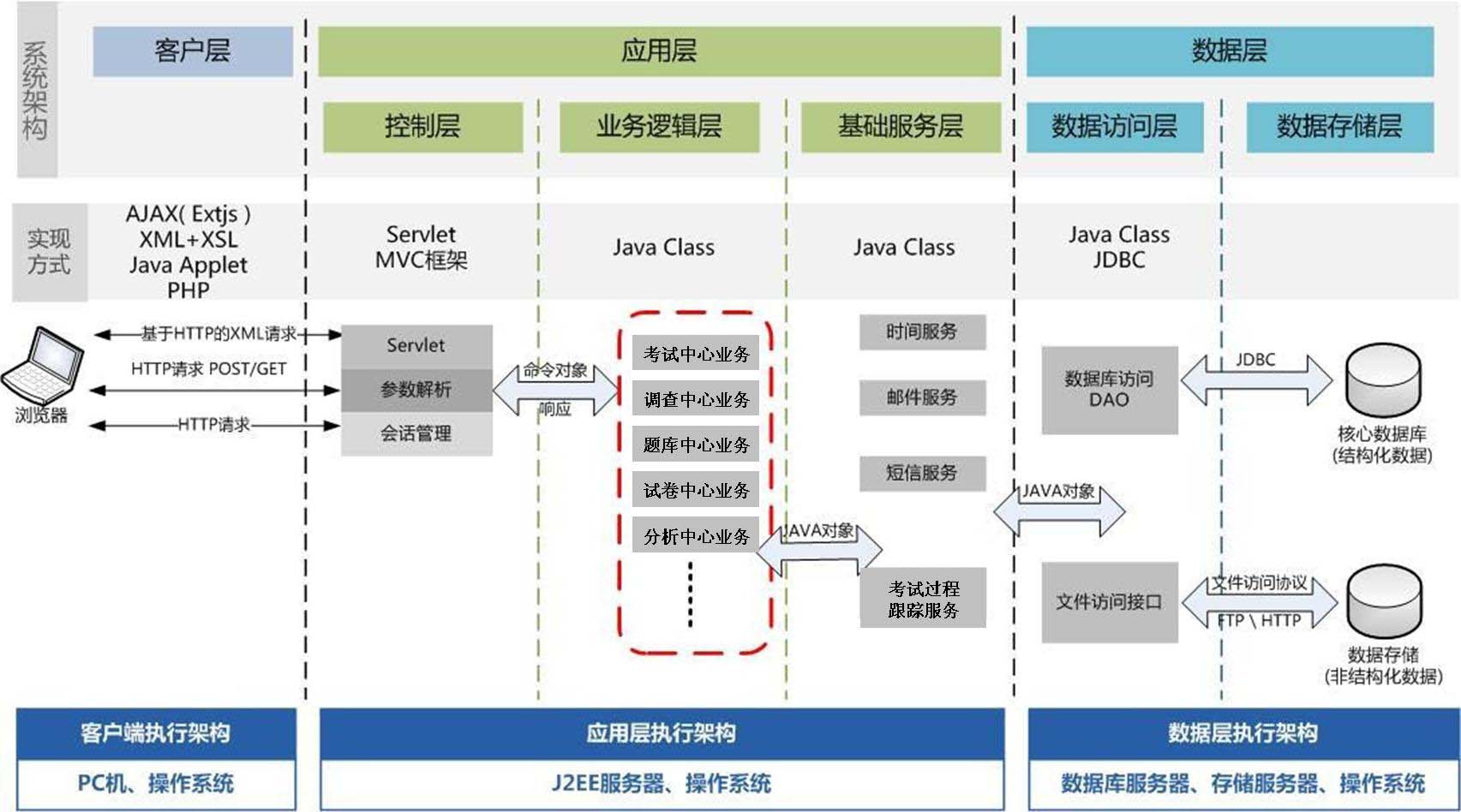


图3.2 系统架构图

## ~~3.3 外部接口（可选）~~

### 3.3.1 硬件接口

说明：描述该软件与外部设备接口。

### 3.3.2 软件接口

说明：描述该软件与其它系统的软件接口，主要说明其它系统提供的编程接口和本系统向其它系统提供的编程接口。

### 3.3.3 其它接口

说明：描述该软件与其它系统的其它类型接口，主要说明其它系统提供的和本系统向其它系统提供的硬件接口、通信协议及规程、数据交换格式等。

## 3.4 数据库设计

说明：对数据库设计中涉及到的各种项目，如数据项、记录、表、数据库等说明它们的标识符、同义名及有关信息。

### 3.4.1 概念结构设计

说明：给出概念模型或ER图。

概念模型（ER图）如图3.3所示。

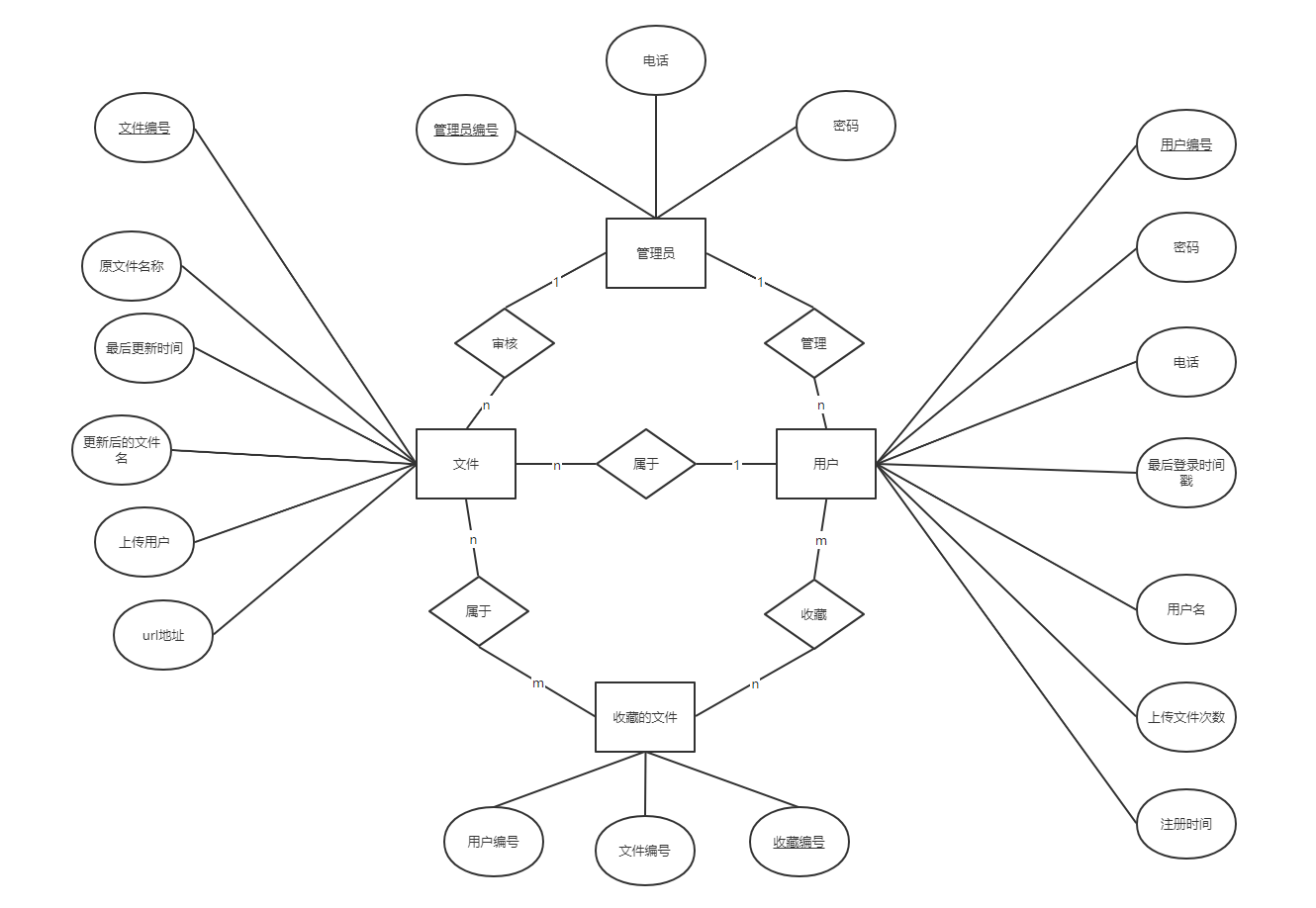


图3.3 ER图

### 3.4.2 表的详细设计

举例：

通过对领域对象、业务对象及对象间关联的分析，对本系统的数据库表进行了设计，如表3.1--表3.8所示。本系统数据库管理采用MySQL。

（1）文件表(file\_table)

文件表用于存储服务器中的文件信息，主要字段有文件编号、文件更新时间戳、文件url、用户编号、文件名，表结构如表3.1所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| file\_table(文件表) | | | | |
| **属性名** | **类型（长度）** | **是否空** | **主键** | **备注** |
| file\_id | int(10) | not null | PRI | 文件的id，在上传成功后自动产生 |
| file\_timestamp | int(10) unsigned | not null |  | 文件最后更新时间戳 |
| file\_url | varchar(255) | not null |  | 附件的url地址 |
| user\_id | int(10) unsigned | not null |  | 上传文件用户的id |
| file\_name | varchar(255) | not null |  | 用于保存用户上传文件的文件名 |
| file\_old\_name | varchar(255) | not null |  | 用于保存用户上传文件的原始文件名 |

表3.1 file\_table(文件表)

（2）用户表(user\_table)

用户表用于存储服务器中的用户信息，主要字段有用户编号、用户密码、用户名、用户电话、用户登录时间戳、用户注册时间戳、上传文件次数，表结构如表3.2所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| user\_table（用户表） | | | | |
| **属性名** | **类型（长度）** | **null** | **主键** | **备注** |
| user\_id | int(10) unsigned | not null | PRI | 用户id |
| user\_password | varchar(20) | not null |  | 用户密码 |
| user\_name | varchar(45) | not null | UNI | 用户名 |
| user\_phone | int(10) unsigned | not null |  | 用户电话号码 |
| user\_last\_login \_timestamp | int(10) unsigned | not null |  | 用户最后登录时间戳 |
| user\_register  \_timestamp | int(10) unsigned | not null |  | 用户注册时间戳 |
| user\_update\_time | int(9) unsigned | not null |  | 用户上传文件次数 |

表3.2 user\_table(用户表)

（3）管理员表(admin\_table)

管理员表用于存储服务器中的管理员用户的信息，主要字段有管理员编号、管理员密码、管理员电话，表结构如表3.3所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| admin\_table(管理员表) | | | | |
| **属性名** | **类型（长度）** | **null** | **主键** | **备注** |
| admin\_id | int(10) unsigned | not null | PRI | 管理员id |
| admin\_password | varchar(20) | not null |  | 管理员密码 |
| admin\_number | int(10) unsigned | not null |  | 管理员电话号码 |

表3.3 admin\_table(管理员表)

（4）收藏表(collect\_table)

收藏表用于存储服务器中的用户收藏文件的信息，主要字段有用户编号、文件编号、收藏编号，表结构如表3.4所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| collect\_table(收藏表) | | | | |
| **属性名** | **类型（长度）** | **null** | **主键** | **备注** |
| users\_id | int(10) unsigned | not null | PRI | 用户id |
| file\_id | int(10) unsigned | not null |  | 文件id |
| collect\_id | int(10) unsigned | not null |  | 收藏文件的id |

表3.4 collect\_table (收藏表)

## 3.5 对象设计

说明：说明用例的实现方式，可画顺序图。给出核心模块的详细设计。

本系统的功能包括登录、注册、…….等。下面对各个模块进行详细设计。

### 3.5.1 借阅用例对象设计

说明：

（1）要画出本用例的设计类图。

（2）说明用例的实现方式，可画顺序图。

（3）如果有复杂算法的话，就给出算法设计（可选）

举例：

（1）用例实现（可选）

借阅用例的实现顺序图如图3.4所示。



图3.4借阅用例顺序图

说明：对以上顺序图加以文字说明。

（2）设计类图

根据顺序图的分析得到借阅用例的设计类图，如图3.5所示。



图3.5 借阅用例的设计类图

类图中Controller包中类的描述如表3.2所示。

表3.2 Controller包中类的描述

|  |  |  |
| --- | --- | --- |
| **类名** | **属性/方法** | **描述** |
| ReaderServiceController | +doPost() | 获取页面请求 |
| +response() | 返回处理结果页面 |
| -getReaderInfo() | 获取读者信息 |
| -borrowBook(bookID: String, cardID: String): Integer | 读者借阅哪本书，并返回借阅结果。1：成功，0：失败。 |

类图中BLL包中类的描述如表3.3所示。

表3.3 BLL包中类的描述

|  |  |  |
| --- | --- | --- |
| **类名** | **属性/方法** | **描述** |
| BookBusinessService | reserveBook(String bookID) | 预订图书 |
| borrow(String bookID,String cardID) | 根据书号和卡号借书 |
| checkValid() |  |
| return(String bookID,String cardID) | 根据书号和卡号归还图书 |
| isOverdue(String cardID,String bookID) | 判断是否超期 |
| sendRemindEmail() | 发送超期提醒邮件 |
| orgnizedEmail(String msg) | 组织邮件内容 |
| isOverBorrowLimit(String cardID) | 判断是否超过借阅本书限制 |

类图中Domain包中类的描述如表3.4所示。

表3.4 Domain包中类的描述

|  |  |  |
| --- | --- | --- |
| **类名** | **属性/方法** | **描述** |
| BookCopy | String copyID | 图书编号 |
| int status | 图书状态。1：在库，2：借出，3：典藏，4：修补 |
| void updateBookCopy (String copyID,int status) | 更新图书拷贝信息 |
| Book | String bookID |  |
| String ISBN |  |
| String title |  |
| String author |  |
| String publisher |  |
| duble price |  |
| int inventory |  |
| int getInventory(String bookID) | 根据书号获取库存量。 |
| void updateBookInfo(Book book) | 更新图书信息 |
| Reader | readerID |  |
| name |  |
| gender |  |
| email |  |
| idendityCard |  |
| Reader findReader(String readerID) | 查找读者信息 |
| addReader(Reader reader) | 增加读者信息 |
| updateReader(Reader:reader) | 更新读者信息 |
| String getReaderEmail(String readerID) | 获取读者邮件地址 |

### 3.5.2 \*\*用例对象设计

说明：格式同上。

## 3.6 界面设计

### 3.6.1 界面设计规范

说明：列出界面设计过程中遵循的规范原则。

举例：

本系统是商家用于发布软件商品信息，用户通过注册进行选购软件商品从而下订单，所以，我们设计界面时要按照以下原则：

（1）界面上控制字体以及文本样式，如，字体大小和颜色等，因为顾客会因为太多的文字而影响其注意力。但是与此同时图片过多，顾客会觉得眼花缭乱，把它们当做广告，并且忽略掉不看。

（2）页面要让顾客容易阅读，这说明，文字以及页面的底面颜色要着重考虑。尽量避免文字颜色和底面颜色接近这个问题，同时页面上的颜色不要过于丰富，对于大部分人来说，浅色底面配上深色字体让人觉得舒服。同时，文字大小最好不大不小。按照人们日常生活的阅读习惯，文字基本都是左起居左。当然，文章的题目是要居中的，这符合大众的习惯。

（3）用对比度高的字体和底面背景颜色，这样可以凸显文字。

（4）页面并非长的好，最好是在一个半的版面内，因为页面过长会让人感觉很臃肿，很杂乱。

（5）此外，页面的容量尽量偏小，尽量不要大于75Kb，因为如果一个网站的页面主题在一定时间内无法显示出来，用户会立刻去看别的东西。

（6）页面用动态页面设置，这样可以避免界面更改大小的时候而影响到其质量

（7）网站的导航要让用户一目了然，用户可以迅速的找到网站上的超链接，页面上要有清晰的导航标识，便于用户辨认。

（8）网站的设计风格要尽量保持一致，即文字，图片，底面颜色，标题，花纹等风格要一致，纵观网站让人觉得是一个整体，这样会让客户觉得舒服，整体布局连贯性好。

### 3.6.2 界面布局设计

说明：介绍界面布局设计的思路。

举例：

本系统采用的是右包含式的布局，因为系统需要，在界面左边加上了导航条。所以进行布局设计时，使整体左对齐，在加上左边的导航条时，左右不会显得对称失调。

但是，同时出现了一个问题，就是使页面的右侧显得苍白。这个问题我本来打算用添加浅底色进行补充，但颜色的搭配比较困难，为了不至于弄巧成拙，所以就放弃了这个想法。当然，这样也有一个好处，就是使页面显得重点突出。

......

## ~~3.7 \*\*设计（可选）~~

说明：列出其他方面的设计，如安全性等。例如，说明在数据库的设计中，如何通过区分不同的访问者、不同的访问类型和不同的数据对象，进行分别对待而获得的数据库安全保密设计考虑。

# 第4章　系统实现

## 4.1 核心功能实现

### 4.1.1 系统主界面

说明：描述核心业务用例的实现，说明从前台到后台整体的实现方案。用文字+实现效果+代码的方式，把每一层之间的相互调用，使用什么样的关键代码做以解释。

举例：

在IE里运行本系统，出现系统的主界面，包括系统登陆、软件商品类别、软件商品信息显示列表、新闻信息列表四大模块。如图4.1所示：



图4.1 系统主界面

相关代码如下：

// 在此处放置用户代码以初始化页面

DataTable tmpda = new DataTable();

tmpda = DataBase.Get\_Table("select \* from SMLB where Plxbh=''");

if (tmpda.Rows.Count > 0)

{

strSplb = "";

for (int i = 0; i < tmpda.Rows.Count; i++)

{

//添加大类别

strSplb = strSplb + "<tr>";

string strTmpLx = "";

}

……

### 4.1.2 \*\*界面

说明：格式同上。

## 4.2 关键技术难点与解决方案

说明：如需要嵌入代码，其格式如下（嵌入代码量不要连续超过一页）：

int shmctl (int shmid,int cmd,struct shmid\_ds \*buf)

{ …

int iResult = 0;

…

while (cmd) {

}

return iResult;

}

注意：不要有超过一页的连续代码，如果超过一页，须在代码中插入2-3行以上的文字描述。

## 4.3 关键算法（可选）

## 4.4 编码规范

说明：采用何种编码规范。举例：

1. 标识符命名，采用哪种形式，是否采用有意义的命名方式，并举例说明。
2. 注释，例如段前注释、段中注释、方法前注释等，举例说明。
3. 标准的书写格式，合理使用空行及缩进，举例说明。
4. 其他需要说明的。

# 第5章　系统测试

说明：这里需要文字描述测试计划，可以包含各阶段使用到的测试策略、测试环境的搭建等。

举例：

票务管理系统的测试本着及早接入的原则，因此早在需求阶段测试人员就对需求的覆盖度、数据模型的正确性都进行了相应的验证测试，在系统开发过程中进行了单元测试，在集成阶段进行了集成测试，在整个系统开发完成后又针对功能性方面进行了系统测试。由于篇幅的限制，下面主要围绕功能测试用例、性能测试和测试总结三个方面进行说明。

## 5.1 功能测试

说明：对核心业务进行功能测试。

举例：

本系统对普通用户、单位用户、系统管理员三个核心模块进行了功能测试，测试设计如下所示：

（1）普通用户登陆模块功能测试

普通用户功能测试包括普通用户注册、登录、职位类别搜索、投递简历、查看新闻资讯、用户留言。测试用例设计如表5.1所示。

表5.1 普通用户功能测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **用例编号** | **测试用例描述** | **操作过程及数据** | **预期结果** | **测试结果** |
| D001 | 正确填写用户注册信息 | 按照系统要求填写用户名、密码、密码提示问题等信息点击“确定” | 系统提示注册成功 | 通过 |
| D002 | 用户名已存在 | 在用户名中填写刚才注册过的用户名，其余选项正常填写 | 系统提示用户名已存在请重新输入 | 通过 |
| D003 | 密码与确认密码不一致 | 输入的密码与确认密码不一致，其余选项正常填写 | 系统提示用户密码与确认密码不一致 | 通过 |
| D004 | 密码长度小于6位 | 输入的密码与确认密码长度小于6位，其余选项正常填写 | 系统提示用户密码长度不能小于6位 | 通过 |
| D005 | 必填项填写不完全 | 用户没有填写完全系统要求的必须信息 | 系统会根据实际情况提示用户哪项没有填写 | 通过 |
| D006 | 重新填写注册信息 | 点击“重置” | 页面回到初始状态 | 通过 |

## 5.2 性能测试（可选）

## 5.3 测试总结

说明：对整个测试过程中出现的用例总数、各版本错误数、错误等级、错误类型等进行统计。并针对出现的现象进行分析和总结。

举例：

系统测试全部bug数为368个，经过开发人员对bug进行了有效的修改，测试人员进行了回归测试，最终保证了系统具备了较好的稳定性和可用性。测试结束后，在测试总结中通过按版本统计、按等级统计、按类型统计多个角度对测试情况进行了客观的统计和分析，具体如图5.1所示。

按版本统计bug的结果，如图5.1所示。



图5.1 按版本统计bug数

按等级统计bug的结果，如图5.2所示。



图5.2 按问题等级统计bug数

按类型统计bug的结果，如图5.3所示。



图5.3 按类型统计bug数

由上面统计图的结果可以看出，……

# 第6章　结论与展望

说明：总结本项目的优缺点并做出改进计划。给出运行与维护、运营方面的计划。

注意：本章是对项目的总结，而不是个人毕设体会，一定不要写成个人体会！

# 参考文献

参考文献总体样本：

[1] 张炯．Unix网络编程实用技术与实例分析[M]，清华大学出版社，2002，33-50．

[2] 徐千祥．Linux C函数库参考手册[M]，中国青年出版社，2002，55-67．

[3] 张青等．Oracle9i中文版基础教程[M]，清华大学出版社，2003，105-130．

[4] 软件设计模式[EB/OL]，www.itisedu.com/软件设计模式.htm，2004．

[5] 许育诚．软件测试与质量管理[D]，海事大学，2004．

[6] 景新梅．软件产业原动力[J]，中国计算机报，2005，45(7)，32-33．

[7] 赵克佳，赵慧．UNIX程序设计教程[M]，清华大学出版社，2001，78-89．

[8] 陈绍英，戴金龙．软件测试案例分析[J]，测试员，2005，32(8)，23-25．

[9] Wendy Boggs，Michael Boggs．UML与Rational Rose 2002从入门到精通[M]，电子工业出版社，2002，90-101．

[10] Sun Microsystems.Inc. JavaTM 2 SDK, Standard Edition Documentation Version 1.4.1 [EB/OL]，Sun Microsystems.Inc，2002．

注意：参考文献**不少于10篇，学术期刊（[J]类期刊）不少于7篇，至少有1篇外文文献，以近三年的参考文献为主**。

# 致 谢

说明：在致谢中，主要表达对导师和其他相关人员的感谢之意以及结合整体的大学学习，对整个毕业设计（论文）进行总体性、概括性总结，表达出设计（论文）的思路、学习收获、对未来进一步学习的设想。对此，仍要实事求是，过分的颂扬反而会带来消极影响。

举例：

在整个毕业设计的各个阶段我都得到了xxx老师、xxx老师和xxx高级工程师的悉心指导和耐心帮助。

（以下略）

大连东软信息学院

毕业设计（论文）原创承诺书

1、本人承诺：所提交的毕业设计（论文）是认真学习理解学校的《毕业设计（论文）工作规范》后，在教师的指导下，独立地完成了任务书中规定的内容，不弄虚作假，不抄袭别人的工作内容。

2、本人在毕业设计（论文）中引用他人的观点和研究成果，均在文中加以注释或以参考文献形式列出，对本文的研究工作做出重要贡献的个人和集体均已在文中注明。

3、在毕业设计（论文）中对侵犯任何方面知识产权的行为，由本人承担相应的法律责任。

4、本人完全了解学校关于保存、使用毕业设计（论文）的规定，即：按照学校要求提交论文和相关材料的印刷本和电子版本；同意学校保留毕业设计（论文）的复印件和电子版本，允许被查阅和借阅；学校可以采用影印、缩印或其他复制手段保存毕业设计（论文），可以公布其中的全部或部分内容。

5、本人完全了解《毕业（设计）论文工作规范》关于“学生毕业设计（论文）出现购买、他人代写、或者抄袭、剽窃等作假情形的，取消其学位申请资格；已经获得学位的，依法撤销其学位。取消学位申请资格或者撤销学位者，从处理决定之日起3年内，学校不再接受学生学位申请”的规定内容。

6、本人完全了解《学生手册》中关于在“毕业设计（论文）等环节中被认定抄袭他人成果者”不授予学士学位，并且“毕业学年因违纪受处分影响学位的学生不授予学士学位，并且无学士学位申请资格”的规定内容。

以上承诺的法律结果、不能正常毕业及其他不可预见的后果由学生本人承担！

学生本人签字：

年 月 日