晋 中 学 院

本科毕业论文（设计）

**题 目** 基于Java在线考试系统

设计与实现

**院 系** 信息技术与工程学院

**专 业** 网络工程

**姓 名** 邓晨阳

**学 号** 1709122108

**学习年限** 2017年9月至2021年7月

**指导教师** 杜涛

**申请学位** 理学 学士学位

2021年4月22日

**基于Java在线考试系统设计与实现**

学生姓名：邓晨阳 指导教师：杜涛

摘 要：随着web技术的进步以及社会对考试这种形式的需求不断变化，传统方式的考试形式不能适应新的需求，必然会得到改进。在线考试已成为了一个十分重要的研究方向。而且今年来，由于新冠疫情的影响，线上开展教育在全国迅速遍及，因此，智能高效的在线考试系统的研发，势在必行。

基于以上需求，本课题采用当前流行的前后端分离架构模式去设计,前端采用Vuejs 技术和Element\_UI组件库配合开发，后端使用 Spring Boot 技术实现，并使用MySQL数据做数据持久化，基于REST风格完成接口设计。本课题设计的在线考试系统将为用户提供一个具有更好适用性以及更加方便的使用感受，同时在一定程度上对于网络考试信息管理的实现具有一定的研究价值。

关键词**：**在线考试；Vue；Spring Boot；MySQL

**Design and implementation of online examination system based on Java**

**Author’s Name: DENG Chen-yang Tutor: DU Tao**

**ABSTRACT：**With the progress of Web technology and social demand for this form of examination constantly changing, the traditional way of examination form can not adapt to the new needs, will be improved. Online examination has become a very important research direction. In recent years, due to the influence of COVID-19, online education has rapidly spread throughout the country. Therefore, it is imperative to develop an intelligent and efficient online examination system.

Based on the above requirements, this topic adopts the current popular front-end separation architecture pattern to design, the front-end adopts VUEJS technology and Element\_UI component library to coordinate the development, the back-end uses Spring Boot technology, and uses MySQL data for data persistence, and completes the interface design based on the REST style. The online examination system designed in this topic will provide users with a better applicability and more convenient use of the experience, at the same time to a certain extent for the realization of network examination information management has a certain research value.

**KEYWORDS:** online examination；Vue；Spring Boot；MySQL

目录

**[1 引言](#_Toc1205972259_WPSOffice_Level1)** **[6](#_Toc1205972259_WPSOffice_Level1)**

[1.1 课题研究背景和意义](#_Toc673219434_WPSOffice_Level2) [6](#_Toc673219434_WPSOffice_Level2)

[1.2 本系统所要实现的基本目标和研究内容](#_Toc1855174842_WPSOffice_Level2) [6](#_Toc1855174842_WPSOffice_Level2)

[1.3 论文组织结构](#_Toc608498701_WPSOffice_Level2) [6](#_Toc608498701_WPSOffice_Level2)

**[2 基本知识和基本原理](#_Toc673219434_WPSOffice_Level1)** **[7](#_Toc673219434_WPSOffice_Level1)**

[2.1 Java语言的简介](#_Toc720540693_WPSOffice_Level2) [7](#_Toc720540693_WPSOffice_Level2)

[2.2 IntelliJ IDEA工具简介](#_Toc467141818_WPSOffice_Level2) [7](#_Toc467141818_WPSOffice_Level2)

[2.3 Vue技术](#_Toc52321694_WPSOffice_Level2) [7](#_Toc52321694_WPSOffice_Level2)

[2.4 Spring Boot技术](#_Toc1049899435_WPSOffice_Level2) [9](#_Toc1049899435_WPSOffice_Level2)

[2.5 MySQL数据库介绍](#_Toc1934160293_WPSOffice_Level2) [9](#_Toc1934160293_WPSOffice_Level2)

**[3 系统分析](#_Toc1855174842_WPSOffice_Level1)** **[9](#_Toc1855174842_WPSOffice_Level1)**

[3.1 可行性分析](#_Toc972079812_WPSOffice_Level2) [9](#_Toc972079812_WPSOffice_Level2)

[3.2 需求分析](#_Toc1837297555_WPSOffice_Level2) [10](#_Toc1837297555_WPSOffice_Level2)

[3.3 非功能性需求](#_Toc792646672_WPSOffice_Level2) [11](#_Toc792646672_WPSOffice_Level2)

**[4 系统详细设计](#_Toc608498701_WPSOffice_Level1)** **[12](#_Toc608498701_WPSOffice_Level1)**

[4.1 设计原则](#_Toc1171553963_WPSOffice_Level2) [12](#_Toc1171553963_WPSOffice_Level2)

[4.2 设计模式](#_Toc29896798_WPSOffice_Level2) [12](#_Toc29896798_WPSOffice_Level2)

[4.3 数据库设计](#_Toc2111794235_WPSOffice_Level2) [13](#_Toc2111794235_WPSOffice_Level2)

**[5 系统实现](#_Toc720540693_WPSOffice_Level1)** **[18](#_Toc720540693_WPSOffice_Level1)**

[5.1 用户登录模块](#_Toc1463473676_WPSOffice_Level2) [18](#_Toc1463473676_WPSOffice_Level2)

[5.2 前台考试子系统](#_Toc1471863441_WPSOffice_Level2) [19](#_Toc1471863441_WPSOffice_Level2)

[5.3 后台管理子系统](#_Toc744723094_WPSOffice_Level2) [23](#_Toc744723094_WPSOffice_Level2)

**[6 系统的测试](#_Toc467141818_WPSOffice_Level1)** **[26](#_Toc467141818_WPSOffice_Level1)**

[6.1 测试方法](#_Toc1026346142_WPSOffice_Level2) [26](#_Toc1026346142_WPSOffice_Level2)

[6.2 黑盒测试用例](#_Toc1210955890_WPSOffice_Level2) [26](#_Toc1210955890_WPSOffice_Level2)

**[7 结束语](#_Toc52321694_WPSOffice_Level1)** **[27](#_Toc52321694_WPSOffice_Level1)**

[7.1 总结](#_Toc917884365_WPSOffice_Level2) [27](#_Toc917884365_WPSOffice_Level2)

[7.2 展望](#_Toc1507486154_WPSOffice_Level2) [28](#_Toc1507486154_WPSOffice_Level2)

**[参考文献](#_Toc1049899435_WPSOffice_Level1)** **[29](#_Toc1049899435_WPSOffice_Level1)**

1 引言

1.1 课题研究背景和意义

随着当今互联网技术的进步，传统线下考试这种形式将面临新的变化，而在线考试就成为人们关切的一个新方向。在线考试系统借助Internet实现不受时间、地域的限制去开展考试，增强考试的灵活性。可以立即按照题库的内容去完成组卷功能。考生可以通过本人的账号和密码登录，将试题标准化存储在服务器中，由系统自动完成判卷，有效确保了试卷的安全性以及考试的公平性。因此，利用互联网的考试必定是将来考试的一个发展新趋向。

1.2 基本目标

本课题设计从功能上有前台系统、后台系统之分：前台供考生考试使用，后台供管理员以及教师用户作信息管理使用。需要赋予用户良好的应用体验,并且尽可能多地提供适合用户的各种相关要求,具体设计项目描述如下：

（1）系统所要实现的基本目标和研究内容统管理员和教师可以使用的功能有：考试管理、题库管理、成绩查询、学生管理、教师管理。

（3）学生可以使用的功能有：管理中心、我的试卷、答题详情页、我的练习、我的分数、给我留言

（4）处理用户需求时高效快捷；如果待响应，会给用户相应提示。

（5）追求良好的视觉设计体验，尽最大努力使界面大方美观。

1.3 论文组织结构

本论文由下面几个部分组成：

第一章 引言，阐述选题意义。

第二章 简要介绍了在系统设计与开发的过程中所需要运用到的各种开发技术、各种开发工具以及整个系统的基础原理

第三章 关于系统的基础分析,首先从其技术上的可行性和经济上的可行性两个方面来深入地分析该系统的应用与发展,并对其需求进行了总体分析。

第四章 系统的详细设计，内容包含有系统的设计原则、总体结构设计、总体功能模块图以及相关数据表的具体字段设计。

第五章 系统实现，内容包含有关于系统中各功能模块实现的详细介绍。

第六章 系统测试 有关于测试系统各功能模块的详细记录。

第七章 结语部分，主要阐述对此次开发设计的经验归纳和对将来的计划展望。

# 2 基本知识和基本原理

2.1 Java语言的简介

Java是一门广受认可好评的编程语言，它也代指一个体系：由一系列软件和开发规范构成。同时，Java具有完美的生态，背后有无数开源社区以及机构提供第三方类库来帮助用户实现各种功能，所以说使用的组件和框架丰富，测试起来相对方便。

2.2 IntelliJ IDEA工具简介

与市场上主流的Java集成开发环境相比，IntellJ IDEA是最受欢迎的IDE，具有更多的智能和便利性。

## 2.3 Vue技术

Vue框架诞生于2014年，其作者为中国人——尤雨溪，也是新人最容易入手的框架之一，不同于React和Angular,其中文文档也便于大家阅读和学习。Vue用于构建交互式的Web界面的库，是一个构建数据驱动的Web界面渐进式框架，该框架遵循CMD规范，并且提供的设计模式为MVVM模式（Model->View->View-Model）和一个可组合的组合型组件系统，具有简单的、灵活的API(接口)。该框架继承了React的虚拟DOM技术和Angular的双向数据绑定技术，是一款较新的功能性框架。

vue且拥有以下特点：

·轻量级

·易上手，学习曲线平稳，文档齐全

·开源，社区活跃度高

·......

同时使用以下vue插件:

(1) vue-cli

vue-cli快速生成vue项目模板。主要的功能有：

·统一的目录结构

·本地调试

·热部署

·单元测试

·集成打包上线

(2) vue-router

vue路由器是vue的官方路由管理器。包含的功能有：

·嵌套的路由/视图表

·模块化的、基于组件的路由配置

·路由参数、查询、通配符

·基于 vue过渡系统的视图过渡效果

·自定义的滚动条行为

(3) vue-cookies

Cookie技术是客户端的解决方案，在vue项目中我们通常要使用到cookie，为此官方为了方便开发人员提供了一个插件Vue- cookies，负责cookie相关的操作。同时vue-cookie这个插件具有易上手以及功能强的特点。

(4) Vuex

vuex是一种状态管理工具。项目中的某些值可能经常被多个组件调用，vue正式提供了专门管理这些值的管理工具-vuex。

(5) Axios

Axios是近年来备受推崇的网络请求库。它负责封装服务器端节点的HTTP请求以及浏览器端的XHR，由于它底层基于promise，所以我们能使用异步方式处理网络请求，这也是ES6所建议的方式。

(6) Element-UI

Element-UI由饿了么官方推出，它的组件库中有非常丰富的组件，可以有效帮助我们快速实现网站样式设计。

## 2.4 Spring Boot技术

Spring Boot是一种从Spring 基础框架4.0衍生而来的框架，它能够以最快的形式构建一个独立的、企业级别的Spring应用，可以以最小的依赖性引入该框架来构建Spring应用程序。并且它还具有以下特点：

（1）嵌入Jetty、Tomcat、Undertow或者Reactor Netty（无需部署war文件）。

（2）使用自动配置，快速构建项目。

（3）提供运行状态检查、运行指标等配置。

（4）没有繁琐而冗余的XML配置。

2.5 MySQL数据库介绍

MySQL是一种关系数据库。因为MySQL一直支持开源以及免费。在许多互联网头部企业中MySQL有很多的应用场景，所以它的稳定性是有保证的。MySQL有着诸多特性：

（1）性能高，很少出现异常宕机。

（2）开源，自主性强、使用成本低。

（3）历史悠久，社区活跃。如果您遇到问题，可以快速获得帮助。

（4）跨平台功能，对多种操作系统的支持以及对许多编程语言的MySQL支持也相当不错。

这些只是MySQL优越性的一部分，正是因为MySQL这些优点，使得它成为最受用户喜爱的数据库之一。

# 3 系统分析

3.1 可行性分析

**（1）技术可行性**

从开发环境上看，环境已经搭建完毕，从开发人员上看，开发者都具有丰富的开发经验和技术储备。因此该系统在技术上讲是完全可行的。

**（2）经济可行性**

由于本次工程主要是作为本科毕业设计的课题,基本上无需任何经济投入。而且这个系统在硬件方面的要求并不是很高,一台电脑也就能够完全满足开发的需要。而且在系统全部部署至云端上线后,还可以为用户带来一笔可观的经济收益,完全能够填补前期研究开发的大量成本费用。所以这个系统是在经济上完全可行的。

3.2 需求分析

1. 系统管理员功能

系统管理员登录并进入后台管理子系统，具有: 题库管理，考试管理，成绩查询，教师管理和学生管理。系统管理员用例图如图3-1所示。

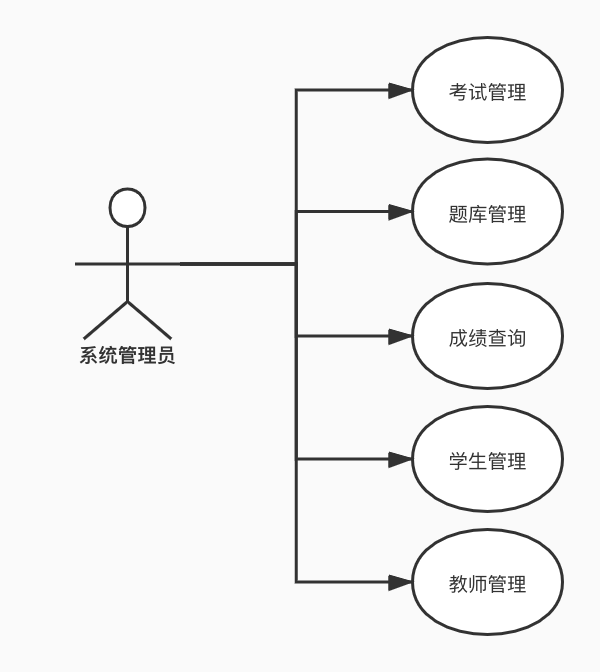


图3-1 管理员用例图

1. 教师功能

教师用户登录后有: 题库管理、考试管理、成绩查询和学生管理。教师用例图如图3-2所示。

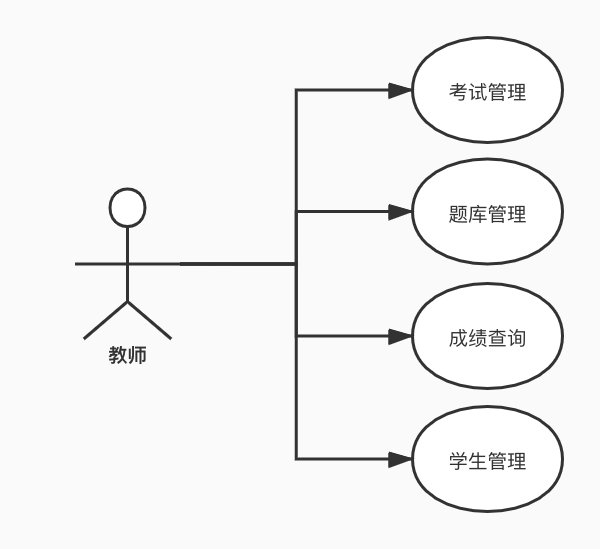


图3-2 教师用例图

1. 学生功能

在该系统中，学生用户登录并进入前台考试子系统，该子系统具有四个功能模块: 我的试卷，我的练习，我的分数和我的信息。学生用例图如图3-2所示。

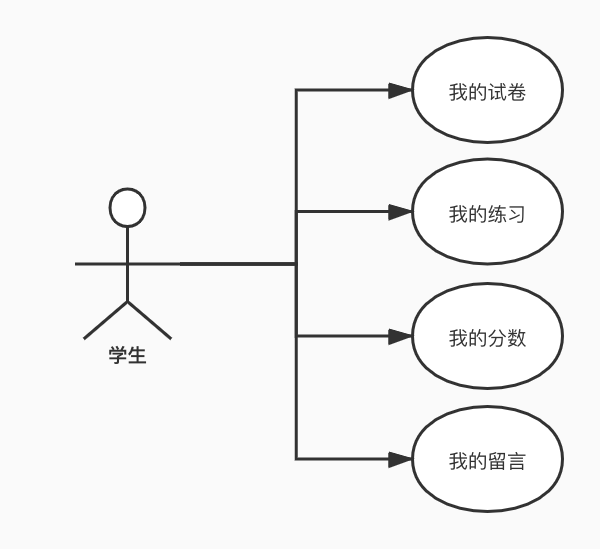


图3-3 学生用例图

3.3 非功能性需求

基于B/S架构通过浏览器（Browser）和服务器（Server）两者建立联系实现的，在登录、管理、考试这些整体功能的界面应当遵循界面简洁、操作简单的原则，避免繁琐问题给用户增加工作量等，具体的要求包括：

(1) 界面需求：界面是进行人机交互的接口，根据大部分的Web应用的操作方式进行UI的布局设计，具备良好的UI界面是吸引用户使用的最直接因素。。

(2) 性能需求：在客户端中使用缓存技术，减少网站的加载时间，优化网站页面。

(3) 操作简单：系统应当尽可能简单，直观并且易于学习。

(4) 多用户考核：系统要尽可能满足考生同时进行在线考试的并发量要求。

(5) 稳定运行: 在评估过程中必须确保应用程序的稳定性，以避免由于崩溃而需要重复检查等故障。

# 4 系统详细设计

4.1 设计原则

（1）设计研究计划，充分利用当前的资源。

（2）遵守软件工程开发的基本原则和方法。

（3）系统的设计应便于数据库的后续维护，然后实现不同用户对数据库请求的处理。

（4）系统的设计中关于数据库设计应满足设计的相关规范。

4.2 设计模式

**4.2.2总体结构设计**

本在线考试系统是基于B/S架构实现的。总体架构如图4-1所示

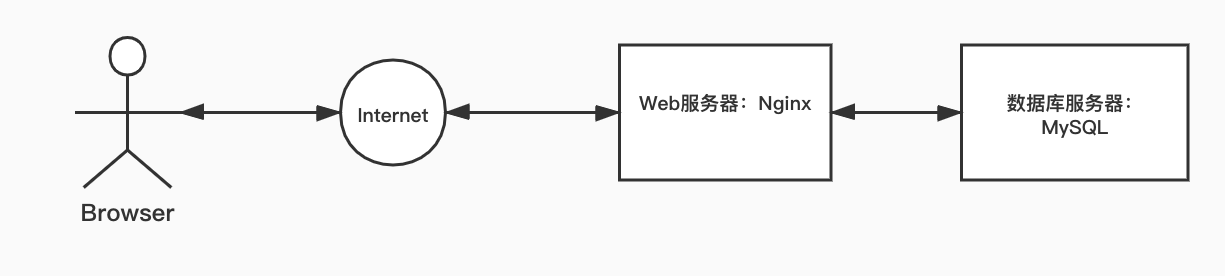


图4-1 系统架构图

**4.2.3总体功能模块图**

系统的总体设计分为三个角色，即管理员、教师和学生。系统管理员用户和教师用户共同有四个功能模块: 题库管理、考试管理、成绩管理和学生管理。系统管理员用户还多具有教师管理功能。学生用户功能模块有: 我的试卷，我的练习，我的分数和我的信息。系统结构图如图4-2所示。

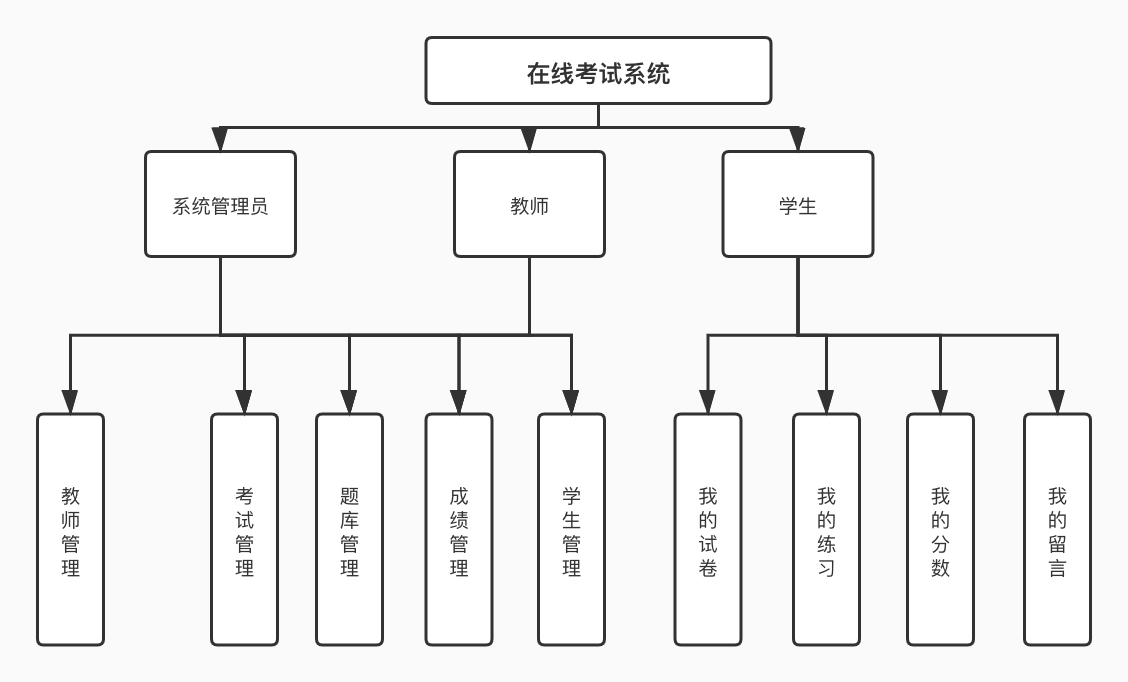


图4-2 系统功能结构图

4.3 数据库设计

**4.3.1数据库表设计**

根据前期的工作分析，为了方便相关数据的持久化，建立以下几张表在MySQL中。

（1）管理员信息表

表4-3 admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 说明 |
| 1 | adminId | int |  | ID号 |
| 2 | adminName | varchar | 20 | 姓名 |
| 3 | sex | varchar | 2 | 性别 |
| 4 | tel | varchar | 11 | 电话号码 |
| 5 | email | varchar | 20 | 电子邮箱 |
| 6 | pwd | varchar | 16 | 密码 |
| 7 | cardId | varchar | 18 | 身份证号码 |
| 8 | role | varchar | 1 | 角色：0管理员1教师2学生 |

（2）考试管理表

表4-4 exam\_manage

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 说明 |
| 1 | examCode | int |  | 考试编号 |
| 2 | description | varchar | 50 | 该次考试介绍 |
| 3 | source | varchar | 20 | 课程名称 |
| 4 | paperId | int |  | 试卷编号 |
| 5 | examDate | varchar | 10 | 考试日期 |
| 6 | totalTime | Int |  | 持续时间 |
| 7 | grade | varchar | 10 | 年级 |
| 8 | term | varchar | 10 | 学期 |
| 9 | major | varchar | 20 | 专业 |
| 10 | institute | varchar | 20 | 学院 |
| 11 | totalScore | int |  | 总分 |
| 12 | type | varchar | 255 | 考试类型 |
| 13 | tips | varchar | 255 | 考生须知 |

（3）填空题题库表

表4-5 fill\_question

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 说明 |
| 1 | questionId | int |  | 试题编号 |
| 2 | subject | varchar | 20 | 考试科目 |
| 3 | question | varchar | 255 | 试卷内容 |
| 4 | answer | varchar | 255 | 正确答案 |
| 5 | analysis | varchar | 255 | 题目解析 |
| 6 | score | int default 2 | 2 | 分数 |
| 7 | level | varchar | 5 | 难度等级 |
| 8 | section | varchar | 20 | 所属章节 |

（4）判断题题库表

表4-6 judge\_question

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 说明 |
| 1 | questionId | int |  | 试题编号 |
| 2 | subject | varchar | 20 | 考试科目 |
| 3 | question | varchar | 255 | 试卷内容 |
| 4 | answer | varchar | 255 | 正确答案 |
| 5 | analysis | varchar | 255 | 题目解析 |
| 6 | score | int default 2 | 2 | 分数 |
| 7 | level | varchar | 1 | 难度等级 |
| 8 | section | varchar | 20 | 所属章节 |

（5）留言表

表4-7 message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 说明 |
| 1 | id | int |  | 留言编号 |
| 2 | title | varchar | 20 | 标题 |
| 3 | content | varchar | 255 | 留言内容 |
| 4 | time | varchar | date | 留言时间 |

（6）选择题题库表

表4-8 multi\_question

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 说明 |
| 1 | questionId | int |  | 试题编号 |
| 2 | subject | varchar | 20 | 考试科目 |
| 3 | question | varchar | 255 | 问题题目 |
| 4 | answerA | varchar | 255 | 选项A |
| 5 | answerB | varchar | 255 | 选项B |
| 6 | answerC | varchar | 255 | 选项C |
| 7 | answerD | varchar | 255 | 选项D |
| 8 | rightAnswer | varchar | 10 | 正确答案 |
| 9 | analysis | varchar | 255 | 题目解析 |
| 10 | score | int default 2 | 2 | 分数 |
| 11 | section | varchar | 20 | 所属章节 |
| 12 | level | varchar | 1 | 难度等级 |

（7）试卷管理表

表4-9 paper\_manage

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 说明 |
| 1 | paperId | int |  | 试卷编号 |
| 2 | questionType | int |  | 题目类型 |
| 3 | questionId | int |  | 题目编号 |

（8）回复表

表4-10 replay

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 说明 |
| 1 | messageId | int |  | 留言编号 |
| 2 | replayId | int |  | 回复编号 |
| 3 | replay | varchar | 255 | 内容 |
| 4 | replayTime | date |  | 回复时间 |

（9）成绩管理表

表4-11 score

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 说明 |
| 1 | scoreId | int |  | 分数编号 |
| 2 | examCode | int |  | 考试编号 |
| 3 | studentId | int |  | 学号 |
| 4 | subject | varchar | 20 | 课程名称 |
| 5 | ptScore | int |  | 平时成绩 |
| 6 | etscore | int |  | 期末成绩 |
| 7 | score | int |  | 总成绩 |
| 8 | answerDate | varchar | 10 | 答题成绩 |

（10）学生信息表

表4-12 student

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 说明 |
| 1 | studentId | int |  | ID |
| 2 | studentName | varchar | 20 | 姓名 |
| 3 | grade | varchar | 4 | 年级 |
| 4 | major | varchar | 20 | 专业 |
| 5 | clazz | varchar | 10 | 班级 |
| 6 | institute | varchar | 30 | 学院 |
| 7 | tel | varchar | 11 | 电话号码 |
| 8 | email | varchar | 30 | 电子邮件 |
| 9 | pwd | varchar | 16 | 密码 |
| 10 | cardId | varchar | 18 | 身份证号码 |
| 11 | sex | varchar | 2 | 性别 |
| 12 | role | varchar | 1 | 角色0管理员1教师2学生 |

（11）教师信息表

表4-13 teacher

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 说明 |
| 1 | teacherId | int |  | ID |
| 2 | teacherName | varchar | 20 | 姓名 |
| 3 | institute | varchar | 20 | 学院 |
| 4 | sex | varchar | 2 | 性别 |
| 5 | tel | varchar | 11 | 电话号码 |
| 6 | email | varchar | 20 | 邮箱 |
| 7 | pwd | varchar | 16 | 密码 |
| 8 | cardId | varchar | 18 | 身份证号码 |
| 9 | type | varchar | 20 | 职称 |
| 10 | role | varchar | 1 | 角色0管理员1教师2学生 |

# 5 系统实现

**5.1 用户登录模块**

在进入系统之前，用户需要通过在后台注册的帐户登录。如果您没有帐户，您可以联系系统管理员或老师在后台管理子系统中注册。

前台使用Axios技术发起异步请求。核心代码:

this.$axios({

url: `/api/login`,

method: 'post',

data: {

...this.formLabelAlign

}

}).then(res=>{

let resData = res.data.data

//根据resData.role 来决定跳转的逻辑

})

后台SpringBoot会在service层使用到JPA这种ORM技术返回给Controller层实体类，最后在Controller层将实体类转换为JSON格式的字符串信息返回给前端数据。核心代码如下：

@PostMapping("/login")

public ApiResult login(@RequestBody Login login) {

// service业务层查询结果

return ApiResultHandler.buildApiResult(‘’,”msg”,res);//封装查询结果

}

**5.2 前台考试子系统**

1. 用户登录

如5.1所述，当学生用户登录后，进入到前台考试子系统中，如图5-1所示：



图5-1 前台考试子系统

可以看到考试前台相关功能：试卷列表、我的练习、我的分数、给我留言。点击右上角的“退出”，即可退出登录

1. 试卷列表模块

刚进入系统或点击“试卷列表”可以查看到当前所有的考试信息，同时支持分页查询和按试卷名称模糊查询考试信息。在点击某个试卷后跳转到“试卷详情页”。如图5-2所示：



图5-2 试卷列表页面

核心代码：

//获取当前所有考试信息

getExamInfo() {

this.$axios(`/api/exams/${this.pageNameber}/${this.pageSize}`).then(res => {

//...

}).catch(error => {

console.log(error)

})

},

1. 答题模块

在我的试卷页面中，点击任意一个试卷跳转到“试卷详情列表”，能够看到考试题型、分值、考生须知、总分等试卷详情。

当我们单击“开始回答”跳转到“回答页面”时，如图5-3所示：



图5-3 考试答题页面

当考试时间结束或考生主动点击结束考试按钮后，跳转到“成绩页面”，如图5-4所示。由于试题答案信息直接存放到数据库中，所以结束考试后可以实现即时判卷、即时出成绩。

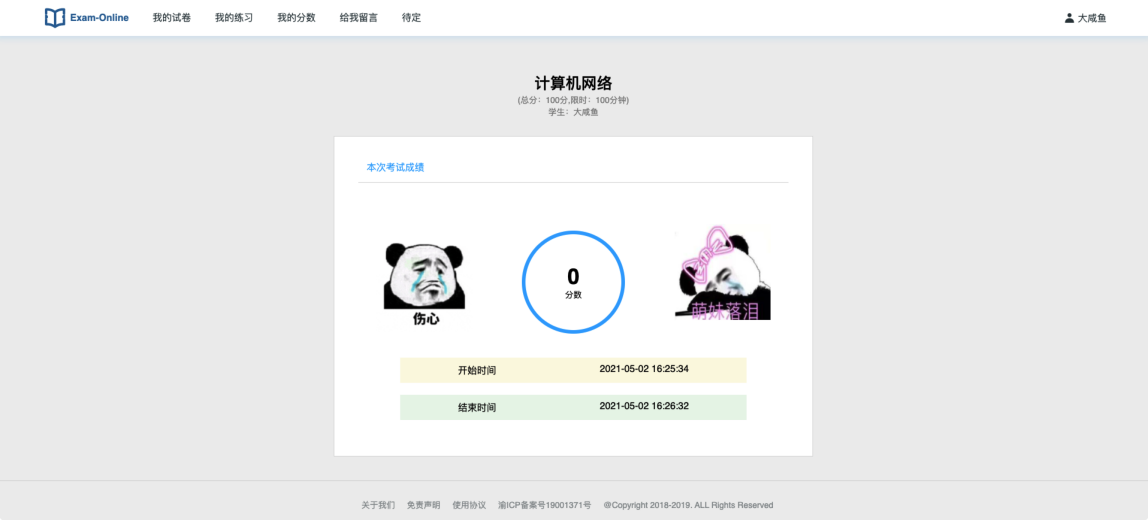


图5-4 考试成绩页面

核心代码：

//答案提交计算分数

commit() {

if(this.time != 0) {

this.$confirm("考试尚有作答时间，是否提前结束","友情提示",{

confirmButtonText: '立即交卷',

cancelButtonText: '再检查一下',

type: 'warning'

}).then(() => {

/\* 交卷 \*/

}).then(res => {

/\* 交卷成功后将成绩写入成绩表 \*/

})

}).catch(() => {

console.log("继续答题")

})

}

},

1. 练习模块

在前台考试子系统中，点击我的练习跳转到“试卷列表页面”。点击任意试卷即可跳转到 “试卷详情列表”，查看本试卷的详细信息。

当点击“开始答题”，跳转到“答题页面”。与考试时的答题有所区别，练习模式答题中能看到正确答案提示、题目解析，如图5-5所示。练习模式交卷后的成绩不会写入到成绩表中。



图5-5 练习答题页面

1. 我的分数模块

点击“我的成绩”，跳转到 “成绩查看页面”，显示当前考生的考试成绩信息: 考试日期、课程名称、考试成绩、是否通过领域。如图5-6所示：



图5-6 分数查看页面

核心代码：

// 获取当前学生的成绩表

getScore() {

let studentId = this.$cookies.get("cid")

this.$axios(`/api/score/${studentId}`).then(res => {

//...

}

})

}

1. 我的留言模块

点击“我的留言”按钮跳转到“留言查看页面”，显示系统中的所有留言信息，用户可以提交留言、回复留言。如图5-7所示：



图5-7留言页面

核心代码：

// 获取所有留言信息

getMsg() {

this.$axios(`/api/messages/${this.pageNumber}/${this.pageSize}`).then(res => {

//...

})

},

**5.3 后台管理子系统**

1. 用户登录

如5.1所述，当教师和管理员用户成功登录后，他们进入后台管理子系统，如图5-8所示:



图5-8 后台考试子系统

相关功能实现有: 题库管理、考试管理、成绩查询、学生管理、教师管理。单击右上角的 “退出” 以注销。

1. 考试管理模块

点击“考试管理”按钮进入到考试管理。在该模块中，系统管理员以及教师可以有查询和添加考试操作。点击考试查询后，跳转到 “考试列表页面”，该页面具有分页功能，还可以编辑和删除指定的考试。在点击添加考试后跳转到“考试添加页面”，在该页面中填写考试相关信息，继续点击“创建考试”按钮保存考试信息。

只有老师发布了考试，学生才能在主页面看到相应的考试信息。有考试安排表以后,才能给该次考试添加题目,对应数据表是exam\_manage。 该表保存该次考试,课程名称,考试时间,所属专业,学院等等信息。

核心代码：

//分页查询所有试卷信息

getExamInfo() {

this.$axios(`/api/exams/${this.pageNumber}/${this.pageSize}`).then(res => {

//...

}).catch(error => {

})

},

1. 题库管理模块

点击“题库管理”按钮进入到题库管理。在该模块中，管理员以及教师用户可以做查询和添加题库等操作。在点击题库查询后跳转到“题库列表页面”，在该页面中以分页的形式显示题目信息。单击添加题库后，跳转到 “试卷列表页面”，其中所有试卷信息均以页面显示，并具有将试题添加到特定试卷的功能。

题库表设计和普通数据表设计有所区别。它分为三个表，即选择题题库表、填空题库表和判断题题库表。每个表保存相应类型的题库，并通过中间表将题库与题库关联起来。这样就组成了一张完整的试卷。

核心代码：

//分页查询所有试卷信息

getAnswerInfo() {

this.$axios(`/api/answers/${this.pageNumber}/${this.pageSize}`

).then(res => {

//...

}).catch(error => {});

},

1. 成绩查询模块

点击“成绩查询”即可进入成绩查询模块。在本模块中，您可以分段查询学生成绩和查询考试成绩。单击 “学生成绩查询” 以分页显示学生用户，然后单击 “查看成绩” 按钮以显示学生用户成绩的折线图。点击成绩分段查询后分页显示考试信息，再点击查看分段，显示出该考试的分数段饼图。

核心代码：

score() {

let studentId = this.$route.query.studentId

this.$axios(`/api/score/${studentId}`).then(res => {

//根据学生Id查询成绩

//...

})

}

1. 学生管理模块

点击“学生管理”，进入学生管理模块。在该模块中，您可以添加、删除和修改学生用户的个人信息，这些信息对应于数据库中的学生表。

核心代码：

//分页查询所有学生信息

getTeacherInfo() {

this.$axios(`/api/students/${this.pageNumber}/${this.pageSize}`).then(res => {

//...

}).catch(error => {});

}

1. 教师管理模块

点击“教师管理”，进入教师管理模块页面。在此模块中可以对教师用户进行增删改差的操作，对应数据库中的teacher表。

核心代码：

//分页查询所有教师信息

getTeacherInfo() {

this.$axios(`/api/teachers/${this.pageNumber}/${this.pageSize}`).then(res => {

//...

}).catch(error => {});

}

# **6 系统的测试**

**6.1 测试方法**

(1) 本系统采用黑盒法进行系统测试。

(2) 黑箱方法是将项目视为黑箱。只是考虑功能能否和用户需求相适应，系统输入、输出数据是否完全正确。

**6.2 黑盒测试用例**

本次系统进行了很多黑盒案例测试，对于这些测试用例，使用了特别严格的测试标准，以保证系统可以最大程度上满足用户的需要。由于篇幅限制，在此仅展示部分测试用例。

（1）系统登录模块的测试如表6-1所示

表6-1 用户登录操作测试

|  |  |  |  |
| --- | --- | --- | --- |
| 测试步骤 | 操作或数据输入 | 显示或数据输出 | 实际输出结果 |
| T-1 | 账号为空，点击“确定” | 显示“登录名不能为空”警告框 | 显示“登录名不能为空”警告框 |
| T-2 | 输入“账号”，并输入错误的“密码” | 显示“密码错误”警告框 | 显示“密码错误”警告框 |
| T-3 | 输入正确的“登陆名称及密码” | 进入系统主界面 | 进入系统主界面 |

（2）系统其他主要功能模块的测试如表6-2所示。

表6-2 主要功能操作测试

|  |  |  |  |
| --- | --- | --- | --- |
| 测试用例 | 操作或数据输入 | 显示或数据输出 | 实际输出结果 |
| T-1 | 教师发布考试信息 | 考试新添加成功 | 显示添加成功的信息 |
| T-2 | 学生进入答题模块 | 显示题目 | 显示题目 |
| T-3 | 结束考试 | 考试成绩 | 提交成功，刷新当前页面显示当前成绩信息 |

经过多次测试和改进，原提出的系统功能已基本实现，各项功能需求基本完成。现在系统处于测试阶段，还存在一些不够优秀的地方。后期投入使用后，结合用户试用后的反馈，对提出的修改建议进行改进。

# **7 结束语**

**7.1 总结**

本文从在线考试系统行业的背景和发展现状入手，揭示了开发该系统的意义。然后介绍了系统的总体设计以及架构，按软件开发的流程首先开始详细开展了系统的可行性研究以及需求分析。然后从系统的详细设计动手分析，之后开始介绍编码实现到最后的系统测试，具体描述了系统的实现过程。

经过本次考试系统的软件设计和实现,我已经掌握了如何使用云服务器等一系列新的软件开发网站的具体操作流程和软件设计的技巧,并对软件开发工具也有了进一步的认识和理解,在软件设计和实现过程中,我学习到了如何在网络上使用云服务器等一系列新技术和知识。在网站的开发中我使用了vue、spring Boot、 MySQL等各种技术类型，加深了我在 web 开发中熟练度的基础。

本系统使用简单，页面美观大方，但由于本人能力尚且不足，目前该系统还存在着一些缺陷，主要表现在以下两个方面：

1. 在数据库设计方面还存在着一定问题，很多的字段属性定义不清，并且没有考虑每个字段的具体范围，因此在后期数据库处理上产生了一定影响，并且数据库的优化上做的还不是很好。

（2）在线考试用户数受系统架构及服务器性能限制，无法支撑大规模的同时在线人数。但目前本人能力尚且不能解决这个问题，等到以后技术更加全面成熟一定妥善解决这个问题。

**7.2 展望**

鉴于目前在线考试行业发展的迅猛姿态，就着先入为主的理念，我想尽快将本在线考试系统平台上线部署并开始运营，之后进行更好的宣传和推广。假以时日，相信可以取得客观的经济效益。

**参考文献**

[1]夏一名. 基于Web的在线考试系统设计与实现[D].电子科技大学,2012.

[2]熊志勇.基于在线考试系统的PHP课程教学探索和应用[J].电脑知识与技术,2021,17(04):150-151.

[3]董晓睿,邸文华,崔健,商凯.基于Spring Boot的在线考试系统的设计与开发[J].电脑编程技巧与维护,2020(10):36-38.

[4]刘金羽.前后端分离的在线考试系统设计与实现[J].电脑编程技巧与维护,2020(04):44-46.

[5]张春雨. 大规模在线考试考务综合管理系统的设计与实现[D].哈尔滨工业大学,2020.

[6]张杰. 基于React+Spring的教学系统设计与实现[D].山东师范大学,2019.

[7]叶惠仙,沈文杰.基于Bootstrap+spring boot框架的在线考试系统开发[J].网络安全技术与应用,2019(12):54-57.

[8]赵雪辉,王金峰,于善彬,董普祥,张雪玉.基于SSM框架的在线考试系统的开发与设计[J].现代信息科技,2019,3(07):82-84.

[9]赵晓君. 基于B/S框架高校网络考试信息管理系统的设计与实现[D].电子科技大学,2015.