



攀枝花学院本科毕业设计（论文）

《算法设计与分析》精品课程教学网站 的设计与实现

学生姓名： 程淼

学生学号： 201410804009

院（系）： 数学与计算机学院

年级专业： 2014 级软件工程

指导教师： 银星 副教授

助理指导教师：

二〇一八年六月

摘 要

传统的教学模式在信息共享和交互方面存在很多缺陷。目前，随着互联网的普及和远程教育的发展，网络学习的学习方式已被越来越多的学习者所接受，并成为教育改革的重要内容之一。在线教学平台不仅解决了信息共享问题，而且增强了师生互动，成为教学实践的辅助平台。在线学习系统个性化学习服务直接影响学习者的学习动机和主动性。

本文介绍了基于 Java 的《算法设计与分析》精品课程教学网站的设计与实现，其中主要介绍了开发的过程、开发工具和框架的选择、功能的设计、数据的表示、开发过程中遇到的问题以及问题的解决方案。根据前期需求分析和课题要求两方面考虑，本系统主要功能有课件信息的管理、视频点播、作业管理、在线测试、后台管理功能等。系统的实现采用 Java 语言，后台框架选用 SpringMVC 和 SpringBoot，数据库采用 MySQL，操作数据库的框架选用常用的对象关系映射框架即 Hibernate。

关键字 课程教学网站，JAVA，MYSQL，设计与实现

ABSTRACT

The traditional teaching model has many defects in information sharing and interaction. At present, with the popularization of the Internet and the development of distance education, learning methods of online learning have been accepted by more and more learners and become one of the important contents of educational reform. The online teaching platform not only solves the problem of information sharing, but also enhances teacher-student interaction and becomes an auxiliary platform for teaching practice. Online Learning Systems Personalized learning services directly affect learners' motivation and motivation.

This article introduced the design and implementation of the excellent course teaching website based on Java's "Algorithm Design and Analysis", which mainly introduced the development process, the selection of development tools and frameworks, the design of functions, the presentation of data, and the development process. Problems and solutions to problems. According to the analysis of the requirements of the previous period and the requirements of the project, the main functions of the system include the management of courseware information, video on demand, job management, online testing, and back office management functions. The implementation of the system adopts Java language. The background framework selects SpringMVC and SpringBoot. The database adopts MySql. The operation database framework adopts the commonly used object relation mapping framework namely Hibernate.

Keywords Course Teaching Website, Java, MySQL, design&implementation

目 录

摘 要.....	I
ABSTRACT.....	II
1 绪论.....	1
1.1 选题背景及研究意义.....	1
1.1.1 选题背景.....	1
1.1.2 研究意义.....	1
1.2 课题研究的主要内容.....	2
2 相关技术介绍.....	3
2.1 JAVA 开发语言简介.....	3
2.2 Thymeleaf 技术简介.....	3
2.3 Spring 框架简介.....	3
2.4 Spring Boot 框架简介.....	3
2.5 Hibernate ORM 框架简介.....	4
2.6 MySQL 简介.....	4
3 系统分析.....	5
3.1 技术可行性分析.....	5
3.2 需求分析.....	5
3.3 用例分析.....	6
3.3.1 教师用户.....	6
3.3.2 学生用户.....	7
4 系统设计.....	8
4.1 系统架构部署.....	8
4.2 总体功能结构设计.....	8
4.3 数据处理流程设计.....	10
4.3.1 重置密码流程.....	10
4.3.2 文件上传流程.....	10
4.3.3 文件下载流程.....	11
4.4 数据库设计.....	12
4.4.1 数据库逻辑结构设计.....	13
4.4.2 数据库物理设计.....	17
5 系统功能实现.....	19
5.1 数据库的配置.....	19
5.2 页面的呈现.....	19
5.3 注册.....	22
5.4 登录.....	24
5.4.1 正常登录.....	24
5.4.2 找回密码.....	25
5.5 视频.....	26

5.5.1 安装 nginx.....	26
5.5.2 修改 nginx 配置文件.....	27
5.5.2 在页面读取视频流.....	27
5.6 试卷.....	28
5.6.1 组织题库资料.....	29
5.7 课件信息的管理.....	30
5.7.1 上传课件.....	30
5.7.2 下载课件.....	32
5.8 作业模块.....	33
5.8.1 根据用户角色访问不同.....	33
5.8.2 教师学生查看作业的范围不同.....	34
5.9 留言模块.....	35
6 系统测试.....	37
6.1 测试记录.....	37
6.2 测试结论.....	39
结 论.....	40
参 考 文 献.....	41
致 谢.....	42

1 绪论

1.1 选题背景及研究意义

1.1.1 选题背景

随着互联网的迅速普及，网络教育在过去二十年中发展迅速，因此在线教育越来越多地用于现代教育系统。依靠互联网，新闻可以快速轻松地传递。与此类似，教育资源也可以通过互联网快速传达给学生。

20 世纪初以美国为首的经济发达国家开始致力于网络教育，网络教学也得到了发展。迄今为止，已经形成了覆盖美国主要教育机构的在线产业，几乎所有学校都可以开展在线教育。整个系统呈现一个网络结构，无论您想学习什么知识和技能，您都可以在网上找到相关信息和帮助，有专门的基础教育和专业教师培训资源以及高等教育资源。

与国外网络教育的深入研究和快速发展相比，中国网络教育起步较晚。目前大多数学校建立了多媒体教室，教师可以通过课件展示更多细节，教学内容生动活泼，增加了课堂容量，缩短了教学时间，提高了效率。但是，由于信息量巨大，学生难掌握细节，分类笔记的难度也越来越大。学生在课堂上学习的时间有限，针对这个问题，将课堂扩展到不受时间和空间限制的互联网，并建立在线教学环境的解决方案应时而生。

通过互联网，学生可以随时浏览和预习课程，还可以通过留言的方式与老师交流。课程教学网站不但可以完整保存信息，对资料分类实现标准化，还可以提高信息传输效率，为用户提供快捷的信息服务，同时还可以实现工作管理和在线测试。

1.1.2 研究意义

教育需求不断增加，教育供给不足是教育面临的根本矛盾。随着互联网的迅猛发展，线下课程教学为在线教学提供了一个很好的模板，网络成为学生学习的好帮手。设计和建立一个良好的课程教学网站，可以促进课程建设的发展，搭建一个网上教学的良好平台，促进高校，社会，师生，学生之间分享教育资源；目前多数高等院校都已拥有自己的教学网站系统，使学校的工作效率大幅度提升，对教学资料的管理也更加方便。使用本系统能给讲授《算法设计与分析》课程的教师提供了一个资源存放和使用的平台，同时也能够给学生提供一

个学习平台，使学生获取资料更加方便。随着技术的更新，未来会有更先进的技术运用在教学网站系统中，促使学习知识的方式升级，适应时代的要求。随着时代的发展，网络教学平台系统将逐步推广到各个学科中，实现学习的多媒体化、网络化、个性化、快速化。

1.2 课题研究的主要内容

计算机的普及极大地改变了人们的生活。目前，计算机信息技术在各行各业和各个领域得到了广泛的应用，因此需要开发各种应用软件。为了以最低的成本，最快的速度 and 最好的质量开发适合各种应用需求的软件，必须遵循软件工程的原则。设计高效的程序不仅需要编程技巧，还需要合理的数据组织和清晰高效的算法。这是计算机科学中数据结构和算法设计的主要内容。对于计算机科学来说，算法的概念是至关重要的。例如，在一个大型软件系统的开发中，设计出有效的算法将起决定性的作用。

本系统是结合当前的社会环境以及目前大众的生活习惯与节奏，依赖 Java 设计的《算法设计与分析》精品课程教学网站。该系统符合传统的 Windows 操作习惯，使用起来基本无任何困难，服务对象为学生和老师。从用户的角度来看，它可以分为两类：学生，教师。本次研究的内容包括了：课件信息管理，视频点播模块，作业管理模块，在线测试模块，系统管理模块等一系列功能。

本文介绍了基于 Java 的《算法设计与分析》精品课程教学网站的开发过程，开发工具和框架的选择，功能的设计，数据的表示，开发过程中遇到的问题以及问题的解决方案。

2 相关技术介绍

2.1 JAVA 开发语言简介

Java 是一种面向对象的编程语言。它不仅吸收了 C++ 语言的各种优点，而且放弃了 C++ 难以理解的多重继承和指针等概念。因此，Java 语言功能强大且易于使用。Java 语言作为静态面向对象编程语言的代表，是面向对象理论的优秀实现，允许程序员以优雅的思维方式进行复杂的编程。

Java 语言是一种快速发展的计算机语言程序。它深刻地展示了程序编程的本质，其简洁严谨的结构和简洁的语法写作为今后的开发和维护提供了保证。由于提供网络应用程序支持和多媒体访问将促进互联网在企业网络中的应用开发。

2.2 Thymeleaf 技术简介

Thymeleaf 是一种 Java XML / XHTML / HTML5 模板引擎，可以在 Web（基于 Servlet 的）和非 Web 环境中使用。它更适合在基于 MVC 的 Web 应用程序的视图层提供 XHTML / HTML5，即使在脱机环境中，它也可以处理任何 XML 文件。它提供了完整的 Spring Framework 集成。

Thymeleaf 旨在成为 JSP 的完全替代品，并实现了自然模板的概念：模板文件可以直接在浏览器中打开，并且仍然可以正确显示为网页。

Thymeleaf 是开源软件，根据 Apache 许可证 2.0 获得许可。

2.3 Spring 框架简介

Spring 框架是 Java 平台的应用程序框架和 Java 的控制容器的反转。任何 Java 应用程序都可以使用该框架的核心功能，但是在 Java EE（企业版）平台之上还可以扩展构建 Web 应用程序。尽管框架没有强加任何特定的编程模型，但它作为 Enterprise JavaBeans（EJB）模型的补充甚至取代，已经在 Java 社区中流行起来。Spring 框架是开源的。

2.4 Spring Boot 框架简介

Spring Boot 是 Spring 的常规配置解决方案，用于创建独立的，生产级的基

于 Spring 的应用程序。它预先配置了 Spring 对 Spring 平台和第三方库的最佳配置和使用“opinionated view”，因此您可以从最小的担心开始。大多数 Spring Boot 应用程序只需要很少的 Spring 配置。它有以下特点：创建独立的 Spring 应用程序；接嵌入 Tomcat 或 Jetty（无需部署 WAR 文件）；提供自己的'starter'项目对象模型（POM）来简化您的 Maven 配置；尽可能自动配置 Spring；提供生产就绪功能，如 metrics，运行状况检查和外部配置；绝对不会生成代码，并且不需要 XML 配置。

2.5 Hibernate ORM 框架简介

Hibernate ORM（简称 Hibernate）是一种 Java 编程语言的对象关系映射工具。它提供了一个将面向对象的域模型映射到关系数据库的框架。Hibernate 通过用直接的，持久的数据库访问和高级对象处理函数来处理对象关系阻抗不匹配问题。

Hibernate 是免费软件，它是在 GNU Lesser General Public License 2.1 下发布的。

Hibernate 的主要功能是从 Java 类到数据库表的映射，以及从 Java 数据类型到 SQL 数据类型的映射。Hibernate 还提供数据查询和检索功能。它生成 SQL 调用并使开发人员无需执行结果集的手动处理和对象转换。

2.6 MySQL 简介

MySQL 是一个开源的关系数据库管理系统（RDBMS）。它的名称是“My”（联合创始人 Michael Widenius 的女儿的名字）和“SQL”（结构化查询语言的缩写）的组合。MySQL 开发项目已经根据 GNU 通用公共许可证的条款以及各种专有协议提供了其源代码。MySQL 由一家单一的盈利公司拥有和赞助，瑞典公司 MySQL AB 现由 Oracle 公司拥有。为了专有使用，可以使用多种付费版本，并提供附加功能。

MySQL 是 LAMP 开源 Web 应用程序软件堆栈的核心组件。LAMP 是“Linux, Apache, MySQL, Perl / PHP / Python”的首字母缩写。使用 MySQL 数据库的应用程序包括：TYPO3, MODx, Joomla, WordPress, Simple Machines Forum, phpBB, MyBB 和 Drupal。MySQL 也用于许多知名的大型网站，包括谷歌（尽管不适用于搜索），Facebook, Twitter, Flickr 和 YouTube。

3 系统分析

3.1 技术可行性分析

本系统使用 java 高级编程语言，此语言简单易学，是一个可移植性强、面向对象编程的语言。目前 java 编程语言的生态圈发展的相当大，发展已经很成熟了。数据库用的是最流行的开源数据库 MySQL。它可以运行在多个操作系统上，支持多种语言，速度非常快，并且还支持大型数据库。另外，前端页面使用 Thymeleaf 动态网页技术用于实现 Java Web 应用程序的用户界面部分。使用 CSS 修改页面样式的页面，JS 实现前后台交互。JS 是最流行的轻量级脚本语言，易于学习，灵活性特别高而且可跨平台运行，极大的提高了用户体验。

3.2 需求分析

教育部于 2003 年启动了教学质量工程，建设了精品课程，以促进现代信息技术在教学中的应用，分享优质教学资源。经过前期的研究意义，可行性等一系列的分析后，结合选题指南的预期目标，综合考虑下，设计了《算法设计与分析》精品课程教学网站，本系统主要包含了以下功能：

① 用户管理：要求拥有教师用户、学生用户、两种用户角色，每种角色拥有不同的权限集合，在系统种访问的范围也不同。包括注册、登录、身份验证、用户基本信息的增加、修改、删除和查询、修改密码、找回密码、退出系统。用户信息主要包括：编号、账号、密码、姓名、性别、电话、邮箱、注册时间、角色等信息。其中密码只能由英文或者数字组成；预留邮箱用于找回密码或者修改密码。教师负责上传学习资料、组织题库资料、解答学生疑问、注册学生成绩管理、学生作业练习批改管理；学生角色主要功能是学习、参加测试、学习交流（留言、提问及回答）、作业提交。

② 课件信息的管理：首先要能够课件的发布，课件发布需要有设置标题和简介的功能。教师登录到系统中，将课件和代码压缩后进行发布。这样，他所教的班级里的学生都可以看到课件了。其次应该有课件标题搜索功能，学生和老师可以根据课件标题来搜索查看课件。

③ 视频点播模块：客户终端能够在远程或局域网里随意自由地点播各种流媒体格式的视音频节目。该模块的数据库包含各种数据库表：评论信息表，用户信息表，区域信息表。通过这些数据库表，统计数据可以很容易地关联用户，

用户可以更好地查看。

④ 作业管理模块：实现教师布置作业、学生上交作业。在师生互动部分，用户可以根据需要上传作业。作业是由教师用户发布,可以添加附件来具体说明作业的详细信息。作业记录保存后，学生用户可以查看到所有的作业信息，通过搜索精确查找相关作业信息，查看作业详情，下载作业消息附加作业说明文件，可提交作业即上传作业文件。

⑤ 在线测试模块：实现教师出题、学生在线考试。学生可以下载单元测试和测试文件库，也可以在线阅读测试问题。

⑥ 系统管理模块：教师管理留言、用户、作业和公告。

3.3 用例分析

本系统是《算法设计与分析》精品课程教学网站，主要涉及到教师用户、学生用户，每种用户拥有不同的权限集合，这是符合开发需求的。

3.3.1 教师用户

本系统的教师用户为默认的初始化用户，即不用注册，系统发布时就已经初始化了教师用户的信息。不过后续需要添加教师用户时，也可以通过先注册为学生用户，然后通过教师设置权限的方式来添加新的教师用户。教师用户拥有本系统的最高权限，可访问整个系统的所有页面；教师主要的业务是上传学习资料、组织题库资料、解答学生疑问、注册学生成绩管理、学生作业练习批改管理，课件信息进行相关的数据操作与维护。教师用户用例图如图 3.1 所示。

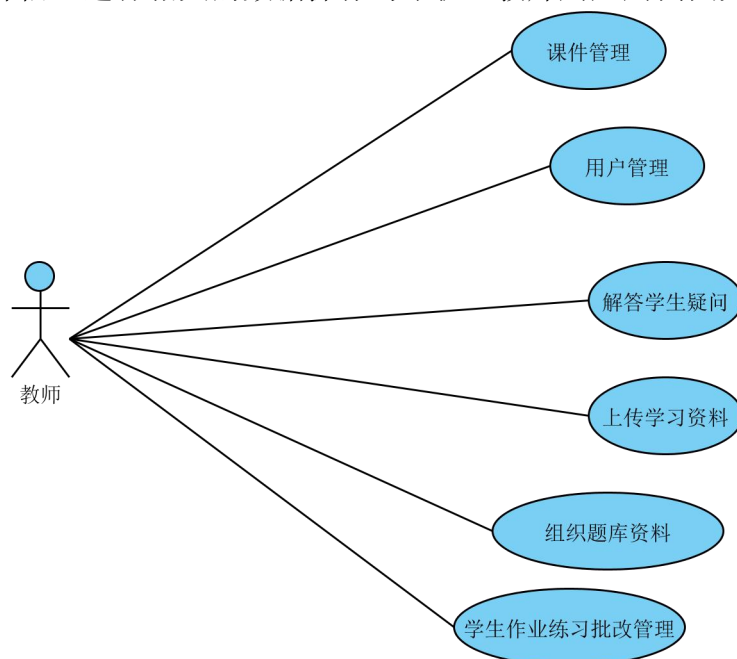


图 3.1 教师用户用例图

3.3.2 学生用户

本系统学生用户是通过在注册页面注册账号而来的，学生用户拥有的浏览本系统的前台的部分界面，学生用户的主要业务是学习、参加测试、学习交流（留言、提问及回答）、作业提交。学生用户用例图如图 3.2 所示。

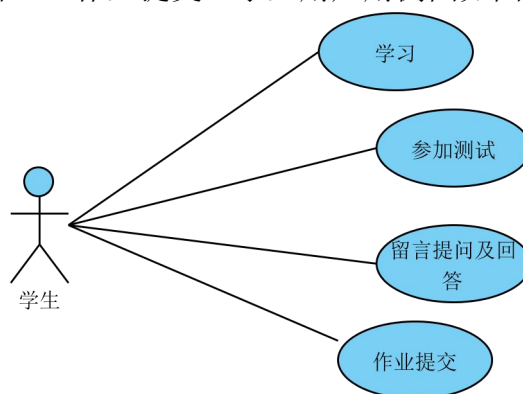


图 3.2 学生用户用例

4 系统设计

4.1 系统架构部署

通过需求分析，该系统计划采用 B/S 结构（浏览器/服务器模式）的三层应用模式。这是当前业内比较广泛应用的网路节后模式，该模型包括三层体系结构：数据表示层，业务逻辑层和数据持久层。系统 UML 部署图如图 4.1 所示：

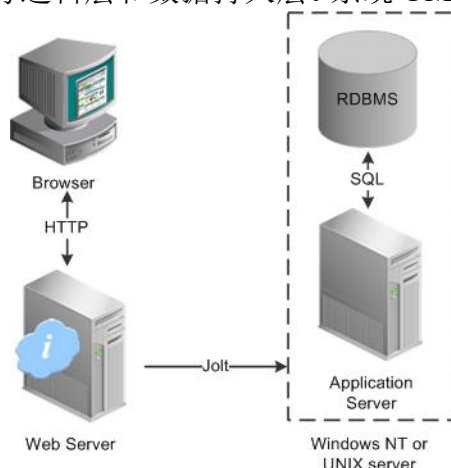


图 4.1 系统部署图

图中 WebBrowser 为数据呈现层，即通常所说的浏览器，用来呈现数据信息以及用户的输入数据信息；ApplicationServer 即为系统的业务逻辑层，业务逻辑层是本系统的主要核心，所有的业务逻辑、对数据的验证、对数据的操作都在这一层；Database 即为本系统的数据持久层，该层的作用是数据持久化，存、取数据都有它来完成。

4.2 总体功能结构设计

本系统只有一个登录界面，教师用户、学生用户都是通过首页访问登录界面，进而进行登录系统。总体结构图如图 4.2 所示：

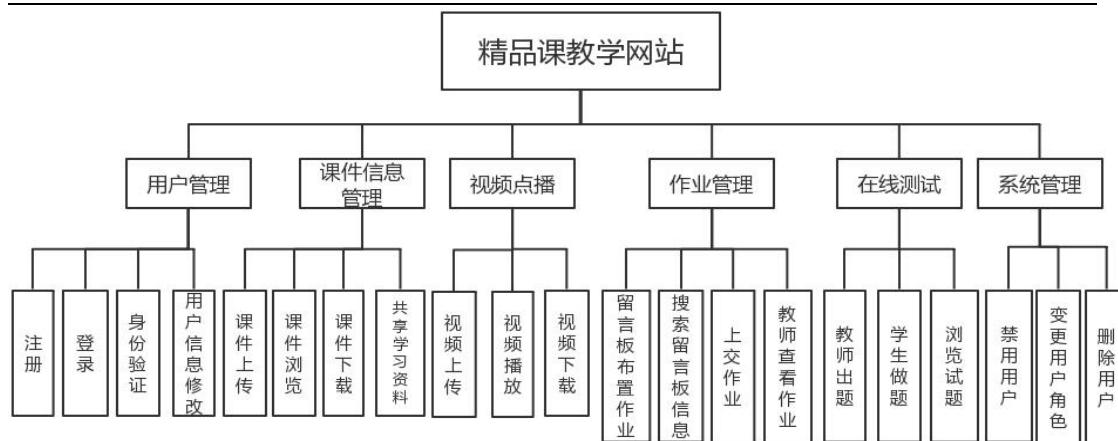


图 4.2 系统功能结构图

注册登录模块：游客只能访问本网站登录页，若想要查看更多信息，则需要先注册为本系统的用户。注册成功登录后方可在本系统上浏览和下载资源。忘记密码的用户可以通过电子邮件重置密码。

课件管理模块：教师上课时使用 PowerPoint 课件，还要在 Sublime 中讲解代码。上完课后许多同学向他索要课件和代码，他感觉这样给每一个同学进行分布太麻烦了。所以他登录到系统中，将课件和代码压缩后进行发布。这样，他所教的班级里的学生都可以看到课件了。学生上课前需要预习，但课本上内容很多，他登录系统后，可以看到课程的课件，打开后，按课件重点预习。课件上传和管理功能：通过 Servlet3 和 Java 的 IO 流链接到上传的课件，来存储到某一个文件夹中来管理添加文件。课件下载，删除功能：右侧的课件标题显示区包含课件下载和删除，以及指向其下载和删除功能的超链接。

视频点播模块：此功能用户能够在远程或局域网里随意自由的点播各种流媒体格式的视音频节目。如果用户想要观看视频并且没有出现在页面上，他可以填写搜索框搜索该视频。如果用户想要查看某种类型的视频，他可以点击系统栏中的视频类别进入类别页面。

作业管理模块：教师用户使用此功能发布作业。学生登录系统后，他可以看到作业的布置。根据作业要求，所需的解题代码在外部编辑器中完成并压缩成一个文件。学生在作业编辑器中输入作业答案，并将压缩文件作为附件提交。提交后，老师可以看到学生的作业。

在线测试模块：教师用户设计录入试题，有单选题，多选题，填空题三种选择。试卷调整修改恰当发布后，学生用户才可以访问自己的主页，在计算机房统一的监督下，打开链接进行测试。学生完成试题提交之后，系统保留学生的答案。教师用户可以浏览系统中学生的答题情况，删除其中一个学生用户的试卷题回答或删除其中一个系统试卷。

系统管理模块：个人信息管理，密码更改，添加，删除，公告信息管理（添加和删除），学生信息管理和课件管理。

用户管理模块：用户管理模块包含了启用或者禁用用户，变更角色，删除角色的三个功能。

4.3 数据处理流程设计

4.3.1 重置密码流程

本系统的重置密码流程为：用户填写邮箱请求重置密码，系统根据请求判断邮箱是否注册，如果是则向邮箱发送重置密码链接，然后用户登录自己的邮箱进行重置密码操作，系统对相关数据进行验证，验证成功保存至数据库。具体流程如图 4.3 所示。

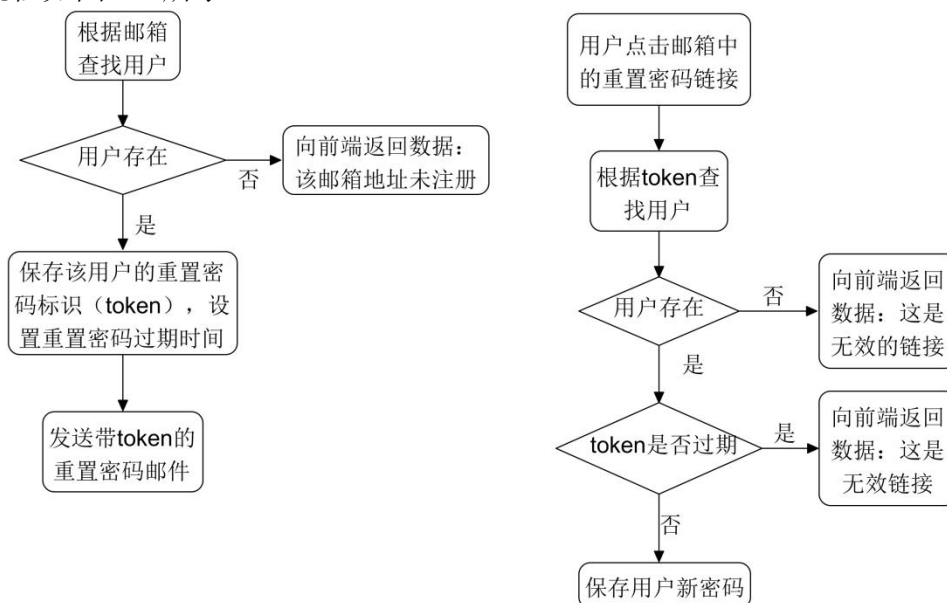


图 4.3 重置密码流程图

4.3.2 文件上传流程

上传文件时，首先让用户选择需要上传的文件，提交后，系统对文件大小进行检验，检验不通过提示相关信息，直到检验通过后才允许上传，然后系统提示上传成功。文件上传流程如图 4.4 所示。

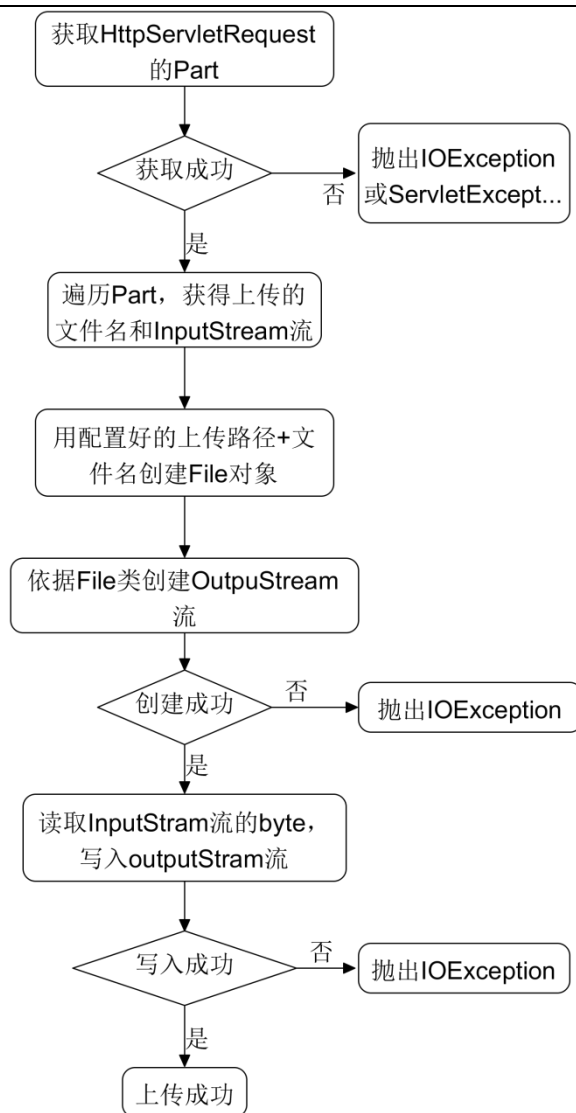


图 4.4 文件上传流程图

文件上传大小限制的配置字段：

最大文件尺寸

spring.servlet.multipart.max-file-size=128000KB

spring.servlet.multipart.max-request-size=1280000KB

4.3.3 文件下载流程

修改数据时，首先从数据库查询相关的记录，用户选择一个需要下载的文件，系统对数据进行有效性检验，检验不通过提示相关信息，重新输入修改信息，直到检验通过。文件下载流程图如图 4.5 所示。

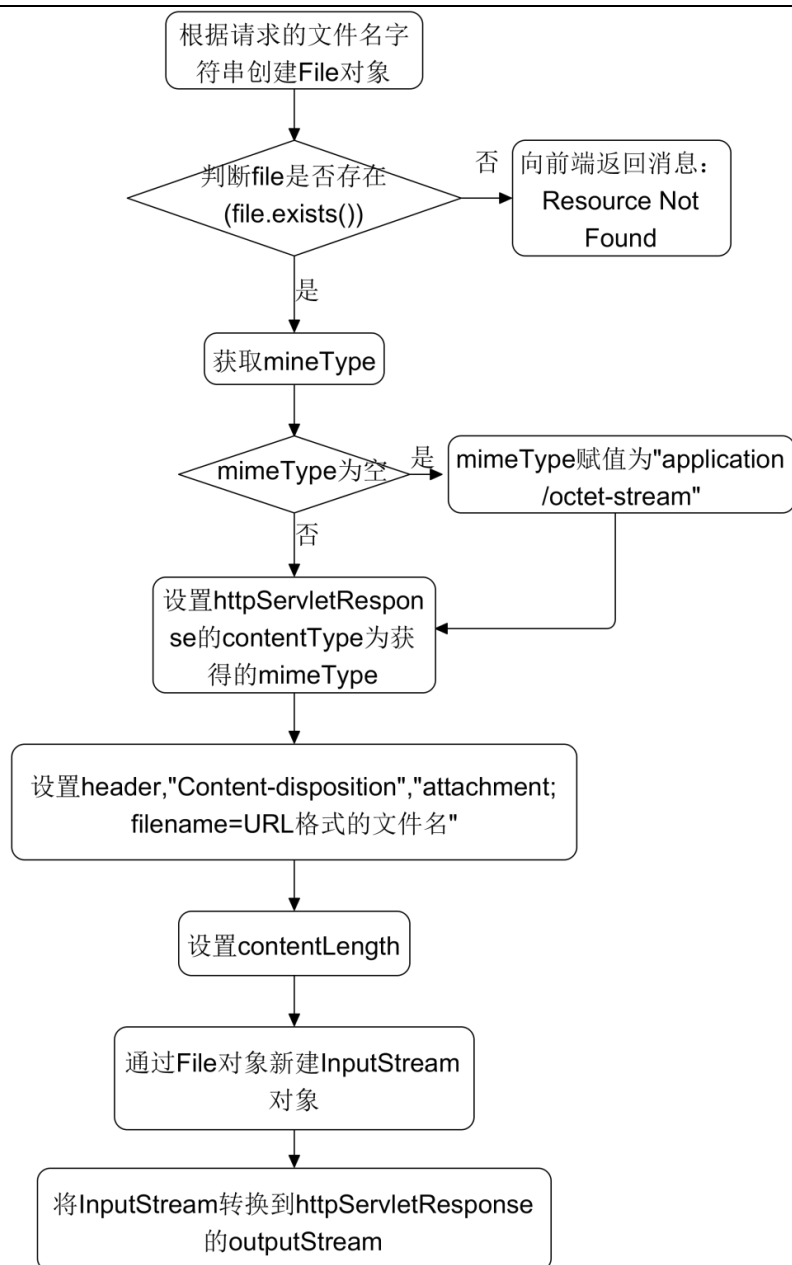


图 4.5 文件下载流程图

4.4 数据库设计

数据库的设计，对于一个系统来说，数据库的重要程度好比人与记忆一样。数据的设计将会直接或间接的影响到系统的运行流畅度，处理数据的效率等。言归正传，本系统采用了当前中小型系统广泛使用的 MySQL 数据库，数据存储编码是 utf-8 编码类型，数据库名为 Pzhu。UTF-8 是万维网最常用的字符编码。每个字符由 1 到 4 个字节表示。UTF-8 向后兼容 ASCII，可以表示任何标准的 Unicode 字符。

4.4.1 数据库逻辑结构设计

数据库逻辑设计的主要作用是将数据库概念设计的 E-R 模型图转换为关系模型，然后在将关系模型转换为数据库逻辑结构。在数据库设计的整个流程中，数据库逻辑设计尤为重要，它将决定系统的整体性能。根据数据库概念设计的 E-R 模型，设计出了以下几个表。

① 角色表（表名：user_role_entity）：角色表用存储角色信息，而本系统的权限是按角色分配权限，所以角色表是必不可少的。角色表的详细结构如表 4.1 所示：

表 4.1 角色表

说明	列名	数据类型	长度	主键	外键	非空
编号	id	char(45)	45	是	否	是
名称	role	varchar(45)	45	否	否	是
用户 ID	user_entity_id	varchar(45)	45	否	否	是
创建时间	created_date	date		否	否	否
最近修改时间	last_modified_date	date		否	否	否

② 用户表（表名：user_entity）：用户表存放用户的基本信息，如姓名，性别，电话，地址等。用户表的详细结构如表 4.2 所示：

表 4.2 用户表

说明	列名	数据类型	长度	主键	外键	非空
编号	id	char(45)	45	是	否	是
账号	name	varchar(45)	45	否	否	是
姓名	display_name	varchar(45)	45	否	否	是
性别	gender	char(1)	1	否	否	是
电话	cellphone	varchar(16)	16	否	否	是
邮箱	email	varchar(50)	50	否	否	是
重置密码标识	confirmation_token	varchar(45)	45	否	否	否
学工号	student_or_e_id	char(45)	45	否	是	是
生日	birthday	date		否	否	否
重置密码过期时间	out_date	date		否	否	否

③ 账户表（表名：account）：用户表不存储密码，账户表存储用户的账

户名和邮箱，以使用户登录时，既可以是注册名称，也可以是邮箱地址。**account**表与 **user_entity** 表是多对一关系。账户表的详细结构如表 4.3 所示：

表 4.3 账户表

说明	列名	数据类型	长度	主键	外键	非空
编号	id	char(45)	45	是	否	是
账户名或者邮箱	account_name	varchar(255)	255	否	否	否
密码	password	varchar(255)	255	否	否	否
是否启用	enabled	tinyint(1)	1	否	否	否
属于的用户	user_entity_id	varchar(45)	45	否	是	否

④ 视频表（表名：**video_entity**）：视频表存储了上传的文件名信息。视频表的详细结构如表 4.4 所示：

表 4.4 视频表

说明	列名	数据类型	长度	主键	外键	非空
编号	id	char(45)	45	是	否	是
视频名称	name	char(255)	255	否	否	否

⑤ 试卷表（表名：**SurveyEntity**）：试卷表用于存放设计的试卷，存储了试卷 ID、试卷完成数量、加入时间、用户 ID 等信息。试卷表的详细结构如表 4.5 所示：

表 4.5 试卷表

说明	列名	数据类型	长度	主键	外键	非空
编号	id	char(36)	36	是	否	是
完成试卷数量	answerNumber	int		否	否	否
创建时间	createTime	datetime		否	否	否
试卷网址	htmlPath	char(36)	36	否	否	否
试卷名字	surveyName	char(36)	36	否	否	否
试卷题目数量	surveyQuestionNum	int		否	否	否
试卷状态	surveyStatus	varchar(45)	45	否	否	否
创建试卷的用户	UserEntity_id	varchar(45)	45	否	是	否

⑥ 题干表（表名：**QuestionEntity**）：题干表用来存放试卷的题干文字，存储了试卷（题干属于的试卷）ID、题号顺序、题干内容、题目类型(多选，单

选，填空)等信息。题干表的详细结构如表 4.6 所示：

表 4.6 题干表

说明	列名	数据类型	长度	主键	外键	非空
编号	id	char(36)	36	是	否	是
试卷 ID	SurveyEntity_id	char(36)	36	否	是	是
题号顺序	orderById	int		否	否	是
题干内容	questionTitle	varchar(255)	255	否	是	是
题目类型	questionType	varchar(45)	45	否	否	是

⑦ 填空题表（表名：BlankEntity）：填空题表用于记录要填入填空题的内容信息，包含了填空的内容、所属于的试卷、所属于的回答者、所属于的题干信息。填空题表的详细结构如表 4.7 所示：

表 4.7 填空题表

说明	列名	数据类型	长度	主键	外键	非空
编号	id	char(36)	36	是	否	是
要填空的内容	answer	varchar(255)	255	否	否	否
所属于的回答者	AnswerEntity_id	varchar(45)	45	否	否	否
所属于的试卷	SurveyEntity_id	varchar(45)	45	否	否	否
所属于的题干	QuestionEntity_id	varchar(45)	45	否	是	否

⑧ 多选题选项表（表名：QCheckboxEntity）：多选题选项表用来存放多选题项，该表存储了选择题项、题项顺序编号、所属于的题干信息。多选题选项表的详细结构如表 4.8 所示：

表 4.8 多选题选项表

说明	列名	数据类型	长度	主键	外键	非空
编号	id	char(36)	36	是	否	是
题项选项内容	optionName	varchar(255)	255	否	是	是
题项顺序	orderById	int		否	否	是
所属于的题干	QuestionEntity_id	varchar(45)	45	否	否	是

⑨ 多选题回答表（表名：CheckBoxEntity）：多选题回答表用于存放多选题的选择信息，包含了回答者信息、所属试卷 ID、所属题干 ID、多选题项 ID（做完多选题的选择结果）信息。多选题回答表的详细结构如表 4.9 所示：

表 4.9 多选题回答表

说明	列名	数据类型	长度	主键	外键	非空
----	----	------	----	----	----	----

编号	id	char(36)	36	是	否	是
做题人信息	AnswerEntity_id	varchar(45)	45	否	否	是
所属的试卷	SurveyEntity_id	varchar(45)	45	否	是	否
所属的题干	QuestionEntity_id	varchar(45)	45	否	是	是
选择结果	QCheckboxEntity_id	varchar(45)	45	否	是	是

⑩ 单选题题项表（表名：QChoiceEntity）：单选题题项表用来记录单选题的题项信息，该表包含了题项内容、题项顺序编号、所属题干(QuestionEntity表的ID)信息。单选题题项表的详细结构如表 4.10 所示：

表 4.10 单选题题项表

说明	列名	数据类型	长度	主键	外键	非空
编号	id	char(36)	36	是	否	是
题目选项	optionName	varchar(45)	45	否	是	是
选项顺序	orderById	int		否	否	否
所属题干	QuestionEntity_id	varchar(45)	45	否	是	是

⑪ 单选题结果表（表名：ChoiceEntity）：单选题结果表用于存放用户单选题选择结果的信息，包含了答题信息（AnswerEntity_ID）ID、试卷、题干、选择的单选题题项信息。单选题结果表的详细结构如表 4.11 所示：

表 4.11 单选题结果表

说明	列名	数据类型	长度	主键	外键	非空
编号	id	char(36)	36	是	否	是
答题信息	AnswerEntity_id	varchar(45)	45	否	是	是
试卷信息	SurveyEntity_id	varchar(45)	45	否	是	是
题干信息	QuestionEntity_id	varchar(45)	45	否	是	否
选择结果项	QChoiceEntity_id	varchar(45)	45	否	是	否

⑫ 试卷回答表（表名：Answer_entity）：试卷回答表用于存放用户完成试卷的记录信息，包含了试卷 ID，用户 ID。试卷回答表的详细结构如表 4.12 所示：

表 4.13 试卷回答表

说明	列名	数据类型	长度	主键	外键	非空
编号	id	char(36)	36	是	否	是
回答者的 ID	user_entity_id	varchar(45)	45	否	是	是

试卷的 ID	SurveyEntity_id	varchar(45)	45	否	是	是
--------	-----------------	-------------	----	---	---	---

⑬ 课件存储记录表（表名：file_entity）：课件存在文件中的名字和其唯一标识符之间的关系，包含了课件 ID、课件名和 UUID 三个字段。课件存储记录表的详细结构如表 4.13 所示：

表 4.13 课件存储记录表

说明	列名	数据类型	长度	主键	外键	非空
课件 ID	id	varchar(45)	45	是	否	是
名称	name	varchar(45)	45	否	否	是

⑭ 留言表（表名：comment_entity）：留言表存储记录表用于存放用户和留言的一对多关系，包含了留言 ID 和用户 ID 两个表示关系的字段。留言存储记录表的详细结构如表 4.14 所示：

表 4.14 留言表

说明	列名	数据类型	长度	主键	外键	非空
留言 ID	id	varchar(36)	36	是	否	是
留言内容	content	LONGTEXT		否	否	是
用户 ID	user_entity_id	varchar(45)	45	否	否	是

⑮ 作业表（表名：homework_entity）：作业表存储记录表用于学生用户和作业的一对多关系。留言存储记录表的详细结构如表 4.15 所示：

表 4.15 作业表

说明	列名	数据类型	长度	主键	外键	非空
作业 ID	id	varchar(45)	45	是	否	是
作业名称	name	varchar(255)	255	否	否	是
用户 ID	user_entity_id	varchar(45)	45	否	否	是

4.4.2 数据库物理设计

数据库的物理设计：设计数据库的物理结构，一本都是根据数据库逻辑设计的结构来选择 RDBMS；结合本系统的数据库概要设计、数据库逻辑设计以及 MySQL 的数据库的存储结构和存储类型，本系统选用的是 MySQL 数据库。我采用的是 Visual Paradigm for UML 10.0 工具来进行数据库的物理设计。本系统的试卷部分物理设计模型图如图 4.7 所示：

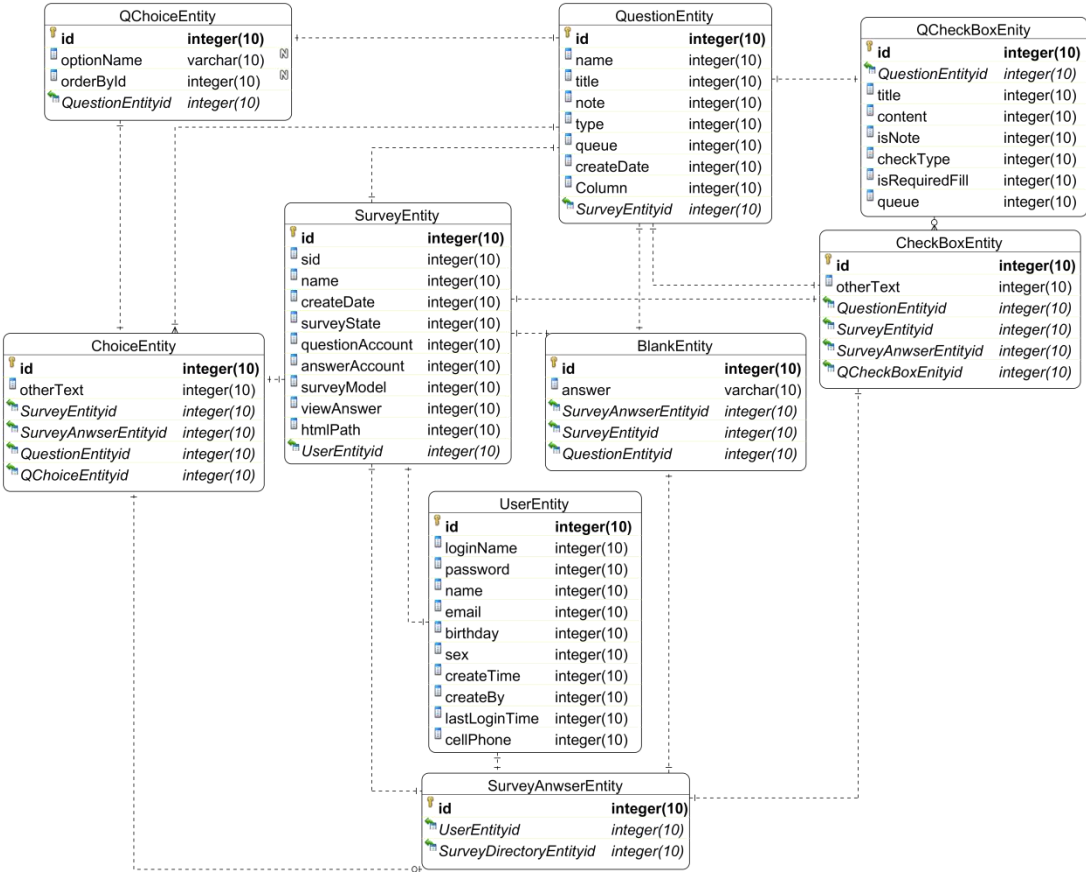


图 4.7 试卷部分数据库物理模型图

5 系统功能实现

5.1 数据库的配置

系统使用的是 Spring Boot Data 框架来管理数据持久层，使用 Spring Boot Data 时 SpringBoot 自动完成需要的 Hibernate 配置，操作数据的 sql 语句由 Hibernate 自动生成。这样便于管理与维护，同时还降低了 Java 代码与 sql 语句的耦合度。Hibernate 还支持对象与数据库的 orm 字段关系映射，这很贴合 Java 的面向对象这一特点。与数据库的连接还用到了 HikariCP(日语中的意思为光)管理连接池。全由 SpringBoot 自动完成相关配置，具体配置如下（application.properties）：

```
# = 数据库连接池，连接名字叫 pzhu 的数据库
spring.datasource.url = jdbc:mysql://localhost:3306/pzhu
# MySQL: 数据库的用户名和密码
spring.datasource.username = root
spring.datasource.password = root
# Keep the connection alive if idle for a long time (needed in production)保持连接数据库
spring.datasource.testWhileIdle = true
spring.datasource.validationQuery = SELECT 1
# = 向控制台显示每次的 sql 查询语句，用于程序开发阶段
spring.jpa.show-sql = true
# Allows Hibernate to generate SQL optimized for a particular DBMS
spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.MySQL5Dialect
```

本系统采用 UUID 作为编号，每新增一条数据，就会生成一个 UUID，每条数据的 UUID 都不相同，这样保证了数据的唯一性；其次，大部分数据的新增都添加了时间的记录，即新增数据时插入当前时间，然后保存至数据库，再提示新增信息成功。

5.2 页面的呈现

页面的呈现运用了 Thymeleaf 的 include 技术，在该技术的支持下，本系统总体设计为四个部分：顶部的导航栏、中部左边的主要内容显示区和中部右边的简要信息提示区以及底部的相关信息区。这样设计不但代码的复用性得到了

提高和重复率的降低，而且更易于维护。

顶部的导航栏属于公共部分，包含了导航按钮、登录和注册等信息，因此，几乎每个页面都一样，只稍需做样式变化即可。其展示图如图 5.1 所示：



图 5.1 顶部导航栏展示图

主要内容显示区没什么说的，单用户点击导航按时，该区域就是显示相应的内容，每个页面内容都不形同，故此区域位当前页面原生内容，未使用 include 进行包含。主要内容显示区如图 5.2 所示：

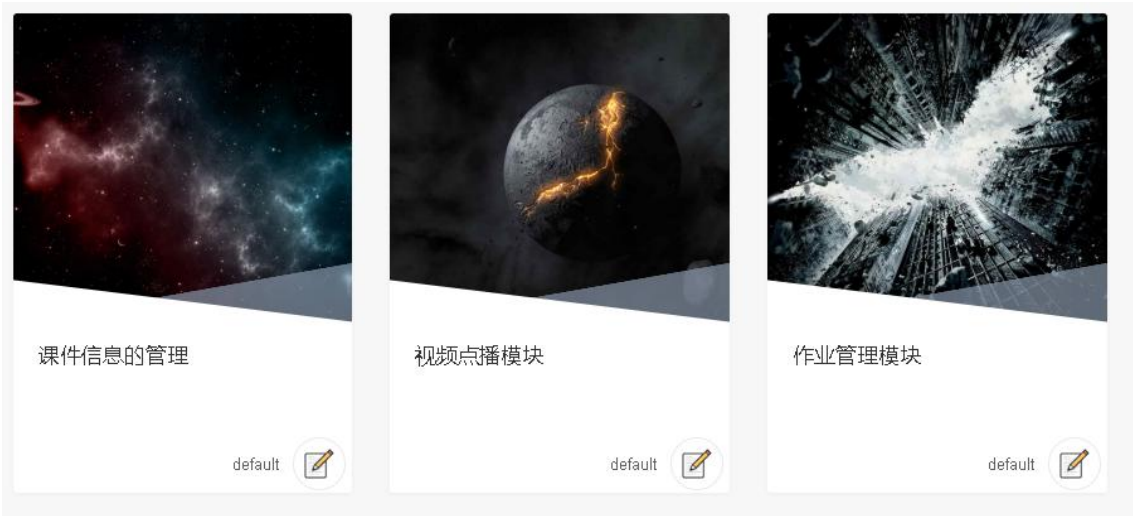


图 5.2 主要内容显示区

部分相关代码如下所示：

```
<!-- th:replace 会替换掉相应的区域 -->
<div th:replace="themes/default/header::header(null,null)"></div>
<!-- 以下是叫 header.html 的文件的部分代码，是页面的通用头部 -->
<div class="navbar-menu">
    <a href="/archives">归档</a>
    <a href="links">友链</a>
    <a href="about">关于</a>
</div>
<div class="navbar-search" onclick="">
    <span class="icon-search"></span>
    <form role="search" onsubmit="return false;">
        <span class="search-box">
```

```

        <input type="text" id="search-inp" class="input" placeholder="搜索..."
maxlength="30" autocomplete="off"/>
    </span>
</form>
</div>

```

无论什么系统都有一个自己的首页，不系统也不列外。本系统首页显示了导航栏、用户可以用的功能模块信息。首页如图 5.3 所示：

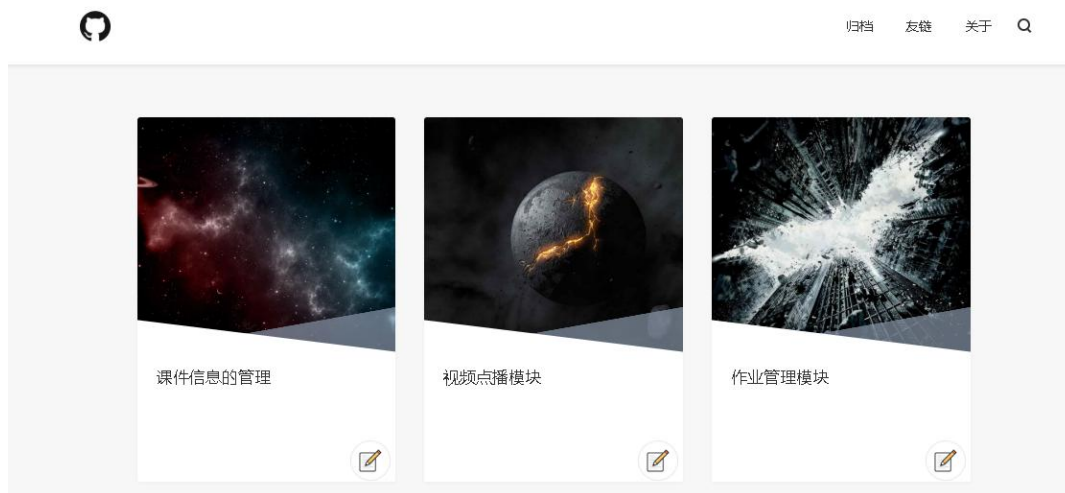


图 5.3 门户首页用户可以用功能

页面中，课件信息管理、视频点播、作业管理模板、在线测试模块的显示运用了 thymeleaf 的 th:each 标签遍历。以下是相关代码：

```

<div class="post-lists">
    <div class="post-lists-body">
        <div class="post-list-item" th:each="app : ${apps.content}">
            <div class="post-list-item-container">
                <a target="view_window" href="#" class="item-thumb bg-deepgrey"
style="background-image:url(/user/img/rand/1.jpg);"
th:href="${app.appLink}"
th:style="background-image:url(/user/img/rand/' + ${app.appImage} +
'.jpg);"'>
                    <div class="item-desc" th:text="${app.appName}"></div>
                </a>
                <div class="item-slant reverse-slant &lt; bg-deepgrey"></div>
                <div class="item-slant"></div>
                <div class="item-label">

```

```

<div class="item-title"><a href="#" th:href="${app.appLink}"
th:text="${app.appName}"></a>
</div>

<div class="item-meta clearfix">
    <div class="item-meta-ico bg-ico-note"
        th:style="background:
url('+@{/user/img/bg-ico.png}+');no-repeat;background-size: 40px auto;'
"></div></div></div></div></div></div>

```

在浏览器输入相应的 url 会向服务器发出请求，服务器收到请求后，分页查询数据库，得到可以使用的应用，将数据填入 Thymeleaf 标注的 html，Thymeleaf 渲染后回馈给前台，前台接收数据后再将其展现出来，若数据库未查到数据，则没有 App 显示。

5.3 注册

凡需要在本网站使用资源的用户，都必须在本网站注册。注册时，用户的输入信息有进行有效性验证，用户需按照提示输入规范的信息方可注册成功。注册成功登录后方可进行相关业务操作。注册页面展示图 5.4 所示：



The image shows a registration form with a dark header containing a GitHub logo. Below the header, the text '快速注册新用户' (Quickly register new users) is centered. The form has two input fields: '登录邮箱' (Login email) with a placeholder '输入邮箱地址' (Enter email address) and an envelope icon, and '登录名称' (Login name) with a placeholder '我们该如何称呼您?' (How should we address you?) and a person icon. A prominent green button labeled '创建账户' (Create account) is below these fields. At the bottom, there is a link '已经有账户?' (Already have an account?) and a button labeled '登录' (Login).

图 5.4 注册界面

系统对用户的信息会进行数据有效性验证，服务器收到注册请求后会进行数据验证，如果邮箱没有占用，则会创建一个用户，设定为锁定状态，然后向其邮箱发送确认邮件，用户点击确认连接，填入密码后完成注册。部分相关代码如下：

```

@PostMapping("/register")
public ModelAndView processRegistrationForm(ModelAndView modelAndView,
@Valid UserEntity userEntity, BindingResult bindResult, HttpServletRequest
HttpServletRequest) {
    modelAndView.setViewName(VIEW_REGISTER);
    // Lookup userEntity in database by e-mail
    UserEntity userExists = userEntityRepository.findByEmail(userEntity.getEmail());
    AccountEntity accountExists =
accountEntityRepository.findAccountEntitiesByAccountName(userEntity.getEmail());
    if (userExists != null || accountExists != null) {
        modelAndView.addObject("alreadyRegisteredMessage", "哎呀！ 邮箱已存在，
换个邮箱试试或者尝试登录");
        bindResult.reject("email");
    }
    //...部分代码省略
    //...保存用户基本信息但设置成禁用状态，等待用户点击邮箱的连接激活
    //...保存用户账号信息，使用户可以用用户名或者邮箱登录，然后调用发送邮件
的接口，验证注册的邮箱有效
    return modelAndView;
}

```

后台邮件验证采用的是 SpringBoot 的開箱即用组件，在 application.properties 配置文件里写入 SMTP 的服务器，用户和密码之后就可以使用。邮件的发件人是登录 SMTP 服务器的用户邮箱。后台发送邮件相关代码：

```

@Service("emailService")
public class EmailService {
    @Autowired
    private org.springframework.mail.javamail.JavaMailSender javaMailSender;
    @Value("${mail.fromMail.address}")
    private String fromMail;
    @org.springframework.scheduling.annotation.Async
    public void sendEmail(org.springframework.mail.SimpleMailMessage
simpleMailMessage) {
        simpleMailMessage.setFrom(fromMail);
        javaMailSender.send(simpleMailMessage);    }}

```

5.4 登录

5.4.1 正常登录

如果需要浏览资源，下载文件，则需要登录后才能访问。用户输入正确的账号和密码后方可成功登录系统。登录界面如图 5.5 所示：

图 5.5 登录界面

如果用户没有登录，系统会自动跳转到登录界面。使用 Spring Security 配置需要保护的 URL，Spring 框架会验证用户的登录状态。登录功能部分代码：

```
@Configuration
@EnableGlobalAuthentication
public class JdbcSecurityConfiguration extends GlobalAuthenticationConfigurerAdapter {
    //...省略部分代码，修改部分 Spring Security 默认配置

    @Bean
    public UserDetailsService userDetailsService(JdbcTemplate jdbcTemplate) {
        RowMapper<User> userRowMapper = (ResultSet rs, int i) ->
            new User(
                //...省略部分代码，用于提取 Sql 查询结果的值，并作为参
                //数初始化 Spring Security 准备的 User 类
                AuthorityUtils.createAuthorityList("ROLE_USER",
                    "ROLE_ADMIN"));
        //..省略部分代码，返回从数据库查询的结果
    }
}
```

```

    }
    @Override
    public void init(AuthenticationManagerBuilder authenticationManagerBuilder) throws
Exception {
        authenticationManagerBuilder.userDetailsService(this.userDetailsService);
    }
}

```

5.4.2 找回密码

如果用户忘记密码，可点击忘记密码来找回密码。找回密码界面如图 5.6 所示：

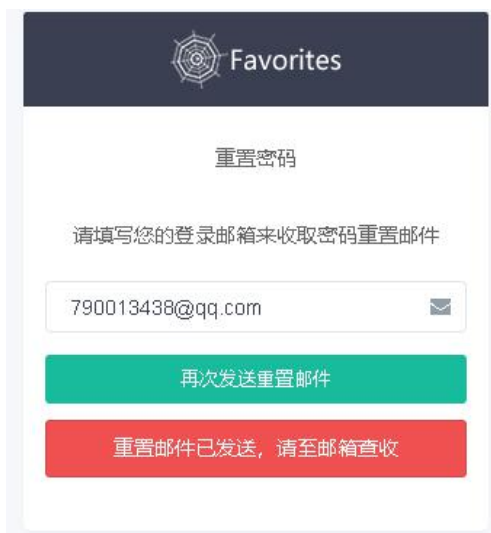


图 5.6 找回密码

用户输入邮箱后，系统就会先检查是否有用户注册相关联的邮箱（如：查询数据库里邮箱地址为 790013438@qq.com 的用户），如果有，然后向用户的邮箱发送邮件确认。用户点击邮箱里的确认连接后，跳转到修改密码界面（如：验证修改密码的 token 是否和用户关联，数据库里是否存在这个 token），密码修改成功显示相关修改成功信息。相关代码如下所示：

```

function sendEmail() {
    var ok = $('#form').parsley().isValid({force: true});
    if(!ok){ return; }
    var url = '/user/sendForgotPasswordEmail';
    $('#sendEmailButton').attr("disabled",true);
    $('#sendEmailButton').text("发送中...");
    $('#errorMsg').hide();
    $.ajax({

```

```
url : url,
data : $("#form").serialize(),
type : 'POST',
dataType : "json",
error : function(XMLHttpRequest, textStatus, errorThrown) {},
success : function(data, textStatus) {
    if(data.rspCode == '000000'){
        $('#sendEmailButton').removeAttr("disabled");
        $('#sendEmailButton').text("再次发送重置邮件");
        $('#errorMsg').html("重置邮件已发送，请至邮箱查收");
        $('#errorMsg').show();
    }else{
        $('#sendEmailButton').removeAttr("disabled");
        $('#sendEmailButton').text("发送重置邮件");
        $('#errorMsg').html(data.rspMsg);
        $('#errorMsg').show();}}
});}
```

5.5 视频

实时消息协议（RTMP,Real-Time Messaging Protocol）最初是由 Macromedia 开发的一项专有协议,用于通过 Internet 在 Flash 播放器和服务器之间传输音频,视频和数据。

5.5.1 安装 nginx

本系统使用编译好的 nginx + rtmp 作为流媒体服务器，支持视频点播。在 <http://nginx-win.ecsds.eu/download/> 下载 nginx 1.7.11.3 Gryphon, 这个版本是唯一在 windows 上编译好的支持视频点播的版本。Linux 操作系统，如 Ubuntu 使用以下命令编译 nginx:

```
sudo apt-get install -y unzip build-essential libpcre3 libpcre3-dev libssl-dev
cd /opt
wget http://nginx.org/download/nginx-1.10.1.tar.gz
wget https://github.com/arut/nginx-rtmp-module/archive/master.zip
tar -zxvf nginx-1.10.1.tar.gz
unzip -o master.zip
cd nginx-1.10.1
```

```
./configure --with-http_ssl_module --add-module=../nginx-rtmp-module-master  
make  
make install
```

5.5.2 修改 nginx 配置文件

修改 nginx 配置文件 conf/nginx.conf, 添加下列内容, 将路径 D:/share 作为视频服务器的存放路径:

```
rtmp {  
    server {  
        listen 1935;  
        chunk_size 4096;  
        application playback {  
            play d:/share;  
        }  
    }  
}
```

5.5.2 在页面读取视频流

使用 video.js 提供的支持播放视频。浏览器需要允许使用 flash:

```
<link href="http://vjs.zencdn.net/5.5.3/video-js.css" rel="stylesheet">  
<script src="http://vjs.zencdn.net/ie8/1.1.1/videojs-ie8.min.js"></script>  
<video id="my-video" class="video-js" controls preload="auto" width="640"  
height="300"  
poster="http://ppt.downhot.com/d/file/p/2014/08/12/9d92575b4962a981bd9af247ef  
142449.jpg" data-setup="{}">
```

```
<source src="rtmp://localhost:1935/playback/1.mp4" type="rtmp/flv">
```

```
<p class="vjs-no-js">播放视频需要启用 JavaScript, 推荐使用支持  
HTML5 的浏览器访问。
```

To view this video please enable JavaScript, and consider upgrading to a web browser that

```
<a href="http://videojs.com/html5-video-support/"  
target="_blank">supports HTML5 video</a>  
</p>
```

```
</video>
```

```
<script src="http://vjs.zencdn.net/5.5.3/video.js"></script>
```


5.6 试卷

试卷使用来进行课后练习或者在线考试。试卷界面如图 5.7 所示：

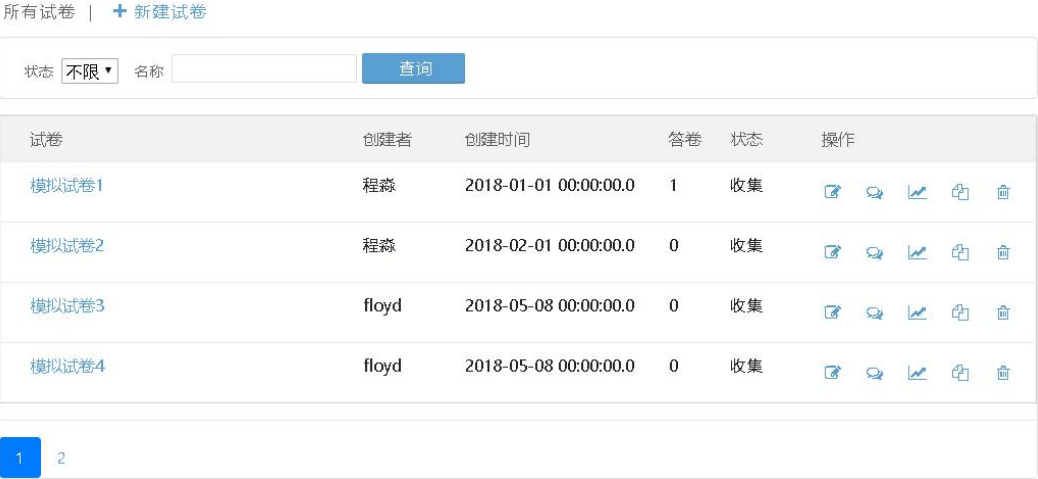


图 5.7 试卷

用户浏览网站时，页面有一个所有试卷的简要统计，显示了试卷的名称，创建者，创建时间，答题数量，方便用户随时查看。用户可点击试卷访问试卷详情页面。简洁试卷如图 5.8 所示：

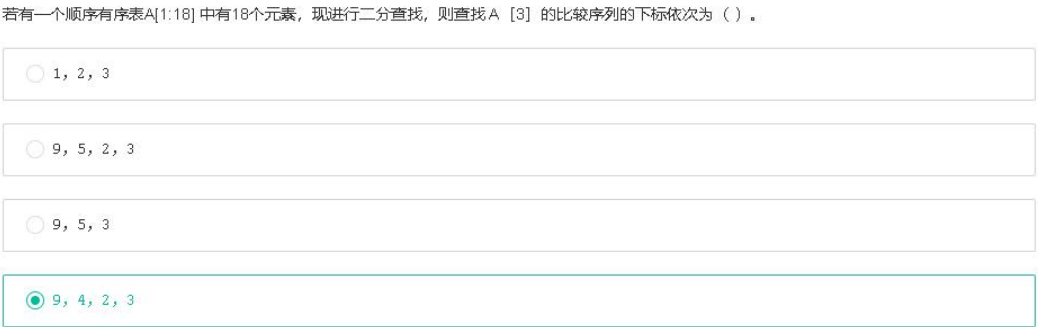


图 5.8 简洁试卷

试卷展示的相关代码如下：

```
public String responseAnswer() throws Exception {
    HttpServletRequest request=Struts2Utils.getRequest();
    String answerId=request.getParameter("answerId");
    List<Question> questions = new ArrayList<Question>();
    try {
        SurveyDirectory directory=new SurveyDirectory();
        if (answerId != null) {
            SurveyAnswer answer = surveyAnswerManager.getModel(answerId);
            questions = surveyAnswerManager.findAnswerDetail(answer);
        }
    }
}
```

```

        User user=accountManager.getCurUser();
        if(answer.getSurveyId() != null && user!=null){
            SurveyDirectory
            survey=directoryManager.getSurveyByUser(answer.getSurveyId(), user.getId());
            if(survey!=null){
                request.setAttribute("questions", questions);
                request.setAttribute("directory", directory);
            }
        }
    } catch (Exception e) { e.printStackTrace(); }
    return RESPONSE_ANSWER;
}

```

5.6.1 组织题库资料

教师可以在试卷设计页面，将题目输入系统，然后点击发布生成试卷。如图 5.9 所示：

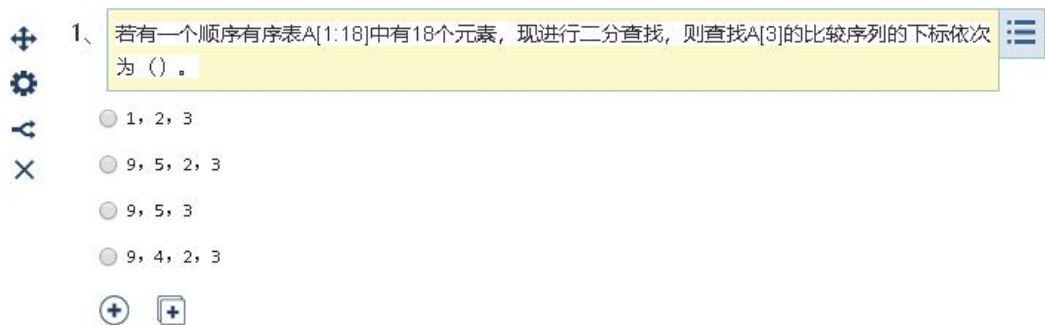


图 5.9 试卷题目输入

相关代码：

```

String url = "/survey!answerSurvey.action?surveyId=" + surveyId;
String filePath = "WEB-INF/wjHtml/" + dateFormat.format(createDate);
String fileName = surveyId + ".html";
new JspToHtml().postJspToHtml(url, filePath, fileName);
survey.setHtmlPath(filePath + fileName);

url = "/survey!answerSurveyMobile.action?surveyId=" + surveyId;
filePath = "WEB-INF/wjHtml/" + dateFormat.format(createDate);
fileName = "m_" + surveyId + ".html";
new JspToHtml().postJspToHtml(url, filePath, fileName);

```

```
survey.setSurveyState(1);
surveyDirectoryManager.save(survey);
```

5.7 课件信息的管理

教师用户进入课件管理系统后，用户可以大致浏览课件，然后选择下载课件，移动课件或者重命名。左侧是一个树形结构，用户可以大致知道课件的位置。

课件预览查看效果如图 5.10 所示：

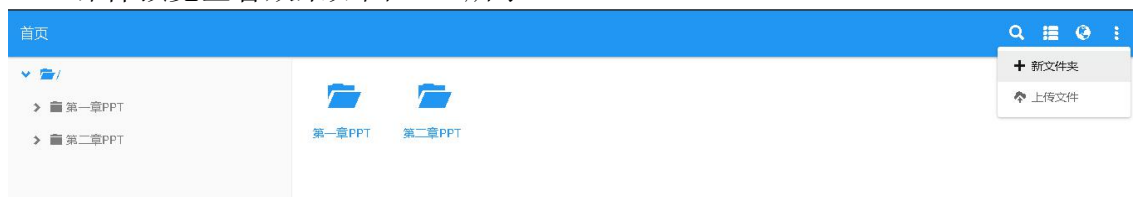


图 5.10 查看界面

部分相关代码：

```
try (DirectoryStream<Path> directoryStream = Files.newDirectoryStream(Paths.get(ROOT,
path))) {
    String DATE_FORMAT = "yyyy-MM-dd HH:mm:ss";
    SimpleDateFormat simpleDateFormat = new SimpleDateFormat(DATE_FORMAT);
    for (Path pathObj : directoryStream) { // 获取文件基本属性
        BasicFileAttributes basicFileAttributes = Files.readAttributes(pathObj,
BasicFileAttributes.class); // 封装返回 JSON 数据
        JSONObject fileItem = new JSONObject();
        fileItem.put("name", pathObj.getFileName().toString());
        fileItem.put("date", simpleDateFormat.format(new
Date(basicFileAttributes.lastModifiedTime().toMillis())));
        fileItem.put("size", basicFileAttributes.size());
        fileItem.put("type", basicFileAttributes.isDirectory() ? "dir" : "file");
        fileItems.add(fileItem);
    } // 省略异常处理的代码
}
```

5.7.1 上传课件

用户可以将相关联的文件放在一个文件夹中，这样进行查找的会更方便，根据文件夹的名字就知道大致会有哪些文件，通过逻辑找到一目了然。

创建文件夹部分代码：

```
// 创建目录
@RequestMapping("createFolder")
public Object createFolder(@RequestBody JSONObject json) {
    try {
        String newPath = json.getString("newPath");
        File newDir = new File(ROOT + newPath);
        if (!newDir.mkdir()) {
            throw new Exception("不能创建目录: " + newPath);
        }
        return success();
    } catch (Exception e) {
        return error(e.getMessage());
    }
}
```

文件主要分为资源文件和作业文件,分别对应资源文件表和作业文件表。教师在发布资源记录和作业记录时,可以上传资源文件。如图 5.11 所示:



图 5.11 上传界面

上传文件部分代码:

```
@RequestMapping("upload")
public Object upload(@RequestParam("destination") String destination,
    HttpServletRequest httpRequest) {
    try {
        // Servlet3.0 方式上传文件
        Collection<Part> parts = httpRequest.getParts();
        for (Part part : parts) {
            if (part.getContentType() != null) { // 忽略路径字段,只处理文件类型
                String path = ROOT + destination;
                File f = new File(path,
                    FileUtils.getFileName(part.getHeader("content-disposition")));
                if (!FileUtils.write(part.getInputStream(), f)) {
                    throw new Exception("文件上传失败");
                }
            }
        }
    }
    return success();
}
```

```

    } // 省略 catch 的异常处理代码
}

```

5.7.2 下载课件

同学查看学习资源并下载他需要的文件。如图 5.12 所示：

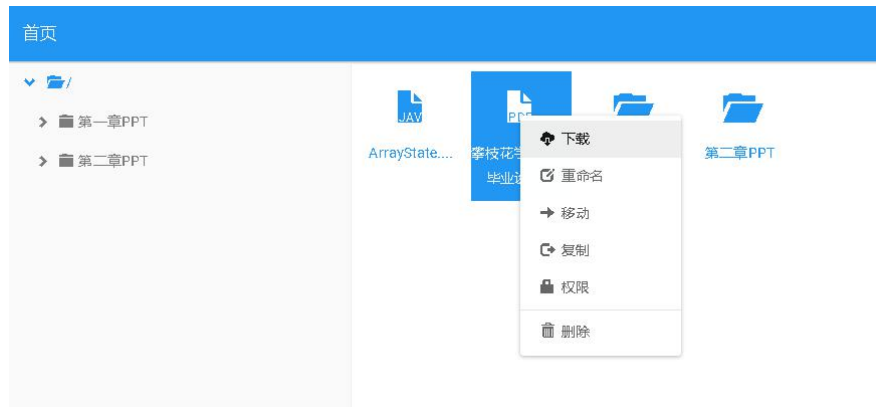


图 5.12 下载界面

下载文件部分代码：

```

File file = new File(ROOT, path);
if (!file.exists()) {
    response.sendError(HttpServletResponse.SC_NOT_FOUND, "Resource Not
Found");
    return;
}
// 获取 mimeType
String mimeType = URLConnection.guessContentTypeFromName(file.getName());
if (mimeType == null) { mimeType = "application/octet-stream"; }
response.setContentType(mimeType);
response.setHeader("Content-disposition", String.format("attachment;
filename=\"%s\"", URLLEncoder.encode(file.getName(), "UTF-8")));
response.setContentLength((int) file.length());
try (InputStream inputStream = new BufferedInputStream(new FileInputStream(file)))
{
    FileCopyUtils.copy(inputStream, response.getOutputStream());
} // 省略异常处理情况

```

5.8 作业模块

5.8.1 根据用户角色访问不同

系统将用户分为教师用户和学生用户，用户登录时不用选择是教师还是学生，系统根据 `user_role_entity` 表的角色分配，判断要呈现的应用 `app`。

下图是学生登录后的主页，学生的主页与教师不同。如图 5.13 所示：

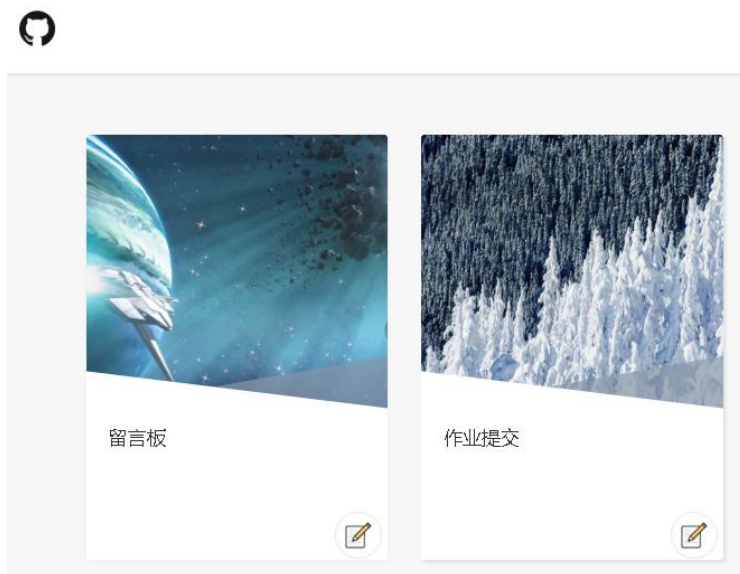


图 5.13 学生主页

下图是教师用户登录后可以看到的功能主页。如图 5.14 所示：

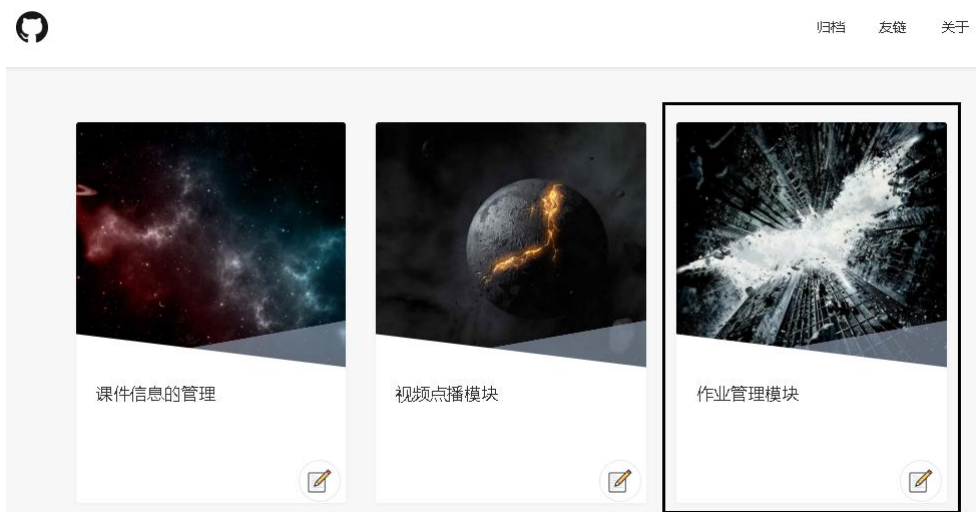


图 5.14 教师可用功能主页

相关判断代码：

```
// 显示用户可用的功能，教师用户和学生用户不一样
modelAndView.setViewName(VIEW_INDEX);
```

```

// 获取当前用户
Authentication authentication =
SecurityContextHolder.getContext().getAuthentication();

String name = authentication.getName();

AccountEntity accountEntity =
accountEntityRepository.findAccountEntitiesByAccountName(name);

modelAndView.addObject("apps",
userEntityRepository.getApp(accountEntity.getUserRole(), PageRequest.of(page, size,
new Sort(Sort.Direction.ASC, "id"))));//这里分页查询，省略向前端返回数据的代码
}

// 查询数据库获取用户的角色
@Formula("(select o.role from user_role_entity o" +
"where o.user_entity_id = user_entity_id)")
public RoleEnum getUserRole() {
return userRole;
}

// 分页查询根据用户角色可以使用的功能
@Query("select o.appName as appName, o.appLink as appLink, o.appImage as appImage"
+ "from UserAppEntity o where o.role = ?1")
Page<AppView> getApp(RoleEnum role, Pageable pageable);

```

5.8.2 教师学生查看作业的范围不同

教师可以查看所有学生的作业，学生只能看到自己的上交作业。下图是教师可以看到的所有学生的作业文件夹。如图 5.15 所示：



图 5.15 教师能查看所有学生的作业

下图是学生用户可以看到的作业文件夹，只限于自己的个人文件夹，不能查看其他学生的作业。如图 5.16 所示：

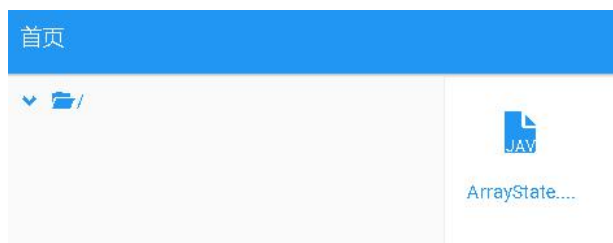


图 5.16 学生查看自己的作业

相关代码：

// FileController.java 的展示该路径下的文件夹功能，教师可以查看所有学生的作业，学生只能查看自己的作业

//...省略部分代码，省略的代码用于改变文件夹的名字\${root/学工号加用户名}

```

        Authentication authentication =
SecurityContextHolder.getContext().getAuthentication();

        String name = authentication.getName();

        AccountEntity accountEntity =
accountEntityRepository.findAccountEntitiesByAccountName(name);

        if (RoleEnum.TEACHER == accountEntity.getUserRole()) {
            dirFile = new File(ROOT + File.separator + accountEntity.getUserName());
            if (!dirFile.exists()) {
                logger.info("创建用户（教师文件夹，管理课件）>" +
dirFile.mkdirs());

                logger.info("创建用户（教师文件夹，管理课件）路径>" +
dirFile.getAbsolutePath());
            }
        } else {
            dirFile = new File(ROOT);
        }

        //...省略部分代码，该向前端返回要显示的数据
    }

```

上传下载作业的功能和上传下载课件类似，调用同一个 **Service** 层的方法，这里不再重复叙述。

5.9 留言模块

将数据从留言表中查出，分页显示，如图 5.17：



图 5.16 留言模块

查询留言。

```
modelAndView.addObject("comments",
commentEntityRepository.findAllOrderByCreateDate(PageRequest.of(page, size,
idASCSort)));
```

保存留言，向数据库中保存用户的 Id，留言内容，留言时间。相关部分代码：

```
@PostMapping(value = "comment")
@ResponseBody
public JSONObject comment(@Valid CommentEntity commentEntity) {      try {
    commentEntityRepository.save(commentEntity);
    JSONObject jsonObject = new JSONObject();
    jsonObject.put("success", true);
    failJSONObject.put("timestamp", System.currentTimeMillis() / 1000);
    return jsonObject;
} catch (Exception e) {
    String msg = "评论发布失败";
    LOGGER.error(msg, e);
    failJSONObject.put("msg", msg);
    failJSONObject.put("timestamp", System.currentTimeMillis() / 1000);
    return failJSONObject;
} }
```

6 系统测试

6.1 测试记录

测试也是所有程序开发不可或缺的环节，因为在整个软件的生命周期中不知道会发生怎样的情况，所以只能尽早的发现错误然后解决错误。一旦程序携带重大问题发布出去，可能会带来严重的经济损失，造成不可挽回的后果。所以本系统也做了测试，测试详情请看下文。本系统在开发的过程中对每个功能都做过相应的单元测试，所以在此环节主要进行黑盒测试。测试用例如表 6.1 所示：

表 6.1 测试用例

编号	测试项	输入	期望结果	输出
1	注册	不输入昵称	注册失败	昵称最少 2 个字符，最多 12 个字符！
		输入正确的用户信息	注册成功	跳转登录页面，提示注册成功！
2	登录	账号：12345678 登录：	登录失败	密码有误，请重新输入
		账号：12345678 登录：12345678	登录成功	跳转网站首页
3	用户管理	点击禁用学生用户	禁用成功	提示框：修改用户状态成功
		教师用户点击审核	审核成功	跳转审核选择页面
3	个人信息	点击基础信息页面保存按钮	基础信息保存成功	提示基础信息保存成功
		点击密码修改保存按钮	密码修改成功	提示密码修改成功
4	资源发布	点击保存	保存成功	弹出保存成功提示框
5	作业发布	点击保存	保存成功	弹出保存成功提示框
6	作业上传	点击选择上传	上传成功	刷新页面，显示上传后的文件夹
7	课件	上传	上传成功	自动刷新页面，显示文件
		下载	下载成功	浏览器提示文件保存路径



图 6.1 注册提示界面



图 6.2 密码错误提示界面

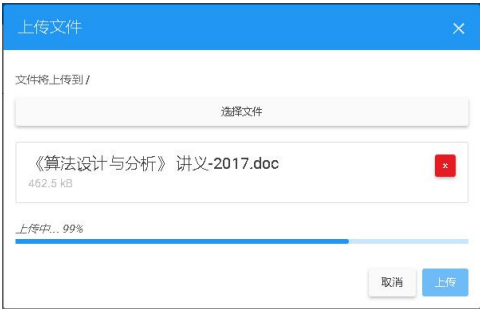


图 6.3 文件上传提示界面

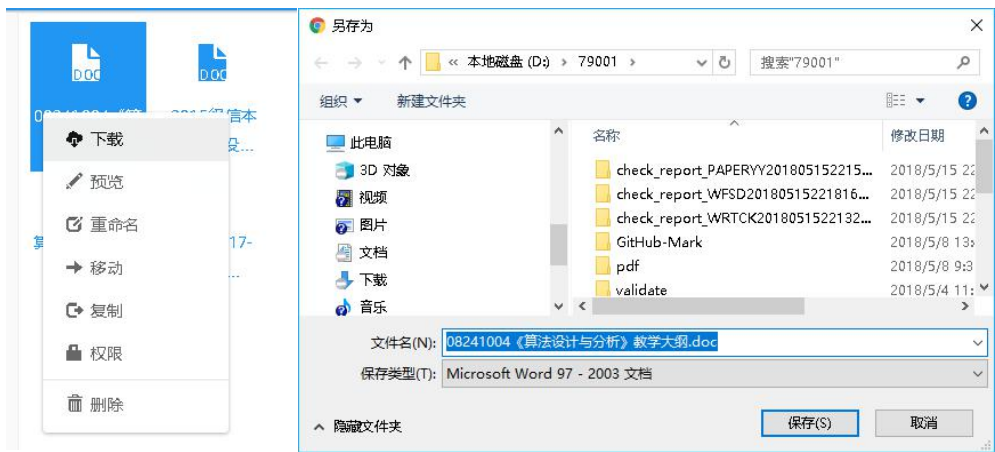


图 6.4 课件下载提示界面

6.2 测试结论

系统测试是校验一个系统是否具有良好的用户体验、完善的可靠性以及安全性的重要环节，故而言之测试即为考验。如果一个系统不能让用户感受到系统带来的便利和快捷。那么这个系统则为失败品。如果我在人工管理上花费的时间比在系统上的时间还少，那么我为什么还会使用这个系统。

所以测试系统实则为发现问题或者使系统使用起来更为方便的解决方案，我们应该尽可能的在测试前列举尽可能多的相关的测试方式和问题。更为全面的测试系统，发现和修改对于大多数用户来说不合理不方便的地方。这便是测试达到的效果。

在此次测试中，我发现并修改了一些问题，但是我知道肯定还有更多的问题并没有被我所发现，这和我没有充分的准备测试方案所导致，这也提醒了我在以后的生活工作中不要为解决问题而解决问题。而是在我们平时日常中，面对问题。先做好对问题的分析和解决方法方法，每一次遇到问题都是对自己的一次考验和测试。我们不能说只要能解决问题就不管其根源，能够解决问题是一种能力，而不断的犯同一个错误这就是自己对问题的应对方式不对。只有从根本上找到了问题的关键才能在以后防范类似的问题出现。

结 论

本次毕业设计完成的主要工作是在对教学平台系统进行需求分析、系统功能结构设计、数据库设计后，采用 Java 语言、Thymeleaf、jQuery、ajax、spring、springmvc、Hibernate 等技术开发了一个教学平台系统，最后撰写论文对整个开发过程中所采用的关键技术、完成的主要工作等进行了论述。

整过开发周期中，先进行了背景及研究意义的调查，然后进行了需求分析、系统设计、数据库设计再到最后的开发并实现。本系统实现了用户的课件管理、作业管理、系统管理、留言板等功能，以及用户的注册、登录、修改密码、修改个人资料等功能。通过不断完善，已经可以正常运行。系统运行稳定，操作简单，维护方便。本系统充分考虑人性化设计，每个模块都尽可能的简化用户和操作人员输入，并有各种的系统提示，以方便用户的使用。

由于本人水平有限，系统中还有很多不尽人意的地方。比如试卷检索比较粗糙，设计时没有过多的考虑数据库查找效率；视频点播的传输流使用 RTMP 协议格式，需要 flash 插件才能播放，没有测试平台兼容性等等。今后有机会尝试优化 SQL 语句，优化 Hibernate 的映射提高查询效率。视频方面可以查实搭建 nginx 服务器进行传输 HLS 协议格式的流媒体文件。本系统设计使用平台在 PC 端，如今大部分用户已习惯使用智能手机移动端，下一版本可以添加兼容移动端的的功能。对用户需求了解的正确与否，直接影响程序设计，系统测试，系统维护的进度和工作量。

参 考 文 献

- [1] 韩瑞丝. 我国体育类国家级精品资源共享课建设及应用状况调查研究[D].首都体育学院,2017.
- [2] 商鹏.UML 课程实践化教学改革方案研究[J].科技视界,2017(17):89-90.
- [3] 张开活. 基于 Web 的 C 语言交互式可视化教学平台的设计与实现[D].西安电子科技大学,2017.
- [4] 韩松.软件工程:面向对象和传统的方法(原书第 8 版)[M].北京:机械工业出版社.2016.
- [5] 张继军,董卫等.Java EE 框架开发技术与案例教程[M].北京:机械工业出版社.2016.
- [6] 宋晏,杨国兴.Java 程序设计及应用开发[M].北京:机械工业出版社.2016.
- [7] 王爱华,汪琼.精品课程与国外开放课程共享利用的对比研究[J].北京:中国远程教育.2010.
- [8] 胥旭,徐恩芹,张桂娟.国内精品课程与 MITOCW 网络教学资源的对比分析[J].北京:中国教育信息化.2012.
- [9] 王艳梅.精品课程的网络资源建设研究与实践[D].上海:上海师范大学.2012.
- [10] 刘乃琦,王冲.《JSP 应用开发与实践》.北京:人民邮电出版社.2016.
- [11] 郑人杰.软件工程概论 第 2 版[M].北京:机械工业出版社.2016.
- [12] Abraham, Silberschatz, Henry, S.Sudarshan .Database Systems Concepts,2015 (8): 11-15
- [13] Bruce Eckel, Bert Bates.Chinese Journal of Computers .2016 (10): 80-83
- [14] Bryan Basham, Bert Bates.Chinese Journal of Computers . 2014 (5): 27-30
- [15] 罗时飞.精通 Spring[M]. 成都:电子工业出版社,2013.
- [16] 王克宏.Java 技术及其应用[M]. 北京:高等教育出版社,2015.
- [17] 郝玉龙.JavaEE 编程技术[M]. 北京:清华大学出版社,2016.
- [18] 张海藩.软件工程导论 第四版. [M]. 北京:清华大学出版社,2015.

致 谢

大学四年生活结束，毕业设计完成之后要进入社会，作为人生中的一个重要节点，在此我要感谢我的父母，把我带到了这个世界，让我学知识，教我学做人，让我体会到世界上最无私的爱，感谢父母把我养育成人。同时也感谢攀枝花学院为我创造宁静致远、优雅先进的学习环境，给了我一个快乐而又难忘的大学生活。感谢这四年教导我们的老师，让我们拥有扎实的专业基础知识。感谢四年来在学习和生活中提供帮助的同学们，让我在大学的学习生活丰富多彩。

整个毕业设计的完成，离不开银星老师的精心指导，不管是初期的需求分析，还是中期系统的设计，又或是最后论文的编写，银老师都耐心的给予我指导和意见。在这个过程中，我的专业技能和文档的编写都得到了一定程度的提高。这也使得我感受到银老师职业精神和责任感。我很庆幸我遇到这样一位导师。在此，我对银星老师表示诚挚的感谢以及真心的祝福。