

毕业设计（论文）原创性声明

本人郑重声明：所提交的毕业设计（论文），是本人在导师指导下，独立进行研究工作所取得的成果。除文中已注明引用的内容外，本毕业设计（论文）不包含任何其他个人或集体已经发表或撰写过的作品成果。对本研究做出过重要贡献的个人和集体，均已在文中以明确方式标明并表示了谢意。

论文作者签名：

日期： 2015 年 6 月 01 日

www.docin.com

摘 要

在互联网高速发展的背景下，学术论文的数量和内容呈现爆炸式增长。研究者对于下载的学术论文的管理往往费时费力，如果能有一个学术论文管理系统协助管理，能够大大提高学术研究的效率。本设计致力于开发一个具有添加，删除，自动归类，多维浏览等功能的学术论文管理系统^[1]。

学术论文是某一学术课题在实验性、理论性或预测性上具有的新的科学研究成果或创新见解和知识的科学记录，或是某种已知原理应用于实际上取得新进展的科学总结，用以提供学术会议上宣读、交流、讨论或学术刊物上发表，或用作其他用途的书面文件。按写作目的，可将学术论文分为交流性论文和考核性论文。

当前许多学校正在逐步完善学术论文管理工作。这需要一个能满足内部管理需求、提高管理效率的应用系统进行内部管理^[2]。学术论文管理系统使管理毕业生论文的工作在该系统的配合下更加高效和便捷。不管是计算机专业人员还是其他用户都能很快上手，操作简便，采用 web 不需要安装，容易普及。

关键字：学术论文；自动归类；多维浏览；论文管理；

www.docin.com

ABSTRACT

In the context of the rapid development of the Internet, the number and content of academic papers presents explosive growth. The management of downloaded papers often takes much time and effort of researchers. An academic paper management system can greatly improve the efficiency of academic research. This design is committed to developing an academic paper management system which has the functions of adding, deleting, automatically classifying, and multi-dimensional browsing.

Academic papers record the new scientific research or creative insights and knowledge from the experimental, theoretical or prediction pointview. or conclude known principles which are applied to progresses in the scientific summary, conference, exchange, discuss, or academic journals. According to the writing purposes, the academic papers can be divided into exchange papers and assessment papers..

At present, many schools are gradually improving the academic papers management which needs a system to meet internal management needs and to improve the management efficiency of internal management system. The management system of the academic papers makes the management of the graduates work more efficient and convenient in the cooperation of the system. The computer professionals and other users can quickly get started, simple operation, using web which do not need to install and is easy to popularize.

Key words: Academic papers; automatic classification; multidimensional browse; paper management;

目 录

第一章 绪 论	1
1. 1 课题的要求，目的和意义	1
1. 1. 1 课题的要求	1
1. 1. 2 课题的目的和意义	1
1. 2 论文的组织结构	1
第二章 学术论文管理系统的工作原理	3
2. 1 学术论文管理系统的介绍	3
2. 2 学术论文管理系统功能	3
2. 2. 1 用户注册	3
2. 2. 2 用户和论文的管理	3
2. 2. 3 上传下载论文	4
2. 2. 4 论文推荐技术	5
2. 2. 5 查询功能	5
2. 3 分类原理	5
2. 4 效果评估	6
2. 5 本章小结	6
第三章 需求分析	7
3. 1 开发的时代背景	7
3. 2 开发环境和编程语言介绍	7
3. 3 Myeclipse 工具的使用原理及 B/S 架构、MVC 设计模式	7
3. 4 功能需求	8
3. 5 运行需求	9
3. 5. 1 界面需求	9
3. 5. 2 故障处理	9
3. 6 本章小结	10
第四章 开发设计与实现及测试	10
4. 1 开发环境的搭建和使用	11
4. 1. 1 Jquery, ajax, json 的使用	11
4. 1. 2 搭建 ssh 框架和使用	13
4. 2 调用 ssh 框架	16
4. 2. 1 主要使用的函数说明介绍	16
4. 3 软件各功能的设计与实现	18
4. 4 智能分类功能的实现	31
4. 5 系统其他功能的实现	33
4. 6 测试过程	34
4. 6. 1 软件的功能	34

4.6.2 测试结果	34
4.7 本章小结	34
结束语.....	35
致 谢.....	36
参考文献:	37



第一章 绪 论

1.1 课题的要求，目的和意义

1.1.1 课题的要求

1. 对论文建模设计论文元数据，保存论文信息。
2. 程序可以自动地，或按照用户提供的参数多维的返回用户所需要的论文。
3. 程序可以自动的解析论文，并根据论文的基本信息将论文保存到对应文档中。
4. 界面美观简洁

1.1.2 课题的目的和意义

在互联网高速发展的背景下，学术论文的数量和内容呈现爆炸式增长。研究者对于下载的学术论文的管理往往费时费力，如果能有一个学术论文管理系统协助管理，能够大大提高学术研究的效率。本设计致力于开发一个具有添加，删除，自动归类，多维浏览等功能的学术论文管理系统^[12]。

当前许多学校正在逐步完善学术论文管理工作，需要一个能满足内部管理需求、提高管理效率的应用系统进行内部管理。然而随着学校的规模不断扩大，学生人数成倍增长，积累了大量的学生信息，急需进行统一管理以防信息流失。学术论文管理系统使管理毕业生论文的工作在该系统的配合下更加高效和便捷^[2]。不管是计算机专业人员还是其他用户都能很快上手，操作简便，采用 web 不需要安装，容易普及。

1.2 论文的组织结构

本文共分为五章，以下几章内容概要如图 1.1：

第一章主要内容为绪论内容，介绍了本次毕业设计课题的要求，目的和意义及论文组织结构如图 1-1 所示。

第二章主要介绍学术论文管理系统的的工作原理，包括了一般性功能的介绍，如用户和论文的管理，上传和下载技术的论文推荐技术，查询分类功能等。

第三章主要是介绍了学术论文管理系统的需求分析，这其中包括了市场需求、功能需求，运行需求等。

第四章主要介绍具体的开发设计和实现，包括环境的搭建，各功能的实现，使用函数的说明，分类功能的方法和具体实现。

结束语主要介绍开发设计完成之后，对开发设计的总结和未来展望，遇到问题怎样解决，以及后续工作。

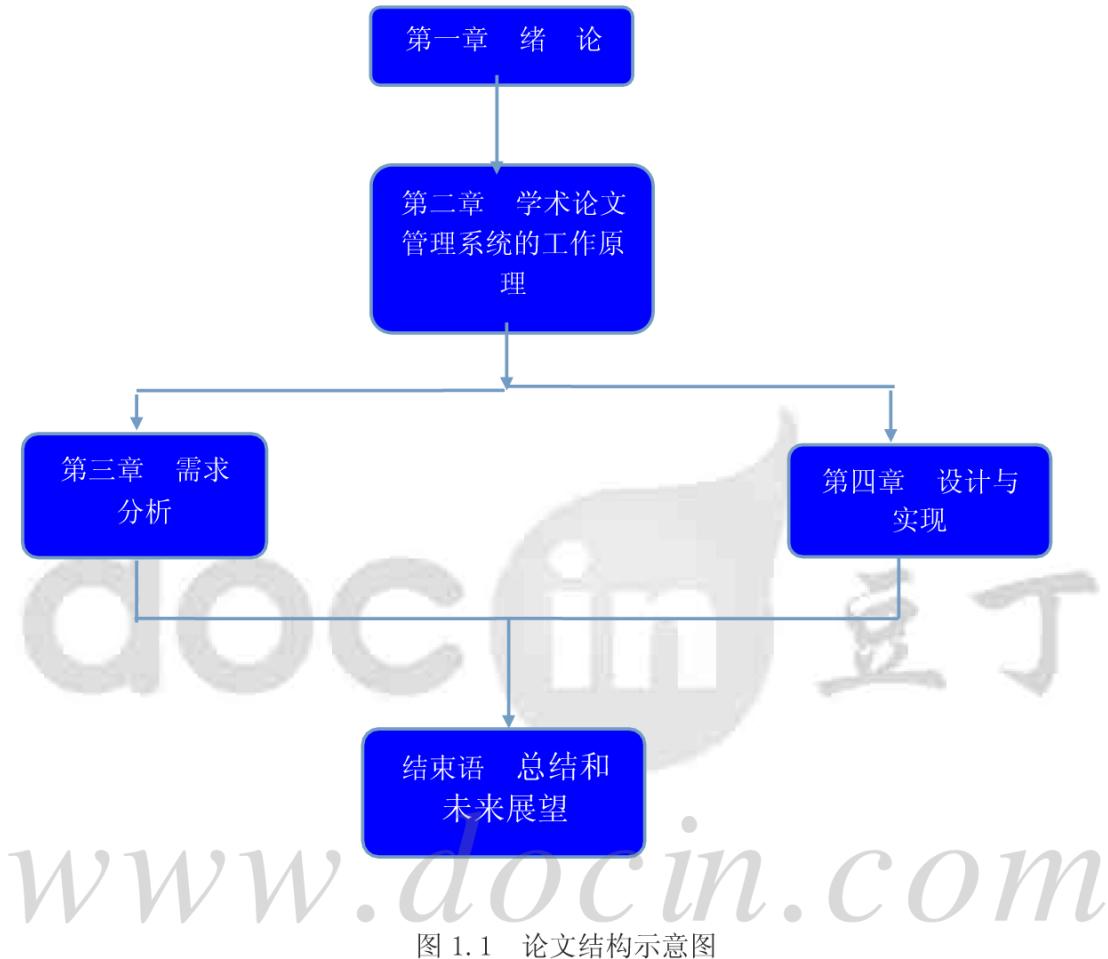


图 1.1 论文结构示意图

第二章 学术论文管理系统的的工作原理

2.1 学术论文管理系统的介绍

学术论文管理系统是通过 web 开发的管理系统软件，使用 B/S 架构，用户只需要在有网络的情况下打开浏览器即可上传、下载和搜索自己想要的论文，和其他客户端程序的系统软件而言显得更方便更快捷。为广大用户提供了更大更全的搜索空间^[9]。

学术论文管理系统面对的广大互联网信息的用户，主要人群是大学生、硕士、教师等，任务就是帮助用户管理他们的论文数据，也可以共享自己的论文造福于社会。针对于学术论文的数量和内容呈现爆炸式增长。研究者对于下载的学术论文的管理往往费时费力，学术论文管理系统因此而诞生了！它能对学术论文协助管理，能够大大提高学术研究的效率，且本设计具有添加，删除，自动归类，多维浏览等功能^[11]。

根据项目的特点，智能上传功能相比简单的存放系统而言能减少用户的上传时间，更方便用户使用该系统软件。

学术论文管理系统主要有三个模块：用户模块、上传下载模块、查询模块。
用户模块：该系统软件是针对于广大用户的，用户被分为管理员用户和普通用户，不同的用户具有不同的权限，在后面将列出。上传下载模块：具有智能上传归类，下载想要的论文数据。查询模块：分为模糊查询模块和精确查询模块。

2.2 学术论文管理系统功能

2.2.1 用户注册

为了方便对论文和用户的管理，用户注册是必不可少的，可进行用户的注册和登录操作。

用户可以给自己定义一个 id，通过这个 id 可以登录系统，使用系统的行为记录将会记录在个人用户中。

2.2.2 用户和论文的管理

有了用户自然少不了对于用户的管理，如果用户忘记了密码怎么办？通过

管理员用户可以找回其他的用户并可以编辑用户。论文是依赖用户产生的，用户通过自己的 Id 登录进系统，上传自己的论文的时候，该论文的所有者即作者将会被指定该 Id，用户和论文的管理因而产生。

管理员后台界面及权限：用户的管理，论文管理,新增用户。

普通用户后台管理界面及权限：自己论文的管理，个人账户的管理

2. 2. 3 上传下载论文

用户使用学术论文管理系统的时候，如果有需要保存和共享自己的学术论文，用户可以上传自己的论文，管理自己的论文，下载他人分享的论文，得到资源的合理分配。

上传自动分类技术：用户上传文件后系统将自动判断用户上传的文件属于哪个类别并进行分类存储。运用 struts2 的上传和下载框架，通过 io 的输入、输出流来写入和读取文件。

IO 中的输入字节流

下面是 IO 中输入字节流的继承图。

- o InputStream
- o ByteArrayInputStream
- o FileInputStream
- o FilterInputStream
- o BufferedInputStream
- o DataInputStream
- o LineNumberInputStream
- o PushbackInputStream
- o ObjectInputStream
- o PipedInputStream
- o SequenceInputStream
- o StringBufferInputStream

InputStream 是所有的输入字节流的父类，它是一个抽象类。

2. ByteArrayInputStream、StringBufferInputStream、FileInputStream 是三种基本的介质流，它们分别将 Byte 数组、StringBuffer、和本地文件中读取数据。PipedInputStream 是从与其它线程共用的管道中读取数据。

3. ObjectInputStream 和所有 FilterInputStream 的子类都是装饰流（装饰器模式的主角）。下表列出了这些流的功能及如何使用它们（具体使用在讲解完装饰器模式后会举几个例子）。

1. IO 中输出字节流的继承图

- OutputStream
- ByteArrayOutputStream
- FileOutputStream
- FilterOutputStream
- BufferedOutputStream
- DataOutputStream
- PrintStream
- ObjectOutputStream
- PipedOutputStream

1. OutputStream 是所有的输出字节流的父类，它是一个抽象类。
2. ByteArrayOutputStream、FileOutputStream 是两种基本的介质流，它们分别向 Byte 数组、和本地文件中写入数据。PipedOutputStream 是向与其它线程共用的管道中写入数据，
3. ObjectOutputStream 和所有 FilterOutputStream 的子类都是装饰流。下表列出了输出字节流的功能及如何使用它们。

2.2.4 论文推荐技术

论文的推荐技术通过数据库中存在的论文进行排序整理，将符合自己规定要求的论文推荐给用户。如：通过下载量、浏览量、和更新时间来划分推荐。

2.2.5 查询功能

查询功能分为模糊查询和精确查询，查询是通过用户的请求发送给数据库，通过数据库的指令来得到相应结果。

模糊查询：通过某一字段，用户通过论文名通过 like 模糊搜索找到对应自己结果相近的最为接近的查询方式。

精确查询：强硬的在规定的类别中进行查询，用户通过准确的字段来一步步缩小查询范围，从而得到自己想要的结果的查询方式。

模糊查询的搜索示例：SELECT 字段 FROM 表 WHERE 某字段 Like 条件

2.3 分类原理

分类原理：数据库中创建类别的表，当用户上传文件时，通过数据库中表的数据和上传的文件进行匹配，自动找到最为合适的分类反馈给用户^[5]。

用户上传文件—> 数据库分析—> 找到匹配选项—> 返回给用户分类结果

数据库中的分类数据来源于学术论文的类别，分类的方法是各种各样的。

如：按研究的学科，可将学术论文分为自然科学论文和社会科学论文。每类又可按各自的门类分下去。如社会科学论文，又可细分为文学、历史、哲学、教育、政治等学科论文。

按研究的内容，可将学术论文分为理论研究论文和应用研究论文。理论研究，重在对各学科的基本概念和基本原理的研究；应用研究，侧重于如何将各学科的知识转化为专业技术和生产技术，直接服务于社会^[13]。

按写作目的，可将学术论文分为交流性论文和考核性论文。交流性论文，目的只在于专业工作者进行学术探讨，发表各家之言，以显示各们学科发展的新态势；考核性论文，目的在于检验学术水平，成为有关专业人员升迁晋级的重要依据^[8]。

2.4 效果评估

分类效果上，数据库将匹配找出对应的文本类别并自动将上传的文本文件进行归类，可以将学术论文更有效的管理。如果用户上传的文件类别在数据库中不存在，用户可以自己定义文件类别并进行保存^[10]。

2.5 本章小结

本章主要介绍了学术论文管理系统的基本工作原理，其中介绍了用户的管理、论文的管理、论文的分类、论文的推荐技术等。学术论文管理系统的目的在于保存用户的学术论文和分享用户的学术论文。该章简单的介绍了论文的分类，论文的查询，上传和下载，论文推荐等简单方法。基本符合管理系统的各组成要素。

第三章 需求分析

3.1 开发的时代背景

在互联网高速发展的背景下，学术论文的数量和内容呈现爆炸式增长。研究者对于下载的学术论文的管理往往费时费力，如果能有一个学术论文管理系统协助管理，能够大大提高学术研究的效率^[3]。本设计致力于开发一个具有添加，删除，自动归类，多维浏览等功能的学术论文管理系统。

3.2 开发环境和编程语言介绍

所在开发环境是 windows 8 64 位操作系统

数据库使用为 MySQL 5.1.6

开发工具 myeclipse

jdk 版本 1.6

服务器选用 tomcat

编程语言为 java html javascript sql

3.3 Myeclipse 工具的使用原理及 B/S 架构、MVC 设计模式

本系统基于技术包含三部分：

1. Myeclipse 工具

Eclipse 是一个开放源代码的、基于 Java 的可扩展开发平台。就其本身而言，它只是一个框架和一组服务，用于通过插件组件构建开发环境。幸运的是，Eclipse 附带了一个标准的插件集，包括 Java 开发工具（Java Development Tools, JDT）^[7]。

虽然大多数用户很乐于将 Eclipse 当作 Java IDE 来使用，但 Eclipse 的目标不仅限于此。Eclipse 还包括插件开发环境（Plug-in Development Environment, PDE），这个组件主要针对希望扩展 Eclipse 的软件开发人员，因为它允许他们构建与 Eclipse 环境无缝集成的工具^[20]。由于 Eclipse 中的每样东西都是插件，对于给 Eclipse 提供插件，以及给用户提供一致和统一的集成开发环境而言，所有工具开发人员都具有同等的发挥场所。

这种平等和一致性并不仅限于 Java 开发工具。尽管 Eclipse 是使用 Java 语言开发的，但它的用途并不限于 Java 语言；例如，支持诸如 C/C++、COBOL 和 Eiffel 等编程语言的插件已经可用，或预计会推出。Eclipse 框架还可用来作为与软件开发无关的其他应用程序类型的基础，比如内容管理系统。

MyEclipse 企业级工作平台 (MyEclipse Enterprise Workbench , 简称 MyEclipse) 是对 Eclipse IDE 的扩展, 利用它我们可以在数据库和 J2EE 的开发、发布, 以及应用程序服务器的整合方面极大的提高工作效率。它是功能丰富的 J2EE 集成开发环境, 包括了完备的编码、调试、测试和发布功能, 完整支持 HTML, Struts, JSF, CSS, Javascript, SQL, Hibernate^[1]。

2. B/S 结构

B/S 是 Browser/Server 的缩写, 客户机上只要安装一个浏览器 (Browser), 如 Netscape Navigator 或 Internet Explorer, 服务器安装 Oracle、Sybase、Informix 或 SQL Server 等数据库。在这种结构下, 用户界面完全通过 WWW 浏览器实现, 一部分事务逻辑在前端实现, 但是主要事务逻辑在服务器端实现。浏览器通过 Web Server 同数据库进行数据交互^[4]。

(1) 维护和升级方式简单。(2) 成本降低, 选择更多。

(3) 应用服务器运行数据负荷较重。

3. MVC 设计模式

MVC 全名是 Model View Controller, 是模型(model)一视图(view)一控制器(controller)的缩写, 一种软件设计典范, 用一种业务逻辑、数据、界面显示分离的方法组织代码, 将业务逻辑聚集到一个部件里面, 在改进和个性化定制界面及用户交互的同时, 不需要重新编写业务逻辑。MVC 被独特的发展起来用于映射传统的输入、处理和输出功能在一个逻辑的图形化用户界面的结构中^[3]。

3.4 功能需求

本系统用例图如下图所示图 3-1 和 3-2。

管理员：上传下载浏览论文，管理论文和一般用户。

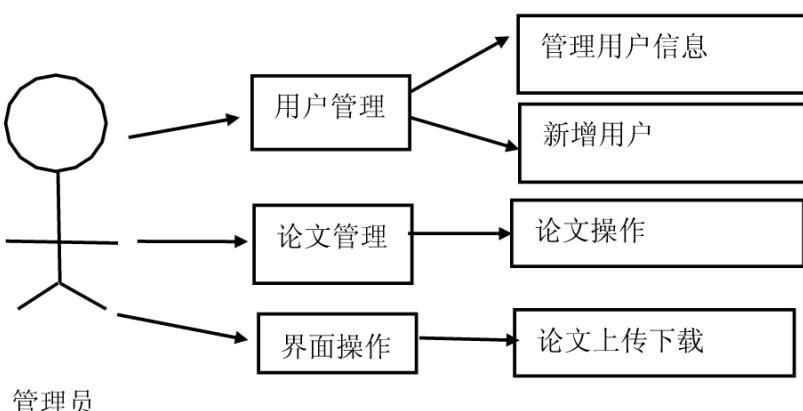


图 3-1 用例图

用户：上传论文管理自己的论文，下载浏览他人论文。管理自己的个人信息，密码操作。

息。

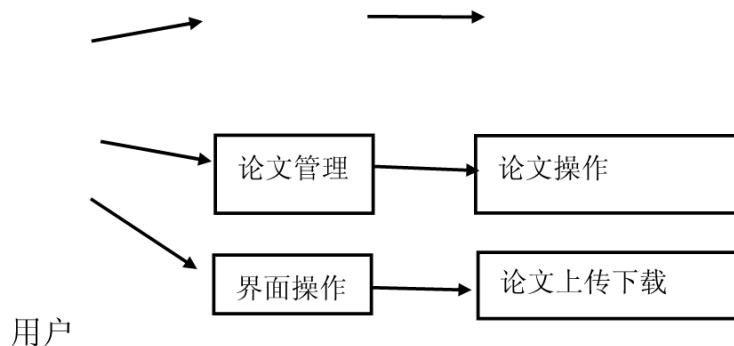


图 3-2 用例图

1. 提供用户注册和管理接口。
2. 用户可以上传和分享自己的学术论文。
3. 提供管理员，方便管理用户和论文。
4. 对论文建模设计论文元数据，保存论文信息。
5. 程序可以自动地，或按照用户提供的参数多维的返回用户所需要的论文。
6. 程序可以自动的解析论文，并根据论文的基本信息将论文保存到对应的文件夹中。

3.5 运行需求

运行平台: pc, 智能手机

运行环境: 在有网络的情况下, 局域网, 互联网等。

运行辅助工具: ie, 谷歌等浏览器

该系统软件为 web 形式的软件, 用户可以在电脑和手机上使用, 只需要打开浏览器访问软件地址即可。

3.5.1 界面需求

用户需要得到一个可视的交互界面, 用户可以注册, 管理自己的账户和论文, 发布论文和下载论文等。平台提供搜索界面, 用户可以在其界面查询到自己想要的结果。各功能都必须要有相对应的功能界面方便用户浏览和使用

3.5.2 故障处理

软件使用过程中如果出现无法刷新等情况, 请检查网络环境的问题。如果非

自身网络问题，联系管理查找服务端问题。如果丢失换忘记密码请联系管理员后修改密码。上传文件可能会出现失败，请留意自己上传的文件是否在规定的范围规格之内，文件类型是否有误。

3.6 本章小结

本章主要介绍软件的需求分析，从功能需求，运行需求，界面需求，故障处理等方面进行全面的分析。功能需求中提出了系统应完成些什么功能，功能之间的联系，为下一步的软件的模块设计打下基础。运行需求中对主要的界面及其出现故障时的处理情况。本章主要为了下一章的系统设计打下基础，简单分析了整个软件的需求情况。



第四章 开发设计与实现及测试

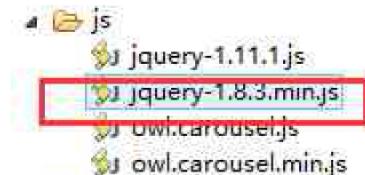
4.1 开发环境的搭建和使用

4.1.1 Jquery, ajax, json 的使用

1.Jquery

在 html<head></head>标签中加上

```
<script type="text/javascript" src="js/jquery-1.8.3.min.js"></script>
```



使用中必须在项目中加入 jquery 的 Js 文件, 如

我们就可以使用 jquery 给我们提供的方法, 通过这些方法加速开发减少开发周期。

Ajax

AJAX 即 “Asynchronous Javascript And XML” (异步 JavaScript 和 XML), 是指一种创建交互式网页应用的网页开发技术。项目中不需要添加其他文件。

Json

项目中必须引入 json 所需要的 jar 包, 这些 jar 包中将提供方法给我们使用, 通过这些方法我们可以从 java 类中得到 json 字符串给前端并返回给用户。

添加 jar 包:



图 4-1 jar 包示意图

(1)三者在项目中的应用, 首先是 struts.xml 中的配置, 如下:

```
<package name="pass" namespace="/" extends="json-default">
```

```

<action name="showtopAction" class="articleAction" method="top">
    <result type="json">
        <!-- 此处将 reslut 的值返回给客户端，root 的值对应要返回
        的值的属性 result
        注意：root 为固定写法，否则不会把 result 的值返回
        给客户端 -->
        <param name="root">result</param>
    </result>
</action>
</package>

```

(2)其次是 java 类中将 List 集合转换为 json 字符串：

```
JSONArray json = JSONArray.fromObject(top);
```

```
System.out.println(json);
```

```
result = json.toString();
```

(3)最后，jsp 页面取到 json 字符串并输出：

```

$(function(){          // —————> 当页面加载的时候调用的方法
    $.ajax({
        url:      "${pageContext.request.contextPath}/newartAction.action",
        // 数据发送方式
        type: "post",           // 接受数据格式
        dataType : "json",
        success : newshow
    });
    function newshow(newww){    //-->ajax 操作成功后调用的方法
        var newjson=eval("(" + newww + ")");
        // alert(json);
        for(var i=0;i<newjson.length;i++){
            $("#imge"+i).attr("src",newjson[i].apic);
            var newspant=document.getElementById ("titl"+i);
            newspant.innerHTML = newjson[i].aname;
            var newrem=document.getElementById ("rema"+i);
            newrem.innerHTML = newjson[i].remark;
        }
    }
}

```

4.1.2 搭建 ssh 框架和使用

Ssh 框架指：struts2, spring, hibernate

使用 ssh 框架体要在项目中引入一些包，myeclipse 中都存在这些 jar 包，我们只需要配置和引用下即可。

配置步骤：

(1) struts2 的配置

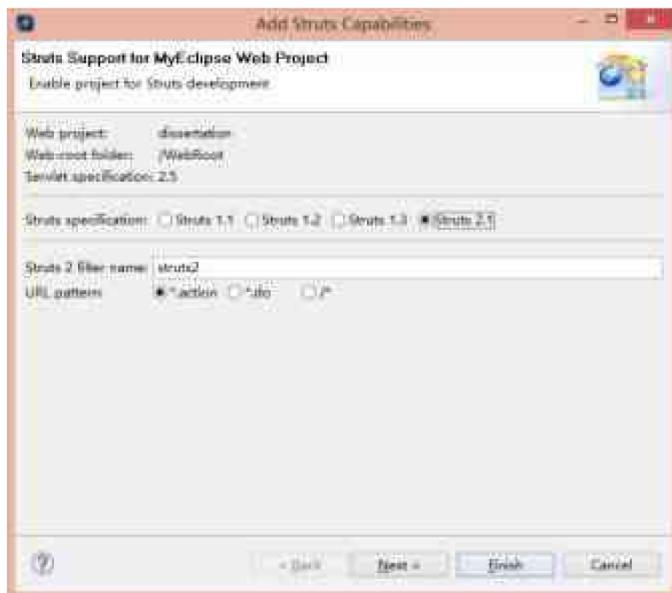


图 4-2 struts2 配置图

Next-选择需要的库文件

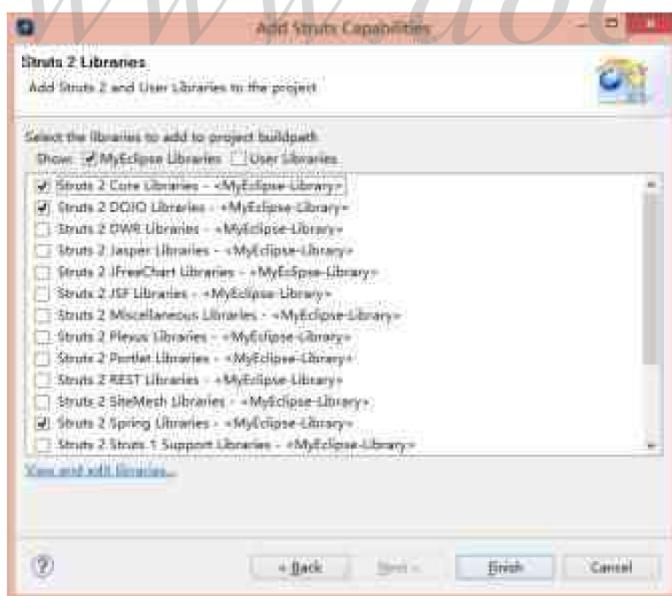


图 4-3 库文件图

(2) 添加 spring 的开发能力

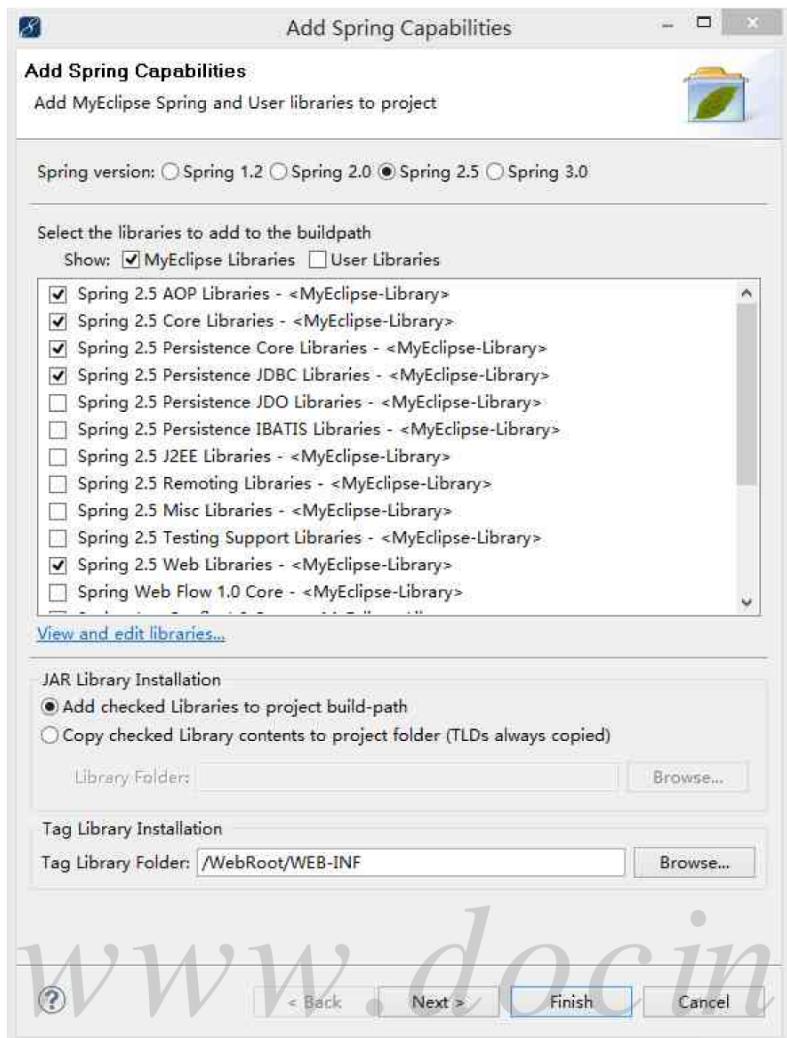


图 4-4spring 开发能力图

(3) 添加 hibernate 的开发能力

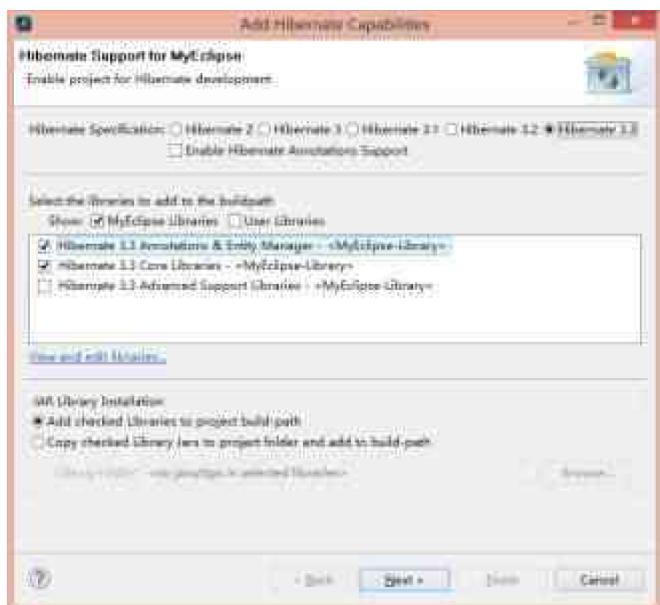


图 4-5 hibernate 开发能力图

(4) 下面简单介绍下 ssh 框架的功能

Struts2：作为 web 层框架应该大体实现以下功能：1. 获取表单内容，并组织生成参数对象 2. 根据请求的参数转发请求给适当的控制器 3. 在控制器中调用业务接口 4. 将业务接口返回的结果包装起来发送给指定的视图，并由视图完成处理结果的展现 5. 做一些简单的校验或是国际化工作

Spring： 事务的处理，控制反转，依赖注入。

Hibernate：主要作用在数据库的操作

1. 数据库连接不需要我们自己来管理，如打开和关闭 2. Hibernate 对原来的 SQL 语句进行了封装，以前写很多 SQL 语句，现在你在保存的时候直接可以用 SAVE (对象) 来实现了。3. 原来是对表操作，现在是对对象操作。

(5) 使用 struts2 框架需要在 web.xml 中进行配置，配置如下

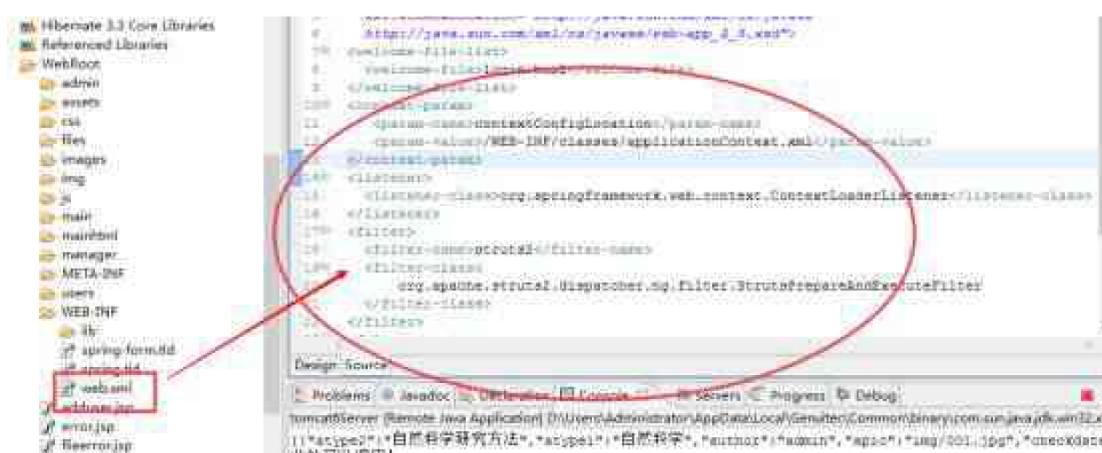


图 4-6 struts2 框架在 web.xml 中配置图

4.2 调用 ssh 框架

4.2.1 主要使用的函数说明介绍

struts2:

<%@taglib prefix="s" uri="/struts-tags" %> 这是 struts2 提供的 jsp 标签，它比 Html 标签更方便。

<form action="qresult.action" method="post" enctype="multipart/form-data">
form 表单中 action 指向 struts.xml 中匹配的 action 来调用 action 类，并执行自己需要执行的方法，如下执行 ArticleAction.java 中的 qresult 方法

```
<action name="qresult" class="articleAction" method="qresult">
    <result name="success">/style-demo.jsp</result>
    <result name="input">/error.jsp</result>
</action>
```

执行完方法后会有返回值，根据返回值达到跳转页面的效果。

Spring:

用来管理 javabean 和事务，代码展现如下：

```
<!--
    用户
-->
<bean id="loginDao" class="com.ap.dao.impl.LoginDaoImpl">
    <property name="sessionFactory" ref="sessionFactory"></property>
</bean>
<bean id="loginBiz" class="com.ap.biz.impl.LoginBizImpl">
    <property name="logindao" ref="loginDao"></property>
</bean>
<bean id="loginAction" class="com.ap.action.LoginAction">
    <property name="lb" ref="loginBizProxy"></property>
</bean>
```

这是用户 javabean 的集合。

事务处理如下：

```
<bean id="transactionManager"
    class="org.springframework.orm.hibernate3.HibernateTransactionManager">
    <property name="sessionFactory"
        ref="sessionFactory"></property>
</bean>
```

<!-- 用户代理 -->

```

<bean id="loginBizProxy"
      class="org.springframework.transaction.interceptor.TransactionProxyFactoryBean">
    <property name="proxyInterfaces">
      <value>com.ap.biz.interfaces.ILoginBiz</value>
    </property>
    <property name="target" ref="loginBiz"></property>
    <property name="transactionAttributes">
      <props>
        <prop key="login*>">PROPAGATION_REQUIRED</prop>
        <prop key="*>">PROPAGATION_REQUIRED</prop>
      </props>
    </property>
</bean>

```

增加代理，控制用户可使用的方法。

Hibernate:

Hibernate 可以将数据库中的表作为对象来操作，首先需要建立自己的需要的数据库和需要的表，先将数据库进行配置，配置正确后可以将表导出成 java 类和一个 xml 文件。

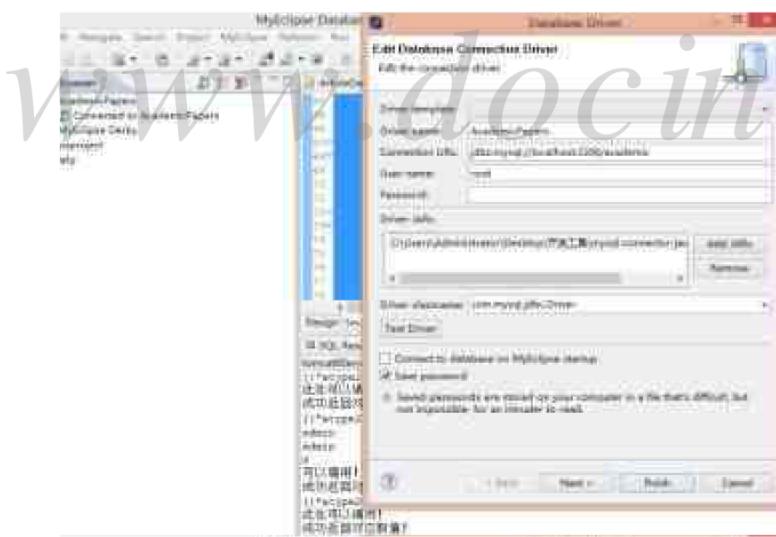


图 4-7 建立数据库

hibernate 中封装了很多对数据库操作的方法，在一定程度上减少了开发者的开发时间。使用之前需要配置连接属性，看是否能连接上数据库，该项目为 ssh 开发，我将 hibernate 和 spring 的配置文件合并在了 applicationContext.xml 文件中，配置如下：

```

<bean id="dataSource"
      class="org.apache.commons.dbcp.BasicDataSource">
    <property name="driverClassName"
              value="com.mysql.jdbc.Driver">
    </property>
    <property name="url"
              value="jdbc:mysql://localhost:3306/academic?useUnicode=true&characterEncoding=UTF-8">
    </property>
    <property name="username" value="root"></property>
</bean>

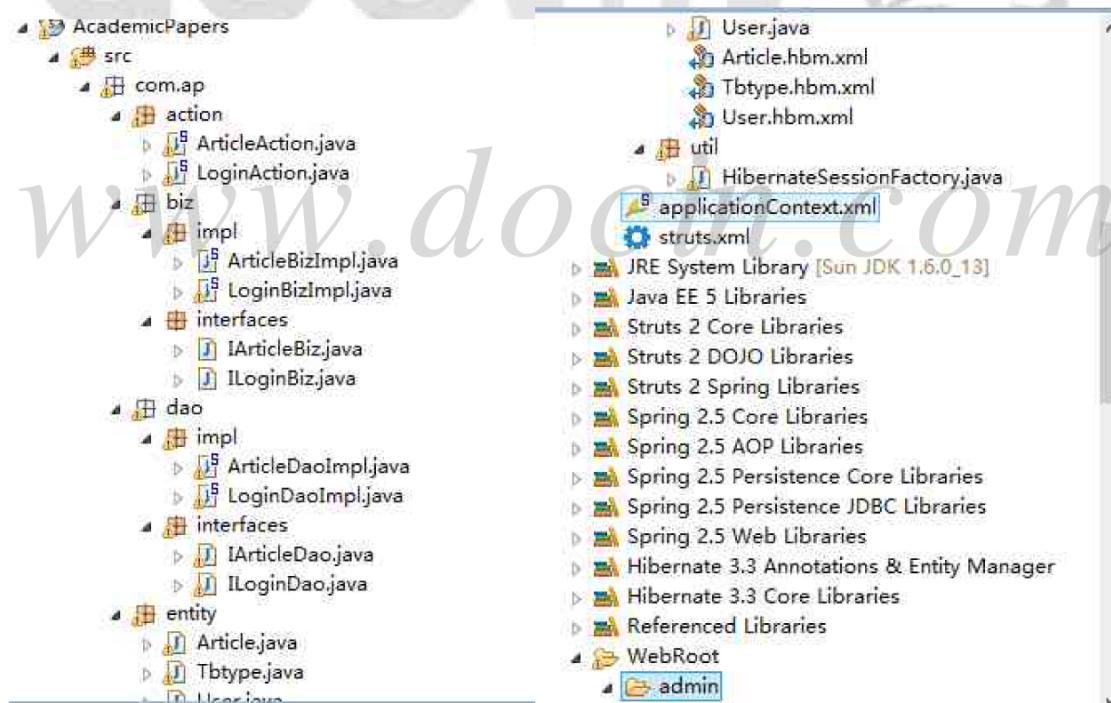
```

java 类中调用的框架代码如下：

this.getHibernateTemplate().find()通过该方法可返回自己想要的结果。

4.3 软件各功能的设计与实现

代码结构如下图：



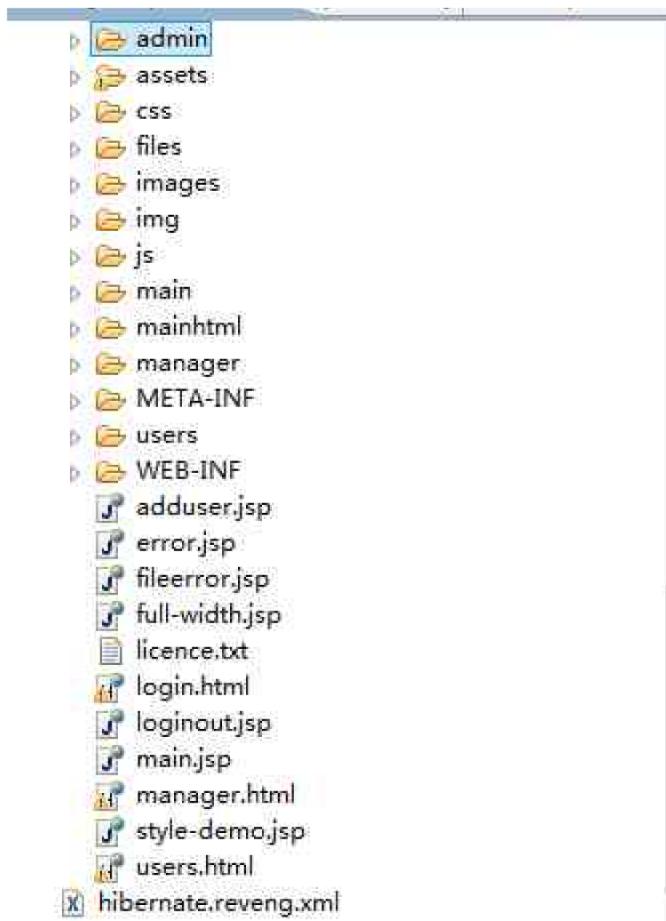


图 4-8 代码结构图

开发环境搭建完成，介绍下数据库中需要做的
新建数据库——> 新建表
数据库名： academic
数据库中的表： article (论文数据表), tbtype (分类表), user (用户数据表)

3 种表创建如下：

```
-- -----
-- Table structure for `article`


DROP TABLE IF EXISTS `article`;

CREATE TABLE `article` (
    `ano` bigint(20) NOT NULL,
    `author` varchar(20) NOT NULL,
    `aname` varchar(20) DEFAULT NULL,
    `atype1` varchar(20) DEFAULT NULL,
    `atype2` varchar(20) DEFAULT NULL,
```

```
'atype3` varchar(20) DEFAULT NULL,  
`apic` varchar(20) DEFAULT NULL,  
`remark` varchar(20) DEFAULT NULL,  
`uploaddate` date DEFAULT NULL,  
`checkdate` date DEFAULT NULL,  
PRIMARY KEY(`ano`)  
) ENGINE=MyISAM AUTO_INCREMENT=7 DEFAULT  
CHARSET=utf8;Tbtype:  
-----  
-- Table structure for `tbtype`  
-----  
DROP TABLE IF EXISTS `tbtype`;  
CREATE TABLE `tbtype` (  
`tno` int(11) NOT NULL,  
`bigtype` varchar(20) NOT NULL,  
`setype` varchar(20) NOT NULL,  
`thtype` varchar(20) NOT NULL,  
`detail` varchar(20) DEFAULT NULL,  
PRIMARY KEY(`tno`)  
}  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;  
-----  
-- Table structure for `user`  
-----  
DROP TABLE IF EXISTS `user`;  
CREATE TABLE `user` (  
`username` varchar(20) NOT NULL,  
`password` varchar(20) DEFAULT NULL,  
`type` varchar(2) DEFAULT NULL,  
`point` bigint(20) DEFAULT NULL,  
PRIMARY KEY(`username`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

1. 提供用户注册，登录和管理接口。

注册页面：前端展示如下：

用户名：

密 码：

确认密码：

图 4-9 注册界面图

用户提供用户名，密码即可完成创建，实现方法

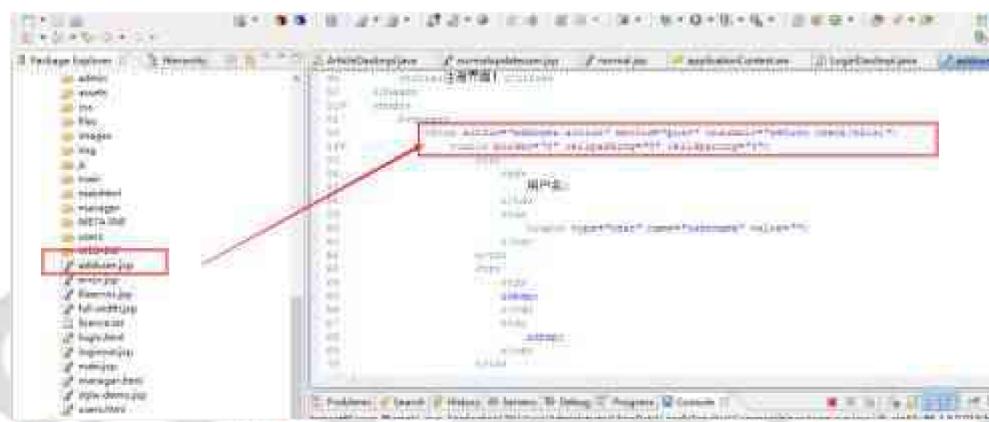


图 4-10 注册界面实现代码图

新建 adduser.jsp 调用 adduser.action，根据 struts2.xml 找到匹配的 Action.java，调用其中的 adduser() 的方法，方法如下

```
public String adduser() throws Exception {
    user.setType("L");
    user.setPoint((long)0);
    boolean flag = lb.adduser(user);
    ActionContext act = ActionContext.getContext();
    Map<String, String> request = (Map<String, String>) act.get("request");
    if(flag == true) {
        request.put("message", "注册用户成功");
        return SUCCESS;
    } else {
        request.put("message", "注册失败，用户名已存在！");
        return INPUT;
    }
}
```

```
}
```

注：lb 是 private ILoginBiz lb; 的对象，最终，调用实现类中的 adduser() 的方法实现在数据库中添加用户的功能。

```
/**  
 * 新增用户  
 */  
  
public boolean adduser(User user) {  
    // TODO Auto-generated method stub  
    User userd=(User)this.getHibernateTemplate().get(User.class,  
user.getUsername());  
    if(userd!=null){  
        return false;  
    }  
  
    else {  
        this.getHibernateTemplate().save(user);  
        return true;  
    }  
}
```

注：返回 false 说明已经有该用户无法继续添加该用户。

登录界面如下：



图 4-11 登录界面图

用户通过用户名和密码登陆系统

DAO 层实现方法如下：

```
public User login(String username, String password) {
    User user=(User)this.getHibernateTemplate().get(User.class,
username);

    if(user!=null){
        if(user.getPassword().equals(password)){
            return user;
        }
    } else {
        return null;
    }
}
return null;
}
```

注：如果用户名 User 类为空，表示搜索不到该用户即没有该用户，只有当用户名和密码完全匹配的情况下才能登陆成功。

用户管理和论文管理页面的展示和 DAO 层的实现。



图 4-12 管理员界面

管理员和普通用户是通过表中的字段来判断的：

username	password	type	point
admin	admin	S	管理员 0
sao	123	L	1000
ww	123	L	0
aa	123	L	0
bb	123	L	0
vc	123	L	0
qa	123	L	0

图 4-12 管理员与普通用户判断图

管理层 Dao 的实现代码：LoginDaoImpl.java 中（用户）

```
/**用户查询 */
public List<User> queryuser(String name) {
    // TODO Auto-generated method stub
    List<User> list = null;
    if(name==null||"".equals(name)){
        list= (List<User>) this.getHibernateTemplate().find("from User");
    }
    else {
        list= (List<User>) this.getHibernateTemplate().find("from User
where username like '%"+name+"%'");
    }
    return list;
}

/**
 * 删除用户
 */
public boolean deleteUser(String username){
    User
    user=(User)getHibernateTemplate().get(User.class,username);
    this.getHibernateTemplate().delete(user);
    return true;
}
/**
```

```
* 根据主键取到对应用户
*/
public User getuserByName(String username) {
    // TODO Auto-generated method stub
    User user=(User)this.getHibernateTemplate().get(User.class,username);
    return user;
}
/***
 * 更新用户
*/
public boolean updateuser(User user) {
    // TODO Auto-generated method stub
    // TODO Auto-generated method stub
    this.getHibernateTemplate().update(user);
    return true;
}

ArticleDaoImpl.java (论文)
/**
 * 删除论文
*/
public boolean deleteArt(Long ano){
    Article art=(Article)getHibernateTemplate().get(Article.class,ano);
    this.getHibernateTemplate().delete(art);
    return true;
}
//管理员管理论文信息
public List<Article> qresultart(Long ano) {
    // TODO Auto-generated method stub
    List<Article> qreart=null;
    if(ano==0){
        qreart= (List<Article>) this.getHibernateTemplate().find("from Article");
    }
    else
    {
        qreart= (List<Article>) this.getHibernateTemplate().find("from Article

```

```

where ano like "%" +ano+"%\"");
}
System.out.println("查询论文成功！");
return qreart;
}

public List<Article> qresultart(Long ano) {
    // TODO Auto-generated method stub
    List<Article> qreart=null;
    if(ano==0){
        qreart= (List<Article>) this.getHibernateTemplate().find("from
Article");
    }
    else
    {
        qreart= (List<Article>) this.getHibernateTemplate().find("from Article
where ano like "%" +ano+"%\"");
    }
    System.out.println("查询论文成功！");
    return qreart;
}

//用户管理自己的论文
public List<Article> norquerart(String author) {
    // TODO Auto-generated method stub
    List<Article> qreart= (List<Article>)
this.getHibernateTemplate().find("from Article where author ='" +author+"'");
    return qreart;
}

```

2. 用户可以上传和分享自己的学术论文。

用户的上传和下载是 struts 提供的方法，界面如下：



图 4-13 用户上传与下载图

调用的方法在 ArticleAction.java 中：

```

// 文件上传
public String updateFile() throws Exception {
    // 基于 myFile 创建一个文件输入流
    InputStream is = new FileInputStream(myFile);

    // 设置上传文件目录
    String uploadPath = ServletActionContext.getServletContext()
        .getRealPath("/files/upload");
    // 设置目标文件
    File toFile = new File(uploadPath, this.getMyFileName());

    // 创建一个输出流
    OutputStream os = new FileOutputStream(toFile);

    // 设置缓存
    byte[] buffer = new byte[1024];

    int length = 0;

    // 读取 myFile 文件输出到 toFile 文件中
    while ((length = is.read(buffer)) > 0) {
        os.write(buffer, 0, length);
    }
}

```

```

}

System.out.println("上传用户" + username);
System.out.println("上传文件名" + myFileFileName);
// System.out.println("上传文件类型" + myFileContentType);

art.setAname(myFileFileName);
art.setAtype1(art.getAtype1());
art.setAtype2(art.getAtype2());
art.setAtype3(art.getAtype3());
//art.setAtype1(myFileContentType);
art.setAuthor(username1);
String upres=iab.upfile(art);

// 关闭输入流
is.close();

// 关闭输出流
os.close();

return SUCCESS;
}
//返回一个输入流，作为一个客户端来说是一个输入流，但对于服务器端
是一个 输出流

```

下载界面如下图，查询论文后可下载对应自己想要的论文：

Table(s)



论文名	大类	类别	简介	作者	下载
城市发展论.pdf	社会科学	经济学	该论文...	admin	下载
人口报告.pdf	社会科学	社会学	人口问题...	nb	下载
金融衍生品.pdf	社会科学	计算机	计算机...	lily	下载
基础金融.pdf	社会科学	哲学	哲学...	aa	下载
基础金融.pdf	社会科学	计算机	计算机应用...	mm	下载
自然科学研究方法.pdf	自然科学	自然科学研究方法	自然科学研究方法...	admin	下载

图 4-14 下载界面

实现的主要方法：

```

//返回一个输入流，作为一个客户端来说是一个输入流，但对于服务器端是
一个 输出流
public InputStream getDownload()throws Exception{

    Article artfile=getuserByAno();
    myFileFileName=artfile.getAname();
    //得到要下载的文件的完整路径
    String
path=ServletActionContext.getServletContext().getRealPath("/")+"files/upload/"+artfile
.getAname();
    //必须得到 HttpServletReponse 对象
    HttpServletResponse response=ServletActionContext.getResponse();
    //设置响应的内容类型
    response.setContentType("application/x-msdownload"); //指定响应动作
是下载
    //将要下载的文件转换成流
    FileInputStream stream=new FileInputStream(new File(path));
    //弹出下载框时，显示什么
    //response.setHeader("Content-Disposition",
"attachment;filename="+name);
    response.setHeader("Content-Disposition", "attachment;filename="+new
String(artfile.getAname().getBytes
("gbk"),"iso-8859-1"));
    return stream;

}

```

3.对论文建模设计论文元数据，保存论文信息。

论文在保存的过程中，上传后有些信息是要记录到数据库中的，并且文件是成功上床到对应的文件夹中。上传的信息将记录在 `article` 表中，文件将保存在 `webapp/files/upload` 下

信息保存 DAO 层的方法如下：

```

public String upfile(Article art){
    Timestamp ts = new Timestamp(System.currentTimeMillis());
    art.setAno(getMaxId()+(long)1);
    art.setUploaddate(ts);
    this.getHibernateTemplate().save(art);
}

```

```

        return "SUCCESS";
    }

    public int getMaxId(){
        Session session
            = this.getHibernateTemplate().getSessionFactory().openSession();
        try{
            Query query = session.createQuery("select max(ano) from
Article");
            Object obj = query.uniqueResult();
            if(obj!=null){
                return Integer.parseInt(obj.toString());
            }
        }catch(Exception e){
            e.printStackTrace();
        }finally{
            session.close();
        }
        return 0;
    }
}

```

4. 程序可以自动地，或按照用户提供的参数多维的返回用户所需要的论文。
系统有两种查询方式，模糊查询和精确查询。如下图显示：



图 4-15 模糊查询与精确查询图

模糊查询：用户输入要查询的论文名称，搜索后，后台通过这个值查询类似的结果并返回给用户

DAO 层实现方法：

```
public List<Article> qres(String name) {  
    // TODO Auto-generated method stub  
    List<Article> qre= (List<Article>) this.getHibernateTemplate().find("from Article  
where aName like '%"+name+"%'");  
    System.out.println("查询成功！ ");  
    return qre;  
}
```

注： name 是用户输入到文本框的值。

精确查询 DAO 层实现方法：

```
public List<Article> surequery(String atype1, String atype2, String atype3) {  
    List<Article> qreart= (List<Article>) this.getHibernateTemplate().find("from Article  
where atype1 like '%"+atype1+"%' and atype2 like '%"+atype2+"%' and  
atype3 like '%"+atype3+"%'");  
    return qreart;  
}
```

注： 用户通过三个类别精确找到文件所属的类别，并锁定在这个范围内符合的结果。

4.4 智能分类功能的实现

智能上传，用户选择好自己要上传的文件后效果如下。



图 4-16 智能上传界面

系统将会自动填充类别，如果没有则可自己编辑。

jsp 中的实现方法：

```
<input type="file" id="file" name="myFile" onchange="gettype()">
当文件控件的值发生变化时将调用 js 方法 gettype(),gettype 定义如下:
<script type="text/javascript">
var fileName="";
function gettype(){
var file = $("#file").val();
fileName = getFileName(file);
alert(fileName);
getaction();
//document.getElementById("bigtype").innerText=${requestScope.mytype.b};

}
function getFileName(o){
    var pos=o.lastIndexOf("\\");
    return o.substring(pos+1);
}
function getaction(){
//var url="gettype.action?typename="+fileName;
// window.location.href=url;
```

```

$.ajax({
    url:
    "${pageContext.request.contextPath}/gettype.action?typename="+fileName,
    // 数据发送方式
    type: "post",           // 接受数据格式
    dataType : "json",
    success : getptype
});
}

function getptype(ptype){
    var typejson=eval("(+ptype+)");
    //alert(typejson[0].bigtype);
    document.getElementById("bigtype").value= typejson[0].bigtype;
    document.getElementById("setype").value= typejson[0].setype;
    document.getElementById("thtype").value= typejson[0].thtype;
}

```

解释：通过 file 的名称，当它的值发生变化时通过 ajax 调用 gettype.action 并把 name 值传递给后台，后台通过返回 json 字符串的方式告诉前段，前段调用 getptype()方法将得到 json 值写入 text 文本框。

4.5 系统其他功能的实现

论文推荐功能，下图为推荐功能的前端展示页面：



图 4-17 论文推荐界面

主页上将展示最近服务器上的资源，并将资源推荐给用户，实现是用 json、html

和 javascript，数据库方面只用到查询。前段 json 的使用在 4.1.1 节进行了讲解。

4.6 测试过程

4.6.1 软件的功能

测试流程图：

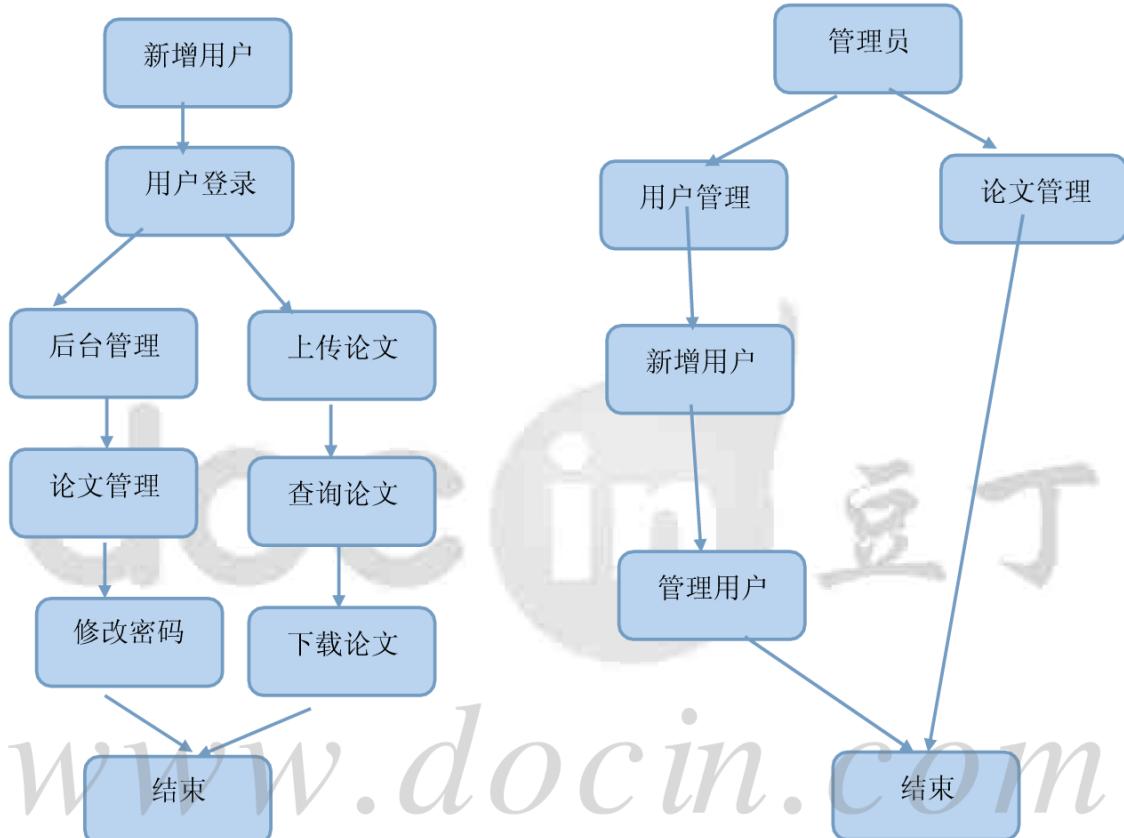


图 4-19 测试流程图

4.6.2 测试结果

1. 管理员用户能修改普通用户信息，普通用户只能修改自己的个人信息。管理员新增的用户可以正常登录。
2. 普通用户只能管理自己，只能下载别人的论文。
3. 上传文件后可以在搜索界面搜索到。

4.7 本章小结

本章通过实验设计和结果分析说明了软件具体实现方法和效果，阐述了每个功能的核心实现方法和作用。

结束语

总结：

本论文于 2014 年底确定论文方向之后，我广泛收集了有关学术论文管理系统方面的相关文献进行阅读，其中包括 jsp,SSH 三大框架等。目前阶段，中大型企业主流使用 eclipse 和 jsp 开发金融和互联网项目，相关的文献和教学相对于其他软件领域还是挺多的，这也是为什么我选择这一门技术开发的原因。在研究的起始阶段，笔者仔细学习和阅读了相关文献，并尝试写了几个 demo 来熟悉这些技术。

学术论文管理系统是用来管理学生和教师的学术论文的，为他们提供了一个交流共享的平台。该系统有助于论文价值的体现，让论文能够更广泛的传播开来，让更多的人来欣赏和评价学术论文，提高论文的质量。本论文系统，使用用户的形式来进行管理，多个用户可以同时访问该系统，用户之间是相互独立的。该系统并没有太多复杂的算法思想，但是它能够满足广大用户的需求。使用 java web 开发省去了用户安装客户端的麻烦。

本论文经过反复测试，符合该软件的需求。说明了该系统在学术论文进行管理方面有显著效果，对学术论文的发展有推动作用。

未来展望：

应该说，学术论文管理系统的实现，就目前来说，无论是理论上还是在设计上，还只是初具雏形，并没有能够达到推广到市场上的优势和机会，因为还是存在不成熟和故障的地方，远没达到完善的地步。而且，国内在这方面的研究，相对于国际水平来说，还很滞后。对目前文本推荐方面的研究，笔者认为还存在以下问题和不足：

(1) 数据获取方面，主要还是依赖用户的上传。没有足够的用户将导致数据的不完善，系统资源的匮乏。

(2) 在智能分类的技术上，因为相关数据较少，无法做到完美的分类，分类复杂将导致页面太过拥挤给人的感觉上烦乱而且不易操作。

笔者认为，未来学术论文管理系统有以下热点与方向：

①对当前的学术论文管理系统再进行改进，使得论文系统有更准确更符合读者兴趣的使用。

②吸引用户的使用，增加用户量让更多的人可以分享自己的论文。

致 谢

本论文在胥备导师的悉心指导下完成的。导师渊博的专业知识、严谨的治学态度，精益求精的工作作风，诲人不倦的高尚师德，严于律己、宽以待人的崇高风范，朴实无法、平易近人的人格魅力对本人影响深远。不仅使本人树立了远大的学习目标、掌握了基本的研究方法，还使本人明白了许多为人处事的道理。本次论文从选题到完成，每一步都是在导师的悉心指导下完成的，倾注了导师大量的心血。在此，谨向导师表示崇高的敬意和衷心的感谢！在写论文的过程中，遇到了很多的问题，在老师的耐心指导下，问题都得以解决。所以在此，再次对老师道一声：老师，谢谢您！

时光匆匆如流水，转眼便是大学毕业时节，春梦秋云，聚散真容易。离校日期已日趋渐进，毕业论文的完成也随之进入了尾声。从开始进入课题到论文的顺利完成，一直都离不开老师、同学、朋友给我热情的帮助，在这里请接受我诚挚的谢意！在此我向南京邮电大学通达学院专业的所有老师表示衷心的感谢。大学的四年时光里，我经历了太多太多事情，也学到了很多必备的知识，为我走出校园踏上社会很大的帮助。

最后要感谢的是我的父母，他们不仅培养了我对中国传统文化的浓厚的兴趣，让我在漫长的人生旅途中使心灵有了虔敬的归依，而且也为我能够顺利的完成毕业论文提供了巨大的支持与帮助。在未来的日子里，我会更加努力的学习和工作，不辜负父母对我的殷殷期望！我一定会好好孝敬他们，报答他们！爸妈，我爱你们！

“长风破浪会有时，直挂云帆济沧海。”这是我少年时最喜欢的诗句。就用这话作为这篇论文的一个结尾，也是一段生活的结束。希望自己能够继续少年时的梦想，永不放弃。

参考文献：

- [1] 冯燕奎, 赵德奎. JSP 实用案例教程[M] 清华大学出版社, 2004, 5: 70-100
- [2] 王家华 软件工程[M] 东北大学出版社 2001 年 3 月 303 页
- [3] 王宜贵 软件工程[M] 北京: 机械工业出版社, 2002: 20-79
- [4] 孙卫琴 精通 struts[M]电子工业出版社 2004 年 8 月 50-70
- [5] 孙卫琴 精通 hibernate[M]电子工业出版社 2005 年 4 月 100-120
- [6] 张洪斌 java2 高级程序设计[M]中科多媒体出版社 2001 年 11 月 85-90
- [7] Jim Arlow UML2.0 and the Unified Process[M]机械工业出版社 2006 年 6 月 30-40
- [8] 林信良 spring2.0 技术手册[M]电子工业出版社 2006 年 6 月 50-100
- [9] 熊节、林仪明、张凯峰、陈玉泉等主编[《CSDN 社区电子杂志——Java 杂志》创刊号] [10] 《程序员》杂志 2007 年第 4 期
- [11] 耿祥义编著. JSP 基础编程 [M]. 清华大学出版社, 2004. 55-162
- [12] 徐建波, 周新莲. Web 设计原理与编程技术[M]. 中南大学出版社, 2005. 185-193
- [13] 孙鑫编著. Java Web 开发详解 [M]. 电子工业出版社, 2006. 189-274
- [14] 林上杰, 林康司编著. JSP2.0 技术手册 [M]. 电子工业出版社, 2004. 3-6
- [15] 萨师煊, 王珊.数据库系统概论(第三版)[M].北京: 高等教育出版社, 1998.
- [16] 蔡剑, 景楠. Java Web 应用开发: J2EE 和 Tomcat[M].北京: 清华大学出版社, 2004.
- [17] Zoya Ali. Designing Object Oriented Software Applications within the Context of Software Frameworks[D]. Ohio State University, 2011.
- [18] Rachit Mohan Garg, Yamini Sood, Balaji Kottana, Pallavi Totlani. A Framework Based Approach for the Development of Web Based Applications Waknaghat[J]. Jaypee University of Information Technology, 2011, 1(1):1-4
- [19] Alice Woudhuysen. China internet: The long march toward e-commerce [J]. the economist intelligence unit. 2007.
- [20] Sanden, Bo. "SYSTEMS PROGRAMMING WITH JSP:EXAMPLE-A VDU CONTROLLER", Communications of the ACM, Vol28, No10, 1985.
- [21] Burgess, R. S. "DESIGNING CODASYL DATABASE PROGRAMS USING JSP", Information and Software Technology, Vol29, No3, 1987.
- [22] Javey, S. "CONCEPT OF 'CORRESPONDENCE' IN JSP", Proceedings of the Hawaii International Conference on System Science, Vol2, 1987.
- [23] Markus Aleksy, Axel Korthaus, Martin Schader. "Use Java and the CORBA realization distribute type system" , Journal of Pingxiang College, No.4, 2005.
- [24] Markus Aleksy, Axel Korthaus, "Martin Schader. Use Java and the CORBA

- realization distribute type system”, Journal of Pingxiang College, No.4, 2005.
- [25] W. Clay, Richardson, Donald, ” Avondolio. The Java high class weaves a distance: JDK 5” , Scientific & Technology Book Review, No.3, 2006.

