

南京理工大学泰州科技学院

毕业设计说明书(论文)

作者: 李薇 学号: 1909030105

学院(系): 计算机科学与工程学院

专业: 计算机科学与技术

题目: 大型数据库课程网站

的设计与实现

指导者: 丁勇 副教授

评阅者: 姜枫 教授

2023 年 05 月

毕业设计说明书（论文）中文摘要

随着互联网时代的到来，传统的课程方式逐渐与网络相结合，目的是为了实
现优质教学资源的共享，要求课程的相关内容通过网站的方式向外开放，打破教
学教育受制于教学资源不足的瓶颈，把教师在日常教学过程中所总结、积累的教
学内容、教学方法、教材等一系列教学资料通过网络平台展示出来，使学生更好
地学习从而进一步提高学生们的知识储备。本课题设计并实现了课程网站系统，
采用 B/S 架构设计，共分为客户端、应用服务器端及数据库端三个部分，利用了
Java 语言、Eclipse 开发平台、MySQL 数据库、Tomcat Web 服务器。功能主要包
括：用户注册与登录、课程章节管理、课程在线学习、课程资源管理、课程资源
查询、课程资源下载等。学生可以查看自己的课程信息、作业和考试成绩，教师
可以管理自己的课程、学生和试题，管理员可以管理整个网站的用户和数据。该
系统具有很高的实用性和安全性，提高了学生和教师的工作效率，方便管理员对
整个网站进行管理，因此可以在教学中得到应用。

关键字 课程网站；Java；B/S；MySQL；Eclipse

毕业设计说明书（论文）外文摘要

Title Design and Implementation of
the Large Database Course Website

Abstract

With the advent of the Internet era, traditional teaching methods are gradually being combined with the Internet to achieve the sharing of high-quality teaching resources. The aim is to break the bottleneck of teaching and education being limited by insufficient teaching resources. Teachers can share a series of teaching materials, teaching methods, and textbooks accumulated in their daily teaching process through a network platform, which can be displayed to students. This enables students to learn better and further improve their knowledge reserves. This project designs and implements a course website system, which is designed using the B/S architecture and is divided into three parts: the client, application server, and database server. It uses Java language, Eclipse development platform, MySQL database, and Tomcat Web server. The functions mainly include user registration and login, course chapter management, online course learning, course resource management, course resource query, and course resource download. Students can view their course information, assignments, and exam scores, teachers can manage their courses, students, and test questions, and administrators can manage the users and data of the entire website. This system has high practicality and security, improves the work efficiency of students and teachers, and facilitates the administrator to manage the entire website. Therefore, it can be promoted and applied in teaching.

Keyword Course Website; Java; B/S; MySQL; Eclipse

目 录

1	引言	1
1.1	研究背景	1
1.2	研究现状	1
1.3	本文组织结构	2
2	相关技术	3
2.1	MySQL	3
2.2	Eclipse	3
2.3	B/S	3
2.4	JSP	3
2.5	Servlet	4
3	系统需求分析	5
3.1	可行性分析	5
3.2	功能需求分析	5
4	系统设计	8
4.1	功能模块设计	8
4.2	系统流程图设计	10
4.3	总类图设计	10
4.4	数据库设计	11
5	系统实现	18
5.1	公共模块	18
5.2	管理员模块	19
5.3	教师模块	25
5.4	学生模块	28
6	系统测试	32
6.1	测试方案	32
6.2	测试用例	32
	结束语	34
	致谢	35
	参考文献	36

1 引言

1.1 研究背景

随着科技、经济的快速发展,用互联网作为平台来进行线上交流或交接工作提供了极大的便利。传统的线下教育管理正逐步与线上教育管理相结合。课程网站不仅是一个桥梁,更是一个巨大的教育资源库,方便师生间信息实时交换。

传统的课程网站仅提供了一些简单的教学素材,对于专业课程缺乏生动的动态描述,导致学生自主学习效率低;配备的教学措施和一些功能设计的缺失,让网站的利用率并不高。

为了解决这些问题,该系统设置了动态多媒体课件,从平台的功能模块入手,探讨改进策略,达到界面简洁,功能完善。本课题基于 JavaEE 平台实现一个大型数据库课程网站,系统主要包括:用户注册与登录、课程章节管理、课程在线学习、课程资源管理、课程资源查询、课程资源下载等功能。

1.2 研究现状

针对课程网站,已经有不少学者对此展开了研究,也开发了一些实用的课程网站,比如:

李慧卿,卢昕等人^[1]设计的学习网站采用 C/S 架构模式实现了多个功能模块,并优化升级系统。

单树倩等人^[2]基于 SpringBoot 和 Vue 框架,采用 SQL Server 数据库进行数据存储、管理,开发了一个课程网站。

李艳杰等人^[3]以 MYSQL 数据库为例,分析了基于平台的数据库综合设计内容及实现过程,包括数据库表设计、查询设计、视图设计、索引设计、触发器设计、存储过程设计等。

纪娜等人^[4]提出依托《JSP 动态网站开发》课程混合教学平台,将学生按照职业兴趣分组、采用项目化教学手段,在 PDCA 过程质量管理方法指导下,使用“教、学、做、评”的教学法实施教学。

施灵等人^[5]弥补了传统管理信息系统效率不高等缺陷,提出了一种基于大数据库的大型管理信息系统设计方案,采用 IBM 数据库,设计上传数据模块、RowKey 软件等功能,实现了对数据的快速计算、分析和处理,有较高的响应速度和扩容

窗口设计，能够更好地满足企业的信息管理需求。

1.3 本文组织结构

第 1 章：引言。从主要研究内容出发，阐述了项目的背景以及意义，结合国内的研究现状，介绍文本的主要工作和组织结构。

第 2 章：相关技术。主要介绍了项目开发软件和开发环境，并对开发过程中所需的关键技术做了简要介绍。

第 3 章：系统需求分析。主要介绍课程网站的可行性和功能需求分析。

第 4 章：系统设计。介绍系统设计的功能模块、软件结构和课程网站运行流程图。

第 5 章：系统实现。主要介绍系统各个功能模块的实现过程。

第 6 章：系统测试。利用黑盒测试方法对系统模块进行测试，找出系统存在的缺陷。

2 相关技术

本课题需要用到 MySQL、Eclipse、JavaEE、JSP 等相关技术实现系统。

2.1 MySQL

数据库不仅是一个实体，更是用来存放数据的一个仓库，同样也是一个能够长期存储在计算机内的数据集合。虽然数据库的存储空间很大，但也不能够随意地存放数据，一旦安排不妥当会导致查询效率低。

存储引擎是 MySQL 的核心，掌控着引擎的创建、数据的查询、数据的存储等功能。MySQL 数据库中的每条数据都有数据类型，主要包括数值型、字符串型、日期型三种类型；而且使用成本很低，运行速度还快，容易学习。

2.2 Eclipse

Eclipse 作为一个可以提供开放源代码的平台，通过插件来形成开发环境，本身只是一组服务，一个框架，但是其扩展性很强，在创建一个项目时，通常需要主动配置 JDK、创建 servers 服务、安装 Tomcat 服务器，环境搭建不成功则无法启动本机项目，亦不能对外提供访问。

2.3 B/S

B/S 架构是基于 Web 的一种开发模式，它将系统的核心功能集中到服务器上，使得系统的开发、维护和使用变得更加简单。同时，B/S 架构还统一了客户端，使得系统的使用更加方便。B/S 架构的系统更加灵活。

2.4 JSP

JSP 技术以 Java 作为脚本语言，在服务器上运行，给用户的 HTTP 请求提供服务。JSP 在静态的页面中插入 Java 代码，实现动态页面的转换，同时还能对数据进行布局。同时，Include 指令可以将另一个文件完全包含到当前文件中。开发者可以使用 JSP 标签扩展 API 添加自定义动作。JSP 技术具有面向对象、广泛使用和高安全性能等优势。

2.5 Servlet

Servlet 是一些运行于 Web 服务器端的 Java 小程序，用来扩展 Web 服务器的功能，它用特定的 Java 解决方案代替了其他 Web 服务器的编程模式。程序处理请求和发送响应的过程是由 Servlet 程序来完成的，它的执行原理是：当服务器接收到客户端浏览器的请求后，会解析请求 URL 路径，获取访问的 servlet 的资源，它还会查找 web.xml 文件，查看是否有对应的<url-pattern>标签内容。

3 系统需求分析

需求分析的目标是整理用户对软件开发提出的要求和需要，形成清晰、规范的文档。

3.1 可行性分析

可行性研究是评估项目是否可行的一种方法，以经济效益为核心，需要全面、系统地分析各种因素对项目的影响，并使用大量数据资料来论证项目是否可行。

3.1.1 技术可行性

JavaEE、JSP、Eclipse 和 MySQL 都是成熟的技术，具有良好的稳定性、可靠性和安全性，已经被广泛应用于程序开发。JavaEE 平台提供了一系列的 API 和工具，可以帮助快速构建 Web 应用程序。MySQL 为确保数据的高可用性和可靠性，支持主从复制和多主复制；提供多种安全功能，如访问控制和数据加密；作为开源的数据库管理系统，可以自由地使用和修改。JSP 技术的跨平台性体现在可以在多种硬件平台上运行，与传统 Web 开发技术颇为相似，比较易于使用。Eclipse 平台支持多种编程语言的开发，开放性体现在可以不断更新和改进程序。综上所述，系统的开发在技术上都具有一定的可行性。

3.1.2 经济可行性

JavaEE、JSP、Eclipse 和 MySQL 都是免费的开源技术，能节省开发成本、提高开发效率的同时还具有良好的社区支持和文档资料。Eclipse 具有完备的功能，在编写运行代码出错时会有提示，能够有效地提高编程的效率，而且 MySQL 数据库功能齐全，能够对数据库中的数据进行处理和备份。另外，应用本系统后可以显著地提升学校对教育的线上课程管理，具有潜在的市场价值。综上所述，本系统在经济上是可行的。

3.2 功能需求分析

整个系统有三种用户：管理员、教师、学生

3.2.1 管理员用户实现的功能

- (1) 管理员管理：修改管理员的身份和用户状态等信息；
- (2) 学生管理：修改学生的身份和用户状态等信息；
- (3) 教师管理：修改教师的身份和用户状态等信息；

- (4) 文章管理：创建、编辑文章信息；
- (5) 论坛管理：更改帖子状态和论坛信息；
- (6) 资料管理：管理员可以在本地下载文件或视频，也能播放本地视频；
- (7) 课程管理：管理员能上传文件和视频，对课程内部信息进行修改；

以下为管理员用户的用例图，如图 3.1 所示：

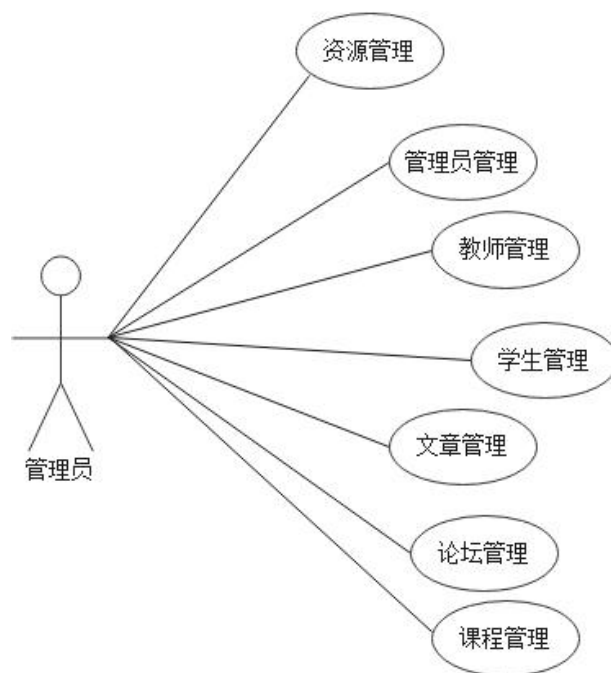


图 3.1 管理员用户用例图

3.2.2 教师用户实现的功能

- (1) 学生管理：可以修改学生的身份和用户状态等信息；
- (2) 课程管理：查阅自己所教的课程信息；
- (3) 文章管理：阅读、添加、删除文章信息；
- (4) 试题管理：查看、修改试题信息；
- (5) 资料管理：对资料信息可任意查看；

以下为教师用户的用例图，如图 3.2 所示：

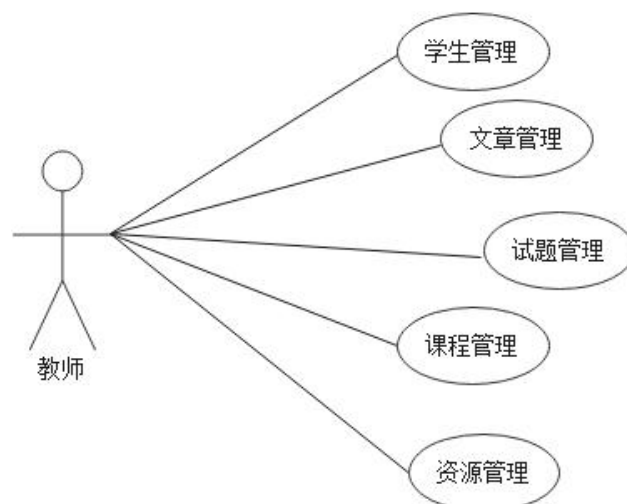


图 3.2 教师用户用例图

3.2.3 学生用户实现的功能

- (1) 学习平台：学生能了解自己的学习信息；
- (2) 视频教程：对于感兴趣的视频教程可以观看、下载；
- (3) 资料下载：可以查找、删除、下载资料；
- (4) 试题练习：用户可以完成试题，提高能力；
- (5) 学习论坛：与教师、同学交流学习信息；
- (6) 关于其他：关于该系统的简介；

以下为学生用户的用例图，如图 3.3 所示：

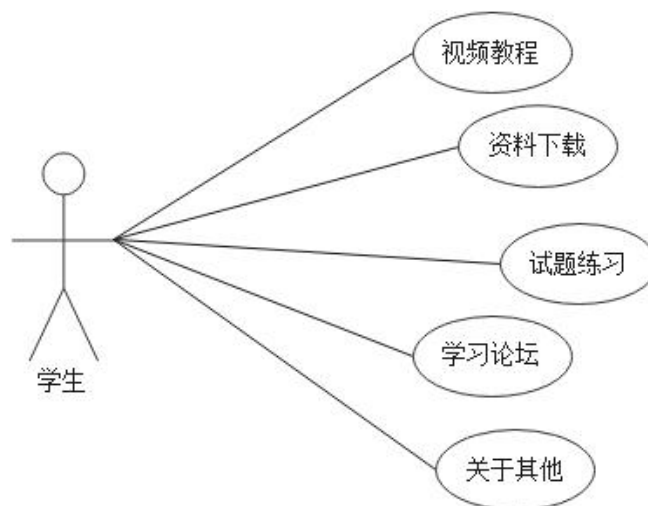


图 3.3 学生用户用例图

4 系统设计

该部分包括功能模块设计和系统流程图设计。

4.1 功能模块设计

该部分包括三类角色的功能模块设计，分别是管理员、教师、学生。

4.1.1 管理员功能模块

管理员功能包括用户管理、个人信息管理、课程管理等部分。管理员具有最高权限，可以指定用户注册登录，管理个人用户信息，包括更改个人信息、密码、用户名等。管理员还可以对教师端、学生端、管理员用户端、文章管理等进行操作。管理员功能模块如图 4.1 所示：

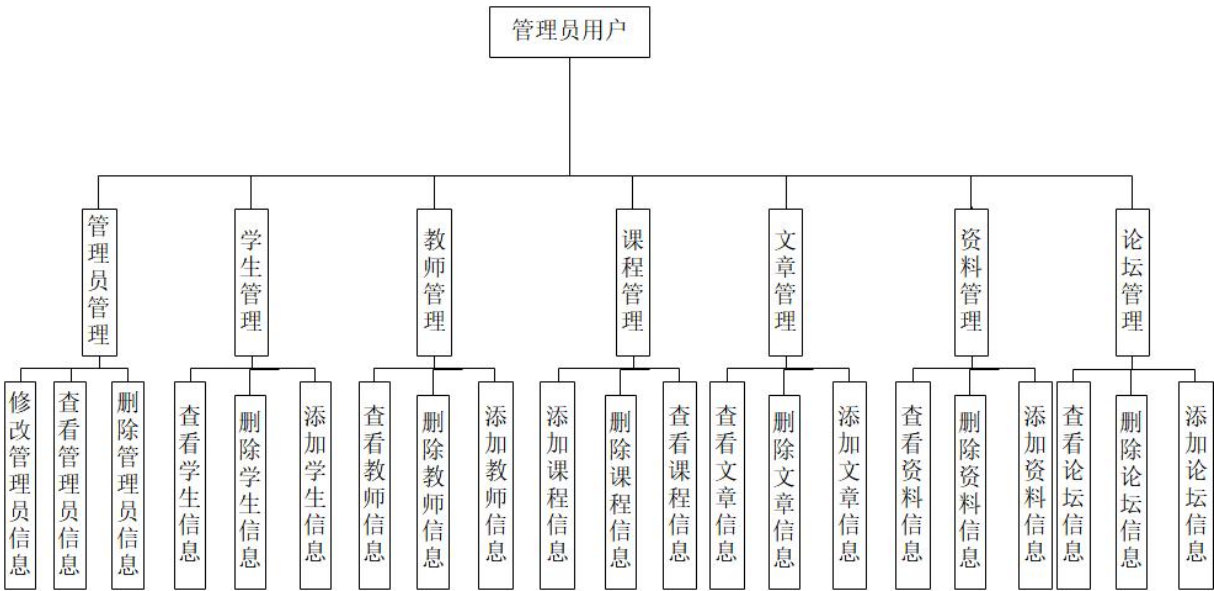


图 4.1 管理员用户功能模块图

4.1.2 教师用户功能模块

教师用户一共有 5 个模块，输入正确的用户名和密码后进入教师页面，可以对学生、课程、试题、资料进行管理，也可以进行删除、新增或修改，教师功能模块如图 4.2 所示：

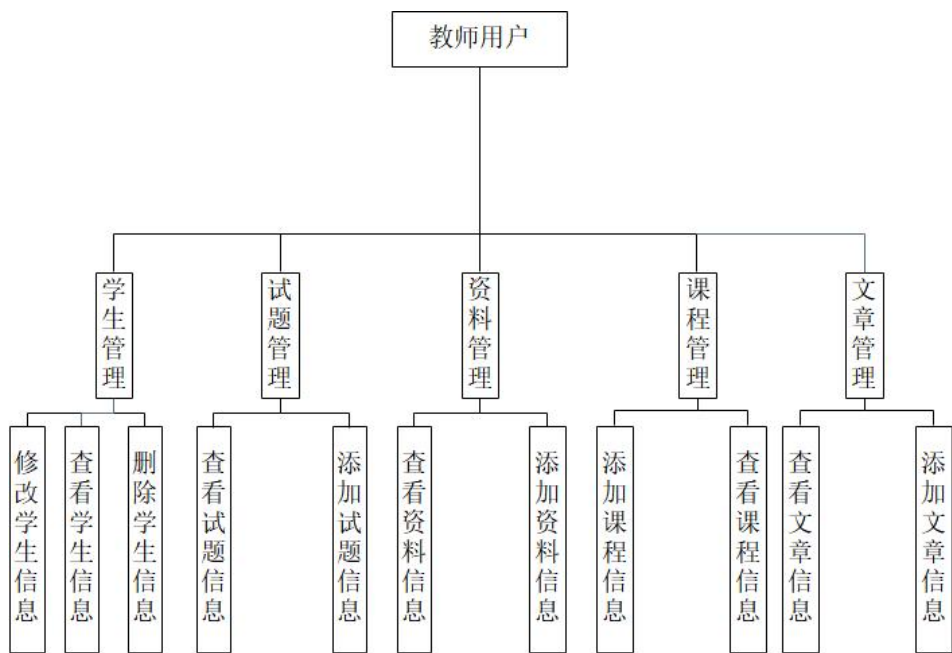


图 4.2 教师用户功能模块图

4.1.3 学生用户功能模块

学生用户一共有七个模块，学生端用户输入正确的用户名和密码就可以进入学生页面。在学习平台可以看见相关书籍的课程介绍；对教师发布的试题练习可以查看，下载并练习，还可以删除；在论坛上可以发布自己的帖子，与同学和老师进行交流功能齐全，学生功能模块如图 4.3 所示：

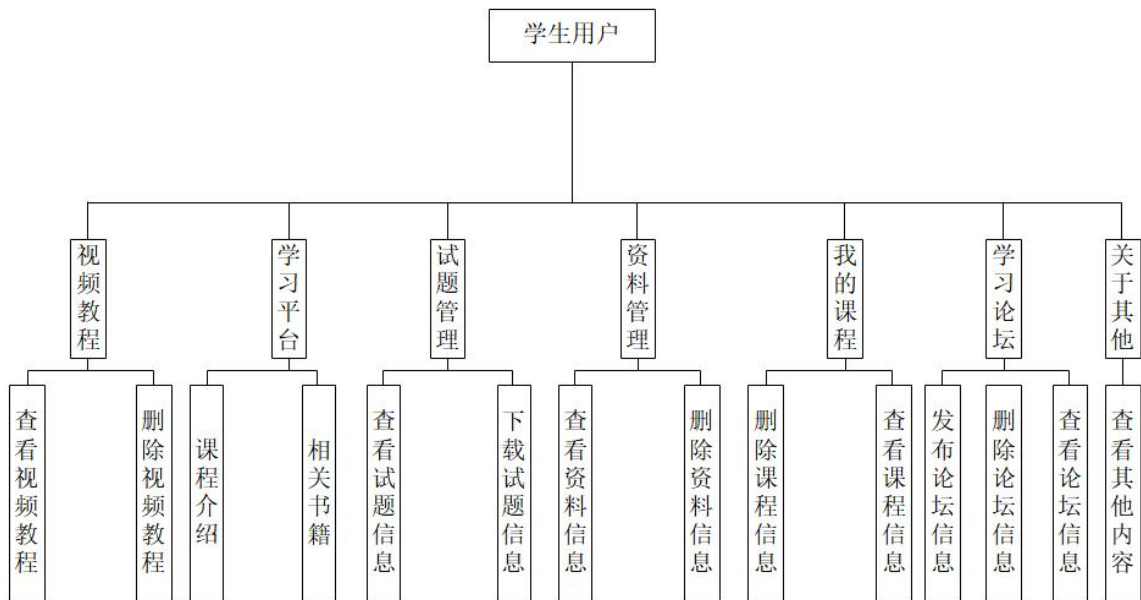


图 4.3 学生用户功能模块图

4.2 系统流程图设计

该系统分别由管理员、学生、教师三个子系统构成。三个用户分别需要输入用户名和密码进行注册登录，进行身份验证，若不成功则返回登录注册页面，反之则进入相对应界面。

管理员用户成功进入页面后系统可以对管理员、学生、教师进行一系列管理，还可以对课程、文章、资源等进行操作；教师用户，进入界面后系统可修改管理学生信息，对视图、资料、课程、文章都有权限进行修改、查看等功能；学生用户相对于其他两个用户权限会少一些，可以在学习平台查看相关书籍的资料，可以下载试题、查看课程、观看视频学习知识，也可以在论坛上发表自己的言论，与同学和老师进入互动，系统流程如图 4.4 所示：

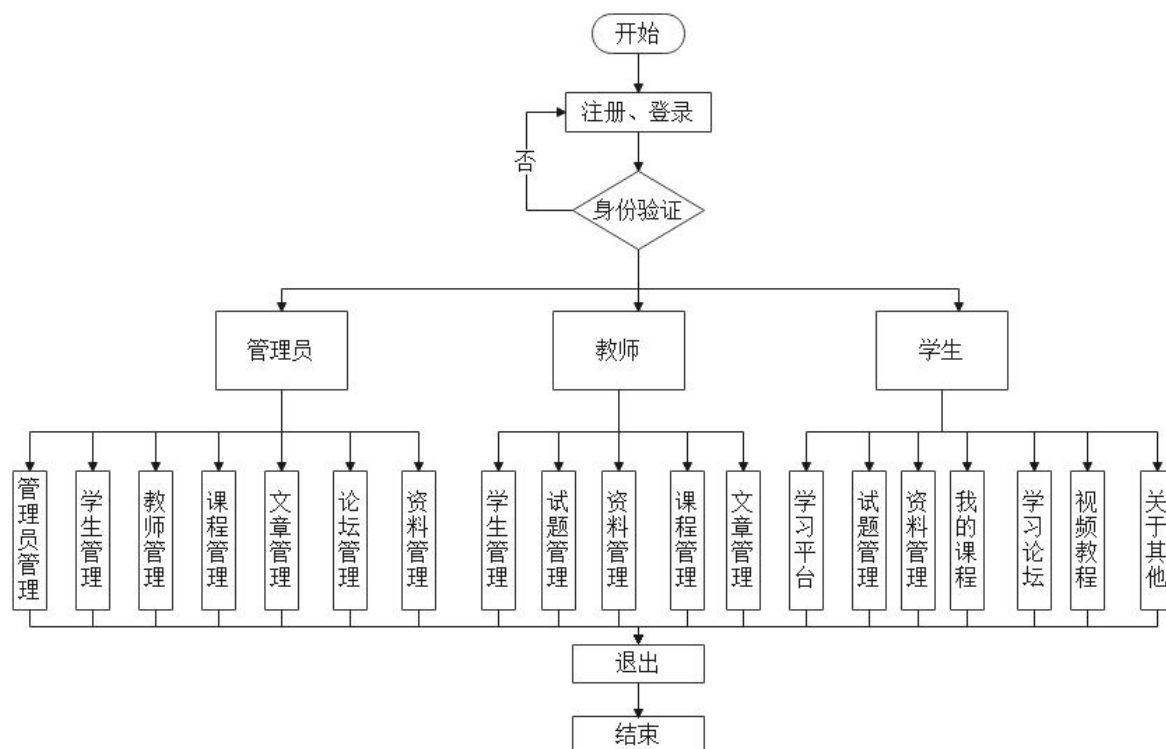


图 4.4 系统流程图

4.3 总类图设计

此图全面地描述各类图之间的联系，管理员具有最高权限，因此一个管理员可以控制管理多个学生用户、教师用户、课程信息、文章信息、论坛信息以及资料管理，故联系为 1: N ；而对于教师和学生而言，一个教师可以查看多个学生信息，所以联系为 1: N ；一个学生可以查看多个课程章节、阅览多个文章信息、发布多个评论、下载多个资料（文件、视频），故联系为 1: N ；同理一个教师

可以查看、发布多个课程、阅览多篇文章及下载、可回复多个学生的帖子、上传查看各种各样的资料，所以联系为 1：N ， 如图 4.5 所示：

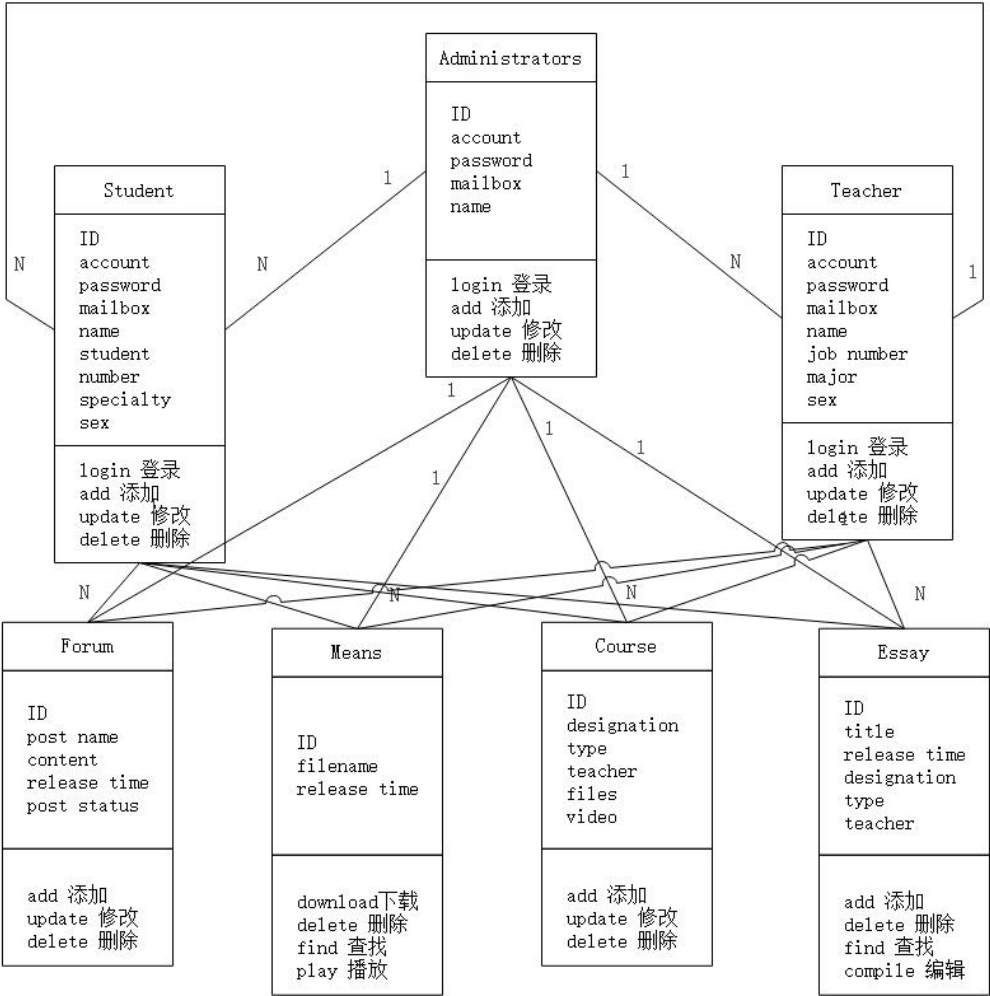


图 4.5 系统类图

4.4 数据库设计

数据库是指对于一个给定的应用环境，构造最优的数据库模式，建立数据库及其应用系统，使之能够有效地存储数据，满足各种用户的应用需求。

4.4.1 概念结构设计

主要利用 E-R 图来展示实体与实体之间的关系，E-R 图中三个核心部分：实体、属性、联系，最后进行规划；拥有一个完整的数据库不仅能提高效率，还能让应用程序的开发变得容易。以下是系统 E-R 图：

(1) 管理员实体图

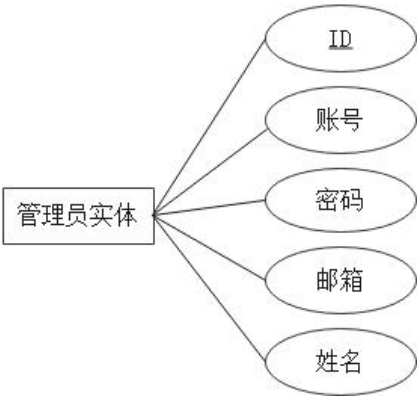


图 4.6 管理员实体

(2) 教师实体图

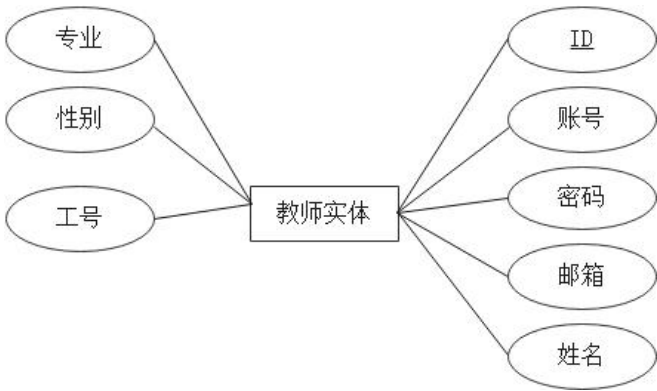


图 4.7 教师实体

(3) 学生实体图

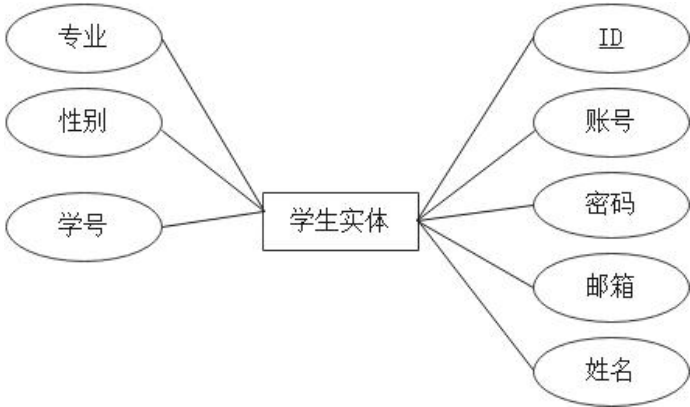


图 4.8 学生实体

(4) 课程管理实体图

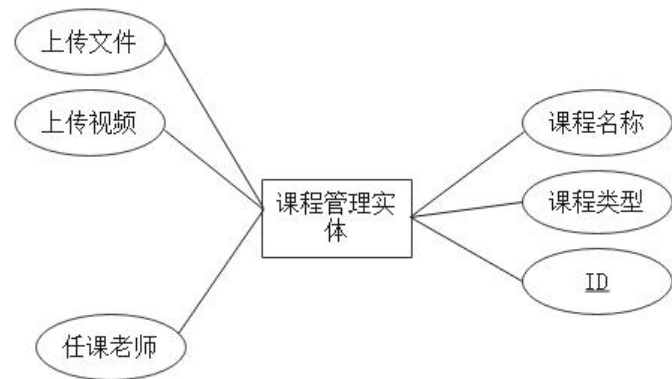


图 4.9 课程管理实体图

(5) 文章管理实体图

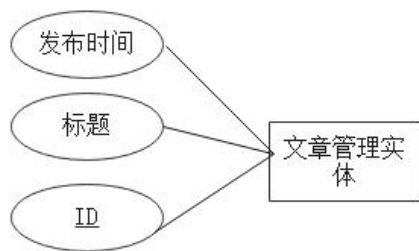


图 4.10 文章管理实体

(6) 论坛管理实体图

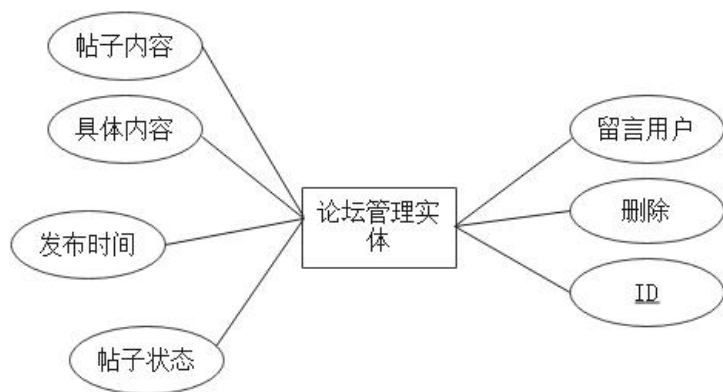


图 4.11 论坛管理实体

(7) 资料管理实体图

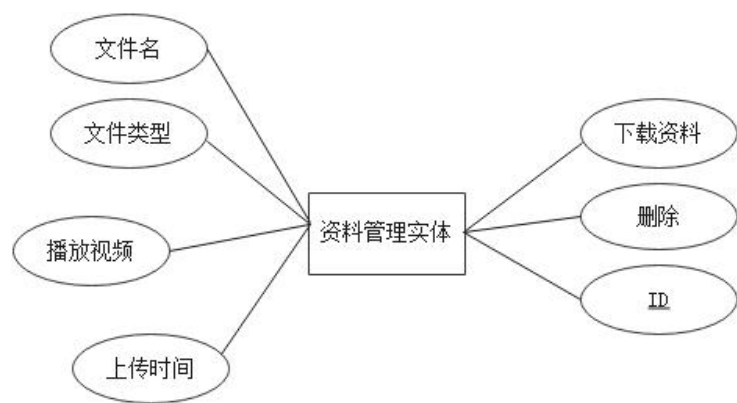


图 4.12 资料管理实体图

(8) 总系统实体图

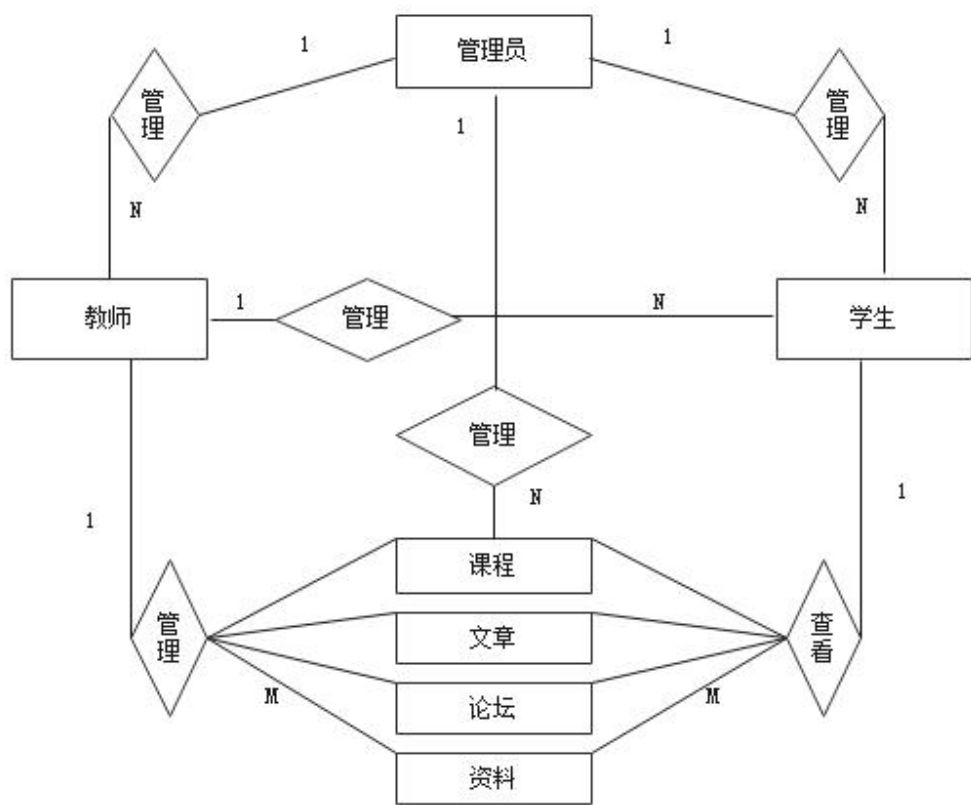


图 4.13 实体联系图

4.4.2 逻辑结构设计

逻辑结构设计就是将概念模型转换成数据模型，也就是将实体（E-R 图）模型转换成关系模型，如下表所示：

(1) 管理员信息表

管理员包括（ID、Name、Password、Email、Sex）

表 4.1 管理员信息表

字段名	类型	是否为空	中文翻译	备注
ID	int	否	编号	主键
Name	varchar(20)	是	名字	
Password	varchar（255）	是	密码	
Email	varchar（100）	是	邮箱	
Sex	varchar（10）	是	性别	

（2）教师信息表
教师包括（ID、Name、Password、Email、Jno、Sex）

表 4.2 教师信息表

字段名	类型	是否为空	中文翻译	备注
ID	int	否	编号	主键
Name	varchar(20)	是	名字	
Password	varchar（255）	是	密码	
Email	varchar（100）	是	邮箱	
Jno	int	是	工号	
Sex	varchar（10）	是	性别	

（3）学生信息表
学生包括（ID、Name、Password、Email、Sno、Sex、Major）

表 4.3 学生信息表

字段名	类型	是否为空	中文翻译	备注
ID	int	否	编号	主键
Name	varchar(20)	是	名字	
Password	varchar（255）	是	密码	
Email	varchar（100）	是	邮箱	
Sno	int	否	学号	
Sex	varchar（10）	是	性别	
Major	varchar（255）	否	专业	

（4）文章管理信息表

文章管理信息表包括（ID、Title 、Article 、Time、Name）

表 4.4 文章管理信息表

字段名	类型	是否为空	中文翻译	备注
ID	int	否	编号	主键
Name	varchar(255)	是	名字	
Title	varchar (50)	是	标题	
Article	text	是	文章操作	
Time	varchar (50)	是	时间	

（5）课程管理信息表

课程管理信息表包括（ID、Cname、Type、Tname、Time）

表 4.5 课程管理信息表

字段名	类型	是否为空	中文翻译	备注
ID	int	否	编号	主键
Cname	varchar(255)	是	课程名字	
Type	varchar (255)	是	类型	
Tname	varchar (255)	是	任课教师	
Time	varchar (255)	是	时间	

（6）论坛管理信息表

论坛管理信息表包括（ID、Title、Pubtime、Detail 、Edituser、State）

表 4.6 论坛管理信息表

字段名	类型	是否为空	中文翻译	备注
ID	int	否	编号	主键
Title	varchar(50)	否	标题	
Pubtime	varchar (50)	是	发布时间	
Detial	varchar (255)	否	具体内容	
Edituser	varchar (50)	是	操作	
State	int	是	贴子状态	

（7）资料管理信息表

资料管理信息表包括（ID、TitleOrig、TitleAlter、Size、Type、Path、Uploadtime、Count、Courseid）

表 4.7 资料管理信息表

字段名	类型	是否为空	中文翻译	备注
ID	int	否	编号	主键
TitleOrig	varchar(255)	是	原始标题	
TitleAlter	varchar (255)	是	更新标题	
Size	varchar (50)	是	大小	
Type	varchar (255)	是	类型	
Path	varchar(255)	是	路径	
Uploadtime	varchar(255)	是	上传时间	
Count	int	是	计数	
Courseid	int	是	课程	

4.4.3 物理结构设计

该系统采用 MySQL 作为数据库存储数据，前端操作使用的是 Navicat 软件，它作为 MySQL 的界面化软件，可以对本机或远程的 MySQL 数据库进行开发、管理；数据文件存储的位置 C:\ProgramData\MySQL\MySQL Server 5.1\Data，数据文件名为 id_cache.db，日志文件是 ib_logfile0.txt。

5 系统实现

5.1 公共模块

公共模块包括用户登录和注册两部分，一共有三种角色，分别是管理员、教师、学生，各自页面功能展示有所不同。

5.1.1 用户登录

用户登录模块封装了 SecurityUtils 类，运用@PathParam 注解、HttpServletRequest 对象和 Subject 对象等。同时，也涉及到了用户身份验证和权限控制的相关操作，如下图所示：



图 5.1 用户登录图

登录页面的核心代码如下：

```
public String login(@PathParam("username")           //关联 login. JSP 页面
String username,
@PathParam("password")
String password,HttpServletRequest request)
{
    User user = us.getUserByUsername(username);
    if(user!=null) {
        if (user.getRoleId() == 2) {           //若用户不为空，得到 RoleId、State 两种结果
            if (user.getState() == 1) {
                Subject subject = SecurityUtils.getSubject();
            }
        }
    }
    ... }
```

5.1.2 用户注册

用户注册这部分使用了 Servlet API 提供的 `setAttribute()` 方法，用于给请求附加一个属性值，以便在页面进行传递和显示。使用了 `if/else` 语句来判断用户是否已经存在，将用户信息存入数据库中，使用 `CryptographyUtil` 工具类对密码进行加密处理，如下图所示：



图 5.2 用户注册图

注册页面的核心代码如下：

```
if(us.getUserByUsername(user.getUsername())==null){
    user.setPassword(CryptographyUtil.md5(password,"salt"));
    return "login"; } //若注册成功，则跳转到 login. JSP 页面
else{
    request.setAttribute("errorMsg", "用户名存在，注册失败");
    return "register"; //注册失败，返回注册
}
```

5.2 管理员模块

管理员具有最大权限，可以实时切换账号的各种角色和是否激活状态。

5.2.1 控制学生端

控制学生端这部分使用了 Spring MVC 框架，其中包含了两个方法：

`deleteUser`（删除用户）和 `changeRole`（修改用户角色）。在修改用户角色时，先使用 `getByID` 方法获取用户对象，然后设置角色 ID，最后调用 `changer` 方法进行更新，如下图所示：



图 5.3 学生列表图

核心代码：

```
public String changeRole(int id,int role) {
System.out.println("====执行了用户角色切换===="+role);
User user = us.getByID(id);                                //使用 get 方法搜索学生 ID
...                                                         //设置角色 ID
    if (us.changer(user)) {
        return "redirect:/getAllUser";                      //角色切换成功
    } else {
        return "error";}}                                  //角色切换失败
```

5. 2. 2 控制教师端

该部分将教师 ID 作为参数，调用 UserService 中的 getTeacherByID 方法获取对应的教师对象，然后根据教师对象的当前状态（0 表示未激活，1 表示已激活），将其状态进行切换，如下图所示：



图 5.4 教师列表图

核心代码：

```
public String changeTeacherState(int id) {
User user = us.getTeacherByID(id);
```

```

System.out.println("====执行了用户状态切换====" + user.getRoleId());
    if (user.getState() == 0) {
        user.setState(1);                //使用 0 和 1 表示激活和未激活两种状态
    } else {
        user.setState(0);}
    if (us.changes(user)) {
        return "redirect:/getAllTeacher";    //更改教师账号状态
    } else {
        return "error"; }}

```

5. 2. 3 控制管理员端

从数据库中获取管理员数据，使用 ID 作为参数。然后调用 UserService 类中的 changer 方法，将管理员对象的角色信息更新。如下图所示：

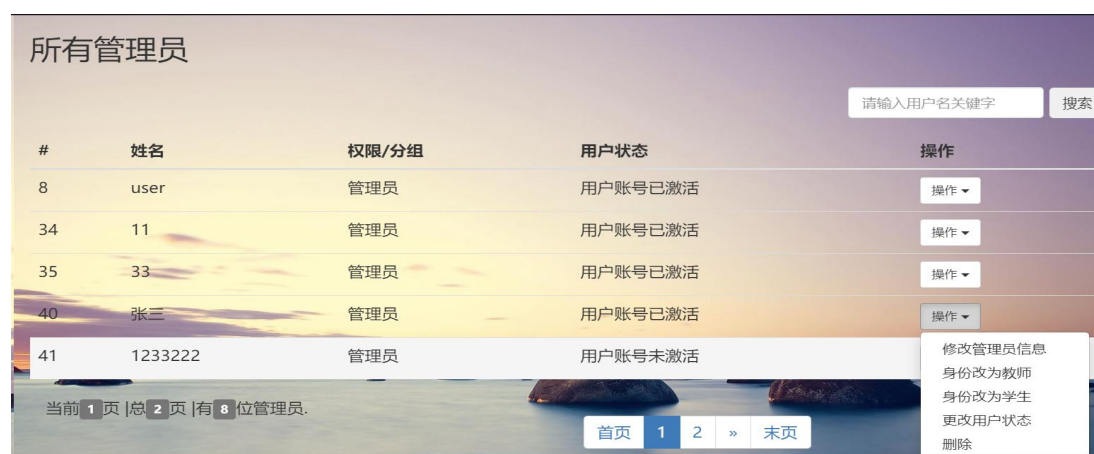


图 5.5 管理员列表图

核心代码如下：

```

public String changeAdminRole(int id,int role) {
System.out.println("====执行了用户角色切换===="+role);
User user = us.getByID(id);                //使用 get 方法搜索管理员 ID
...
user.setRoleId(role);                      //设置角色 ID
    if (us.changer(user)) {
        return "redirect:/getAllAdmin";    //角色切换成功
    } else {
        return "error";                    //角色切换失败
    }
}

```

5. 2. 4 课程管理

首先创建了一个 Course 对象,并使用 SimpleDateFormat 语法构建日期格式。然后调用 courseService 类的 addCourse 方法将课程对象添加到数据库中,如下图所示:

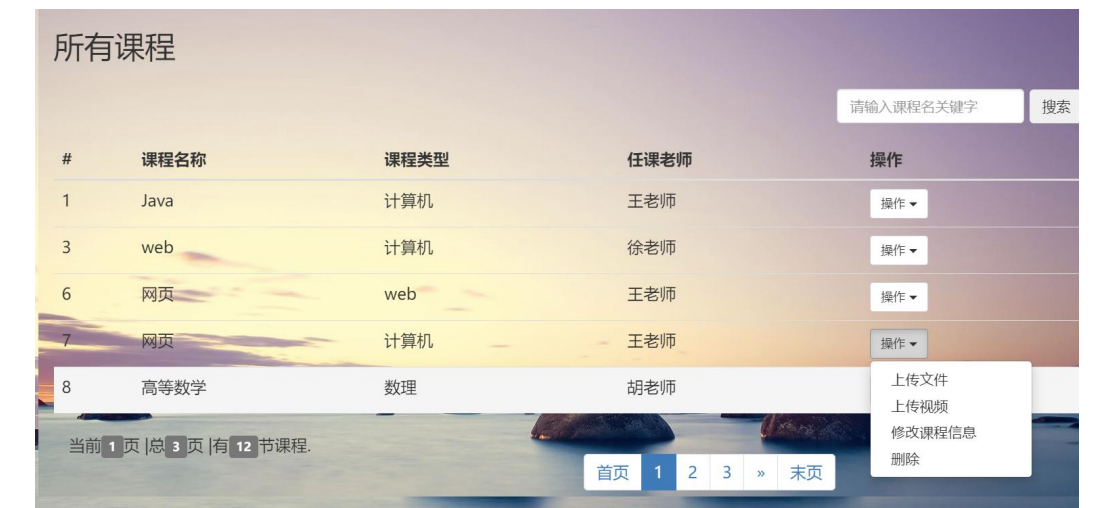


图 5.6 课程管理信息图

核心代码:

```
public String insertCourse( String coursename,           //插入教师名、课程名、课程类别
SimpleDateFormat dateFormat =           //使用 SimpleDateFormat 语法构建日期格式
new SimpleDateFormat("yyyy.MM.dd");...
    if (courseService.addCourse(course)) {
        System.out.println("");           //添加成功
        return "redirect:/getAllCourse";
    } else {...}}
```

5. 2. 5 文章管理

自定义了 DateUtil 类来获取当前时间,使用 HttpServletRequest 和 HttpSession 对象来获取和设置请求和会话属性,用@RequestParam 注释来获取请求参数。此外,还使用 NewsService 类来执行数据库操作,如下图所示:

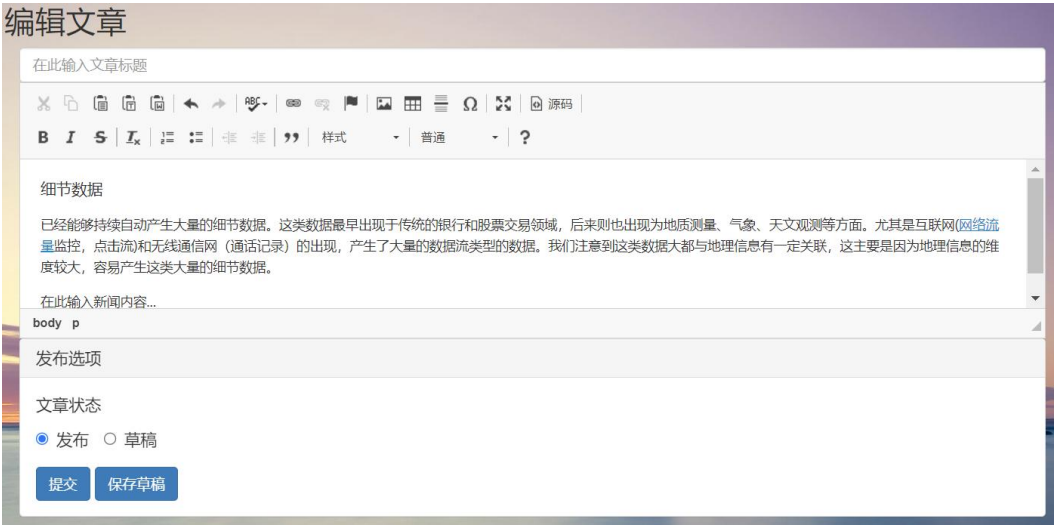


图 5.7 文章管理信息图

核心代码：

```
public String deleteNewsById(HttpServletRequest req
,@RequestParam(value="id")Integer id){
System.out.println("==执行了删除==");
    if(ns.deleteNewsById(id)){
        req.setAttribute("msg", "");           //删除文章成功
        return "redirect:/anews";
    } else{
        return "error";
    }
}
```

5.2.6 论坛管理

使用 AdmindeleteTopic 方法接收一个帖子 id 作为参数,调用 TopicService 类中的 delete 方法删除该帖子，并返回一个重定向到 selectAllTopic3 页面的字符串，如下图所示：



图 5.8 帖子管理列表图

核心代码：

```
public String updateState(@PathParam("id") int id){
    Topic topic = ts.selectTopicById(id);
    System.out.println("====="+topic.getState());           //执行了帖子状态切换
    if(topic.getState()==0){
        topic.setState(1);           //用 0 和 1 表示开启和屏蔽两种状态
    }    ...}
```

5.2.7 资料管理

（1）调用 ss.deleteSourceById 方法删除数据库中的记录，然后利用 HttpServletRequest 对象获取文件的真实路径，创建一个 File 对象并判断文件是否存在，如果存在则调用 delete 方法删除文件，如下图所示。

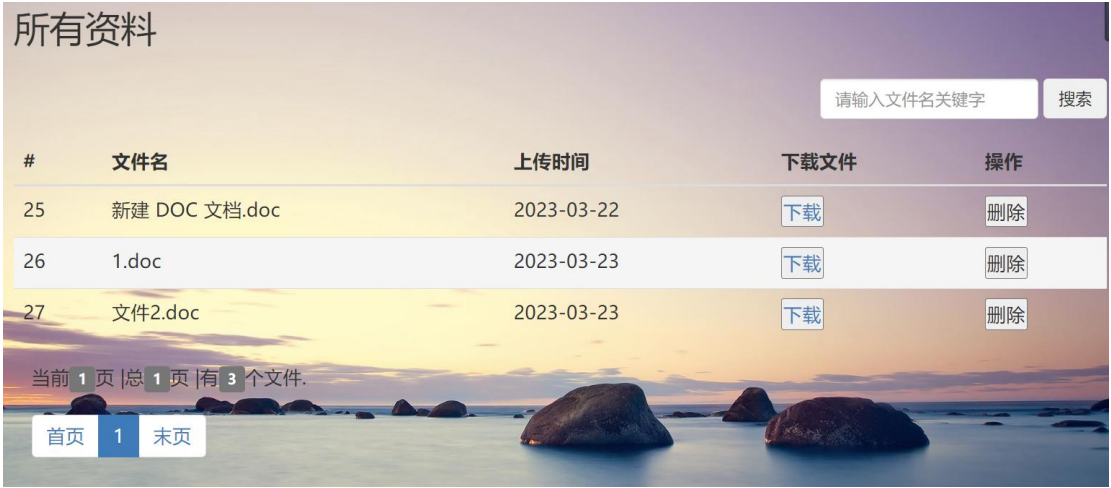


图 5.9 文件上传列表

核心代码：

```
public String deleteSourceById(Integer id,
    HttpServletRequest request) {           //利用 delete 删除文件
    Source source = ss.findSourceById(id);
    if(source!=null) {
        String filename = source.getFilename();
        ss.deleteSourceById(id);
        ...
    }
}
```

（2）接收一个 id 参数，表示要播放的文件的唯一标识符。通过调用 vs.findById(id) 方法，获取上传文件的实体对象，然后创建一个 ModelAndView 对象，其中包含视图名称和模型数据，并将其返回，如下图所示。



图 5.10 视频上传列表

核心代码：

```
public ModelAndView adminplay(@PathParam("id")
int id, ModelMap model) {
FileEntity entity = vs.findById(id);           //上传文件实体
ModelAndView view = new ModelAndView("admin/movie/playVideo", model);    //播放视频
return view;
}
```

5.3 教师模块

教师的权限仅次于管理员，不同于管理员的是多了试题管理部分。

5.3.1 学生管理

该模块使用了@RequestParam 注解来获取请求参数，HttpSession 和 HttpServletRequest 来获取会话和请求对象，if-else 语句来判断用户是否为空，getUserByID 方法来获取用户信息，setAttribute 方法来设置请求属性，最后返回了一个视图名，如下图所示：



图 5.11 学生修改信息图

核心代码：

```
public String tchangeKeypage(@RequestParam
(value = "id", required = false) Integer id,
HttpSession session,HttpServletRequest req) {
    if( id!=null){                                //使用 if 语句判定用户是否为空
        User user = us.getUserByID(id);          //使用 get 方法获取用户 id
        req.setAttribute("user", user);
    }    else{                                    //使用 else 语句与 if 相关联
    .... } return "teacher/user/tupdate";}
```

5. 3. 2 文章管理

该部分使用 PageHelper 插件来实现分页查询功能，调用 selectAllNews 方法查询所有文章，并将结果封装到 PageInfo 对象中，最后将 PageInfo 对象存储到 HttpServletRequest 对象中，并返回一个 JSP 页面用于显示查询结果，如下图所示：



图 5.12 文章列表图

5. 3. 3 课程管理

接收一个 Course 对象和一个 HttpServletRequest 对象作为参数，并返回一个字符串，取当前日期并将其格式化为字符串，然后将其设置为课程对象的时间属性。调用 CourseService 的 update 方法来更新课程信息，如下图所示：



图 5.13 我的课程信息图

5. 3. 4 试题管理

(1) 接受三个参数：一个 Itembank 类型的 List 集合 list，一个 Itembank 类型的 List 集合 i 和一个整数 num。从集合 i 中随机选取 num 个元素，添加到集合 list 中，并从集合 i 中移除已选取的元素，最后返回更新后的集合 list，如下图所示。



图 5.14 我的题库信息图

核心代码：

```
public List<Itembank> ran(List<Itembank> list,List<Itembank> i,int num){
    for (int j = 0; j < num; j++) {
        Random random = new Random();
        int rom=random.nextInt(i.size());
        list.add(i.get(rom));
        i.remove(rom);
    }
    return list;
}
```

(2)将当前用户的 ID 从会话中获取，并将其设置为 Examination 对象的 userid 属性，然后将 Examination 对象插入到数据库中，并创建一个 ExamVO 对象列表。使用 count () 方法迭代 ExamVO 对象列表，并将每个对象的测试和答案添加到列表中，最后使用 for 循环迭代方法来遍历 List 集合中的元素，如下图所示。

习题信息

练习类型

判断题

练习题目

牛顿发现了万有引力吗?

选项A:

对

选项B:

错

答案:

对

确定

图 5.15 试题添加信息图

核心代码：

```
public String exam(Examination e, HttpServletRequest req,
HttpSession session) {
int userid = (int) session.getAttribute("id");
int c = 0;
    for (int i = 0; i < list.size(); i++) {           //使用 for 循环迭代方法
        if (list.get(i).getAnswer1().equals(list.get(i).
getItembank().getAnswer())) {
            c++;}
    }
```

5.4 学生模块

学生的权限是最小的，除了在学习论坛可以发表、评论帖子而外，对大部分功能的操作权限只有查看、删除。

5.4.1 学生总界面

这部分定义了三个请求处理方法，分别对应不同的 URL 请求。supdate() 方法映射了 ‘supdate’ 请求，about() 方法映射了 ‘about’ 请求，shome() 方法映射了 ‘shome’ 请求，如下图所示：



图 5.16 学生模块总展示图

5.4.2 我的课程

使用 Spring MVC 的 Controller 方法，用于处理跳转到“我的课程”页面。首先从 session 中获取教师的用户名，然后使用 PageHelper 插件进行分页查询该教师的课程列表。查询结果存储在 PageInfo 对象中，最后将该对象存储在 HttpServletRequest 对象中，返回到指定的视图页面，如下图所示：



图 5.17 我的课程信息图

核心代码：

```
public String myCourse( //跳转到我的课程
    HttpServletRequest req, HttpSession session) { //RequestParam 注解用于获取请求参数
    PageHelper.startPage(pn, 5); //实现分页查询
    PageInfo pageUser = new PageInfo(courses); //实现分页查询
}
```

5.4.3 试题练习

接收一个 HttpServletRequest 对象作为参数，并返回一个字符串。创建了一个空的 Itembank 列表，并通过调用 itembankService 的三个方法分别获取三种类型的 Itembank 对象，并将它们存储在 i1、i2 和 i3 列表中，如下图所示：



图 5.18 试题练习信息图

核心代码：

```
public String publishCourse(  HttpServletRequest req) {
List<Itembank> i1= itembankService.selectTypeOne();    //跳转题目
req.setAttribute("list", list);           //setAttribute 将随机选取的题目列表设置为请求属性
return "student/exercises";}                //得出成绩，返回首页
```

5.4.4 学习论坛

首先打印出 edituser 参数的值, 然后调用 findAllByName 方法来查询指定用户名下的所有主题, 查询结果使用 PageHelper 进行分页处理, 每页显示 5 条记录, 查询结果被封装在一个 PageInfo 对象中, 并设置为 req 对象的属性, 最后, 方法返回一个字符串 ‘student/topicList’, 用于指示返回的视图名称, 如下图所示：



图 5.19 所有帖子信息图

5.4.5 视频教程

创建了一个文件输入流对象 in，用于读取指定路径下的文件‘fileDir’。如果文件不存在，则会抛出 FileNotFoundException 异常并打印异常信息，如下图所示：



图 5.20 视频信息图

5.4.6 资料下载

资料下载部分使用 Spring 框架中的@RequestParam 注解来获取 HTTP 请求中的参数，其中包括当前页码和每页显示的记录数，还使用了 PageHelper 插件来实现分页查询，将查询结果封装为一个 List 集合，并返回给前端页面，如下图所示：



图 5.21 文件资料信息图

6 系统测试

该测试旨在检验整个系统的完整性和一致性，包括硬件、软件和操作人员的协同作用，可以发现系统分析和设计中的错误，确保系统的稳定性和可靠性。

6.1 测试方案

系统测试时非常重要的一个步骤，本系统采用黑盒测试的方案，黑盒测试主要用于验证系统是否有不正确或者遗漏的一些功能，性能上有没有达到要求，主要测试以下这几部分：

- （1）用户登录模块的测试；
- （2）系统主要功能的测试；

6.2 测试用例

该部分是指对某款软件产品进行测试，验证是否满足特定需求，测试输入、执行条件和预期结果均需明确指定。

6.2.1 用户登录模块测试

对三个用户登录进行测试，测试结果如下表所示：

表 6.1 用户登录测试表

模块名称	测试用例	预期结果	实际结果	记录结果
管理员登录模块	用户名：null 密码：null	用户名长度不能小于 2 个字母	登录失败，请输入正确用户名	通过
教师登录模块	用户名：null 密码：null	账号未注册	登录失败，用户账号未注册	通过
学生登录模块	用户名：null 密码：null	账号未激活	登录失败，用户无权限	通过

6.2.2 新建课程模块测试

对管理员用户中的新建课程进行测试，结果如下表所示：

表 6.2 新建课程测试表

模块名称	测试用例	预期结果	实际结果	记录结果
新建课程模块	管理员用户点击新建课程	进入新建课程的新页面	进入课程信息的新页面	通过
新建课程模块	添加名称、类	其中一个未添	没有提示，但	通过

块	别、教师	加，列表中不能展示	列表中不显示	
新建课程模块	添加好新数据，最后点击确定	页面跳转到课程信息列表，显示出新的数据	跳转成功	通过

结束语

通过这几个月的努力，终于完成了课程网站的毕业设计，收获很多。

该系统综合运用到数据库、JSP 架构等知识，实现了管理员、教师、学生三个用户的登录，以及相关内容页面的展示。

通过这次毕业设计，让我对 JavaEE 技术、黑盒测试、Eclipse 平台以及 Web 前端开发有了更深层次的了解，当理论与实践相结合时，总能碰撞出一些奇妙的火花，所以对于理论知识的掌握也是非常重要的。

由于时间、学识的限制，该系统仍然存在一些问题，比如功能不够完善、数据库设计不够规范等。

毕业设计过程中遇到的技术难题，会更好地督促我查漏补缺，在今后的工作中不断完善自己的专业知识。

致 谢

此次毕业设计首先要感谢我的指导老师，他时时督促我们完成相应部分，让我们对时间有了规划，对每一部分进行评价并给出修改建议，有效节约我们的时间，为我们完成课题提供了非常大的帮助。

其次要感谢我的父母，不辞辛苦为我创造好的生活和学习环境，我一定不负你们。

在大学期间，还要感谢我室友、同学们，大家其乐融融相处，却又不缺乏相互竞争的激情，身处这样的环境，让我感觉每一天都过得很充实。

最后我要感谢我的母校南京理工大学泰州科技学院，为我们提供了一个优秀的学习环境，激励我们前进，让我们不断绽放自己的光芒。

毕业设计的结束意味着我们大学生涯的结束，也是迈入社会的开始，我将推开一扇新的大门，迎接这充满挑战的人生。

参 考 文 献

- [1] 李慧卿, 卢昕, 徐新爱. 数据结构课程学习网站的设计与实现[J]. 电脑知识与技术, 2022, 18(09):30. 32.
- [2] 单树倩, 任佳勋. 基于 SpringBoot 和 Vue 框架的数据库原理网站设计与实现[J]. 电脑知识与技术, 2021, 17(30):40. 41+50.
- [3] 李艳杰. 信息化 2.0 背景下基于平台的数据库综合设计与实现[J]. 黑龙江科学, 2021, 12(24):120. 122.
- [4] 纪娜. 基于 OBE 和 PDCA 的《JSP 动态网站开发》课程教学方法研究[J]. 科技风, 2022(4):126-128.
- [5] 施灵. 基于大数据库的大型管理信息系统优化设计研究[J]. 信息与电脑(理论版), 2022, 34(03):119. 121.
- [6] 于启红, 刘杰. 软件工程专业导论[M]. 南京大学出版社:, 202108. 205.
- [7] 丁子建, 陈柏祥, 卢佳欣, 张钰松. 教学辅助系统数据库设计[J]. 数字技术与应用, 2021, 39(07):153. 155.
- [8] Jack Belzer, Albert G. Holzman, Allen Kent. Encyclopedia of Computer Science and Technology:Volume 14 . Very Large Data Base Systems to Zero.Memory and Markov Information Source[M]. CRC Press:2021. 04. 29.
- [9] 施珺, 纪兆辉, 赵雪峰. Java 语言实验与课程设计指导[M]. 南京大学出版社:, 202101. 304.
- [10] Malhotra D, Malhotra N. Data Structures and Program Design Using Python:A Self-Teaching Introduction[M]. Mercury Learning & Information:2020. 11. 30.
- [11] Hong Wu, Wu Hong, Cheng Caijun. Research on Search Optimization Based on Oracle Database[J]. Journal of Physics: Conference Series, 2020, 1648(4).
- [12] 曹祺. 大数据时代图书馆信息系统的系统分析与设计[M]. 武汉大学出版社:, 202005. 163.
- [13] 徐立臻. 数据结构基础实验指导[M]. 南京东南大学出版社:, 201906. 265.
- [14] Preston Zhang. Practical Guide to Large Database Migration[M]. CRC Press:2019. 04. 15.
- [15] 贾轩, 王栋轩. 基于 SSM 框架下用户注册登录界面的设计与实现[J]. 信息系统工程, 2019(02):54.