

摘 要

在高等教育的逐渐普及下，学生人数逐年增加，计算机技术发展速度也非常快，在网络上办公教学渐渐流行。由于当今社会各种无法控制的因素增加，线下现场考试就越来越繁琐。此外，手工评分和提交文件也会使效率大大折扣，给老师添麻烦给学校带来压力，这就是为什么开发了 this 在线考试系统。就技术的使用而言，Vue.js 用于使页面更清晰、更美观。系统的前台由 JSP 页面显示，后台使用 SpringBoot 框架，数据库使用流行的开源关系 MySQL 数据库。系统管理员通过实现对所有学生、教师和其他员工以及考试信息的统一管理，使数据存储更容易。教师可以使用该系统在评判学生试卷和执行评分过程时发布试题。学生可以在线作答并查看过去的考试成绩。在线考试系统可以非常方便地满足日常的考试、阅卷等需求，节约大量的人力物力资源。

关键词：在线考试；Springboot 框架；MySQL 数据库

ABSTRACT

With the gradual popularization of higher education, the number of students has been increasing year by year, and the speed of computer technology development is also very fast. Office teaching on the internet is gradually becoming popular. Due to the increasing number of uncontrollable factors in today's society, offline on-site exams are becoming increasingly cumbersome. In addition, manual grading and submission of documents can greatly reduce efficiency, causing trouble for teachers and putting pressure on schools, which is why this online exam system has been developed. In terms of the use of technology, Vue.js is used to make pages clearer and more aesthetically pleasing. The front end of the system is displayed on JSP pages, the backend uses the SpringBoot framework, and the database uses the popular open source relational MySQL database. System administrators make data storage easier by implementing unified management of all students, teachers, and other employees, as well as exam information. Teachers can use this system to publish test questions when evaluating student papers and executing the grading process. Students can answer online and view past exam scores. The online exam system can easily meet the needs of daily examination and marking, and save a lot of human and material resources.

Key Words: Online exam; SpringBoot framework; MySQL database

目 录

第 1 章 绪论	1
1.1 课题背景及意义	1
1.2 国内外研究现状	1
1.3 主要研究内容	2
第 2 章 关键技术介绍	3
2.1 JavaEE 语言	3
2.2 SSM 框架	3
2.3 JSON 前后端交互	4
2.4 Tomcat 服务器	5
2.5 本章小结	5
第 3 章 需求分析	6
3.1 可行性分析	6
3.2 系统功能分析	6
第 4 章 系统设计	10
4.1 功能模块设计	10
4.2 数据库实体设计	10
4.3 数据库表设计	12
第 5 章 系统实现	16
5.1 系统总体的逻辑结构图	16
5.2 登录模块的实现	16
5.3 管理员功能模块的实现	18

5.3.1 学生管理.....	18
5.3.2 教师管理.....	22
5.3.3 管理员管理.....	23
5.3.4 班级管理.....	24
5.3.5 单选题管理.....	24
5.3.6 判断题管理.....	28
5.3.7 填空题管理.....	28
5.3.8 简答题管理.....	29
5.3.9 科目管理.....	30
5.3.10 公告管理.....	30
5.3.11 修改密码.....	33
5.4 教师功能模块的实现.....	34
5.4.1 添加试卷.....	34
5.4.2 教师评阅.....	36
5.5 学生功能模块的实现.....	37
5.5.1 在线考试.....	37
5.5.2 查看考试试卷.....	40
5.5.3 考试成绩.....	41
第6章 系统的测试.....	44
6.1 系统测试的目的.....	44
6.2 系统可用性测试.....	44
6.3 系统典型测试用例.....	44
6.4 系统性能测试.....	45
6.5 本章小结.....	46

结论	47
参考文献	48
致 谢	49

第 1 章 绪论

1.1 课题背景及意义

之前 11 月 11 日还仅仅是光棍节，十年后就变成了风靡全国以及全世界最具人气的购物节，只看其中一个平台，销售数据就高的吓人，所以说以往的生活方式正在悄然被互联网改变。同样受互联网冲击的还包括教育领域，各大在线教育平台早在十年前就开始陆续成立并发展，比如像我们熟知的腾讯课堂、钉钉、淘宝教育、网易云课堂等。涵盖的教学有实用软件、IT 与互联网、外语学习、生活家居、兴趣爱好、职场技能、金融管理、考试认证、中小学、亲子教育等十余大门类^[9]。

就现阶段来讲，在教育领域一种转变正在悄然发生---传统的课堂教育转向在线教育。互联网的发展是催化剂，与传统教育产生化学反应生成的这种历史性变革。优势在于考试管理方面及教学效率的提升。通过这样一个平台，教师可以进行组卷和发布试题，学生可以在网上做题然后查看自己过去成绩并进行分析。

1.2 国内外研究现状

近些年在国内，有些测试已经被网上测试的形式取代，此测试方法也在高等院校中广泛应用，增强了测试与阅卷的工效，节约大量的手工劳动。任日丽在网上阅卷程序的题库中，以文本文档形式保存着试卷及其参考答案，通过 Java 语言将数据库与动态服务器串通起来，用户使用学生端登陆界面即可参加测试，此举体现于其在 2017 年发表的《智能阅卷系统的设计与开发》中^[7]。在测试过程中，学生线上作答，阅卷时，程序会将试卷随即分发给阅卷老师，安全有效且公平公正。该系统结合网上扫描、网络工程以及数据匹配技术，分类题型后老师与电脑共同参与批改工作。批改结束，将所有分数汇总，就是最后分数。

在海外对于网上教学早有研究，线上测试也已有较长时间的实践。海外多国创建了多种形式的网上测试系统，并将其应用于实际。有研究发现，批改试卷由手工演化到机器再由其进化至电脑，准确度直线上升，几乎无错误。简答题方面，他们用过 PEG、E-rater，国内采用的是数据库匹配和及智能匹配算法等，都是以现存语言为基础。有科研人员在研究中，运用了拟合和 N 元语感值量化方法。意思是从库中抽取 N 个词组，并算出其支持度。如果在门槛值以下，就研究语言问题段落然后筛选出问题种类，在学生写作中予以参考。

1.3 主要研究内容

此在线考试系统是基于 Springboot，使用较流行的 SSM 框架开发，MyBatis 可以与系统适配，结合起来可以高效的达成实际工作需求。此系统基于 Web，开发语言是 Java，然后用关系数据库实现。系统有三个角色：学生、教师和管理员。主要功能模块是：用户模块、试卷模块、答题模块、公告模块。本文基于该系统的基本功能来撰写的，其主要结构是：

第 1 章 绪论：对此在线考试系统的选题背景、研究内容、研究方法的介绍，并对在线考试系统作大致的前瞻。

第 2 章 关键技术介绍：怎样更具效率、更好的设计一个系统，用到的何种技术，技术的主要效用，技术间的交织，会在本章中提及。

第 3 章 需求分析。此章节重点阐述了系统需求，将实际的需求问题扭转为计算机问题，用建模的方法对系统需求进行分析，罗列待解决的问题，对系统提出问题。

第 4 章 系统设计。经过上一章对问题的总结，进而对该系统进行总览设计，其中内容分别是功能模块设计、数据库设计等。

第 5 章 系统实现。基于以上设计，如何用系统化的技术来实现，比如界面是怎样形成的，一个界面是如何被前端转换为前端页面的。前端和后端如何互动沟通，如何实现事件的登记等。

第 6 章 系统测试。此章节为最后一章，根据以上所介绍的系统的设计实现，做一次功能测试，同时也写一份测试用例。

第 2 章 关键技术介绍

随着 Web 技术的发展和应用中，其技术的不断发展被各种计算机技术的发展所推动。开发人员工作量的减少和应用的使用更加方便得益于之前的 C/S 模式开发转向到现在越来越多公司选择 B/S 模式的 web 应用进行开发。各种在线系统的设计与实现一开始是为了给普罗大众更持久的服务，这就使得系统需要持续地修复和更新，因此对系统的主要技术进行了谨慎的打磨，使得系统项目持久运转且更易维护。

2.1 Java EE

Java EE 也可以被叫做 J2EE，“Enterprise”，“Edition” 被其中的两个“E”代表，正如这个名称所指代的那样，此语言是为企业应用程序而开发的一种 Java 编程语言。这个语言还没被发布出来的时候，尽管 Java 已经在企业程序中得到了大量的应用，由于工作人员的自身原因，也会产生许多问题，特别是维护方面。在 J2EE 问世之后，客观来讲它确实是很厉害的一门语言，它有一套标准是以接口为规范的进而把整个编程行业很好的管理了，从而促使了 Java 的前进。J2EE 被分成了三个阶层，分别是表示、业务逻辑、数据存取层次，从下图可以看到。这样的优点就在于可以将各个项目划分的很清楚，因为这种级别的应用一般规模都大，所以对程序员进行分工会更加高效也好进行管理和维护。如下图 2.1 所示。

图 2.1 Java EE 体系结构图

2.2 SSM 框架

三大架构：Spring，SpringMVC 和 MyBatis，就是 SSM 架构。Spring 是其最主要的，也就是所谓的设计性架构。IOC 和 AOP 是主心骨理念，就是 Inversion of Control，主要方法是采用反射，减少不同对象间的耦合性是工厂模式创造以 IOC 为接口容器的主要目的，将工作人员的控制权转移。也是对 OOP 理念的一个不错的体现。“面向组织的程序设计”被叫做 AOP，添加到了面向对象程序设计，将一个板块分解开来的是 aspect，将新代码添加到新板块里，益处就是减少板块耦合，系统管理变得更高效。因此在 SSM 架构中 Spring 架构最主要。Servlet 被 springMVC 包装，可以和 Spring 架构连接起来，因为它自己也是从 Spring 架构中演化出来的。半自动框架比较全自动框架来讲，可以让查询变得更高效，Mybatis 就属于 ORM (object relational

model) 半自动框架, 表多起来查询更方便, 它同时也能更好的与数据库交互, 并且可以大幅降低人工, 比如敲代码的量。SSM 架构的整个安装位置见下图 2.2.

图 2.2 SSM 框架结构图

2.3 JSON 前后端交互

以 Web 开发为基础的系统, 最初将服务端脚本模式嵌入传统页面早已经发展起来。在这一过程中, 可以发现各种各样的 Web 应用, 系统也由 PC 端过渡到了移动端。Web 应用着实让我们大开眼界, 同时也为我们办了许多事。然而, 之前的旧的嵌入脚本的模式现已经无法满足当代人在实际生活中所面临的各项挑战。所以诞生了 JSON 和 AJAX。Javascript Object Notation 简称 JSON, 是一种对象表示法, 又称 JS objectSpectrum (对象简谱), 用于后端响应并请求数据的一种轻量级数据交换的方法。前端可以很快地在接收到后台的 JSON 数据后得到阐述。JSON 交互的优点很多, 例如快速的数据处理, 解释更容易等等。JSON 是基于 ECMAScript 开发的 Javascript 规范中的一部分, 因此 JSON 被 Javascript 根本上支持。可以像 JSON.parse 那样, 将字符串去序列化为 JSON。

2.4 Tomcat 服务器

Tomcat 是 Apache 旗下的服务器, 此服务器开放源代码, 专门来处理用户的各种要求, 该系统就采用了 Tomcat 伺服器。除了 Tomcat 之外, 还有 Microsoft 公司的 IIS, Tencent 公司的腾讯云等, 但是因为 Tomcat 最受欢迎且这个系统还处在最初阶段, 所以采用的是 Tomcat。在 Tomcat 辅助下做项目仍然很容易。服务器除此之外需要做的事情还有不少, 就不一一介绍了。

2.5 本章小结

这个章节就开展在系统的设计与实现这块展开赘述, 对存在于系统的主要技术进行分析, 如后端所需要的技术如 Spring boot 中的 SSM 框架、前后端交互的 JSON, 关系数据库等。

第 3 章 需求分析

系统的需求分析是整个系统开发过程中最重要的一步，也是整个系统的设计与实现过程中的重要组成部分。该步骤指的是针对所做项目整体展开一个需求的探索，若分析的内容妥当，就能精确的处理好实际所遇到的困难。若不能及时进行分析，将会对以后的系统实施产生影响。需求分析同样决定着一个系统的优劣，若不能进行妥善分析，那么后续的系统设计主要方向就会偏离。

3.1 可行性分析

可行性分析是一种对系统的可执行性的判断。在对其进行全方位分析时，我们主要从三个方面展开分析，分别是：技术可行性、经济可行性、法律可行性。若这三个方面都没问题，那系统基本就可行。

(1) 技术可行性

为了成功开发此以 Web 为基础在线考试系统，我们将 Spring、Springmvc、MyBatis 进行集成，使用 Springboot 架构，以 Java 为基础的 Web 实现技术方案。很完善的框架，像 SSM，Java 的发展趋势也很迅猛，其在很多地方都有很好的解决问题方法，比如网站开发。Springboot 是另一个可以迅速完成全部项目的解决途径。因此，使用以 Springboot 为基础的方法来开发这个项目，是非常好的选择。所以在技术层面上来讲，这完全可以做到。

(2) 经济可行性

经济方向上来讲，该项目开发过程中，必须有一台主用的个人电脑，而生产过程中则必须有一个网络服务器、一个数据库服务器。所以就成本而言，一台电脑成本不高人人几乎都有，所以在这个层面上也没问题。

(3) 法律可行性

法律方向上来讲，该系统对于第三方没有任何问题。Springboot 是开源的，Mysql 数据库的源代码也是开放的。各项信息及与之有关的东西，都在法律范畴内。git 来进行开放源代码的管理，故法律上也可行。

3.2 系统功能分析

针对在线考试系统设计与实现是首先该系统是在 Web 基础上进行开发的，学生、教师和管理员组成用户管理模块。

学生用户的功能需求主要包括如下：

- (1) 账户管理：学生拥有自己的账户信息，能够对自己的账户信息进行管理包括账户的登录密码信息的维护，账户头像的设置等。
- (2) 在线考试：用户可以选中需要参加考试的试卷进行作答。
- (3) 自动阅卷：试卷提交后系统可以自动评判试卷。
- (4) 成绩查询：用户可以查看想要查看的试卷成绩。
- (5) 成绩走向统计：用户可以看到本人的成绩变化趋势。
- (6) 公告查看：用户对公告可以进行查看。

教师用户的功能主要包括如下：

- (1) 账户管理：教师拥有自己的账户信息，能够管理并修改自己的账户密码等信息以及更换头像等。
- (2) 公告查看：用户对公告信息进行查看。
- (3) 在线阅卷：老师可以在线进行阅卷也可以使系统自动阅卷。
- (4) 成绩统计：老师可以看到所管理的学生的成绩分布饼状图。
- (5) 发布试卷：老师可以亲自组卷并发布从而使学生作答。

管理员功能需求主要包括如下：

- (1) 用户管理：管理员可以对所有的用户进行管理，包括对学生、教师和管理员用户信息的增加删除和修改。
- (2) 班级管理：管理员需要对班级信息进行管理。
- (3) 科目管理：管理员需要对科目信息进行管理。
- (4) 公告管理：管理员可以发布公告信息到系统中。
- (5) 账户管理：管理员拥有自己的账户信息，能够对自己的账户信息进行管理包括账户的登录密码信息的维护，账户头像的设置等。
- (6) 题库管理：管理员需要对各类题库进行管理。

学生，教师和管理员用例图如图 3.1-3.3 所示：

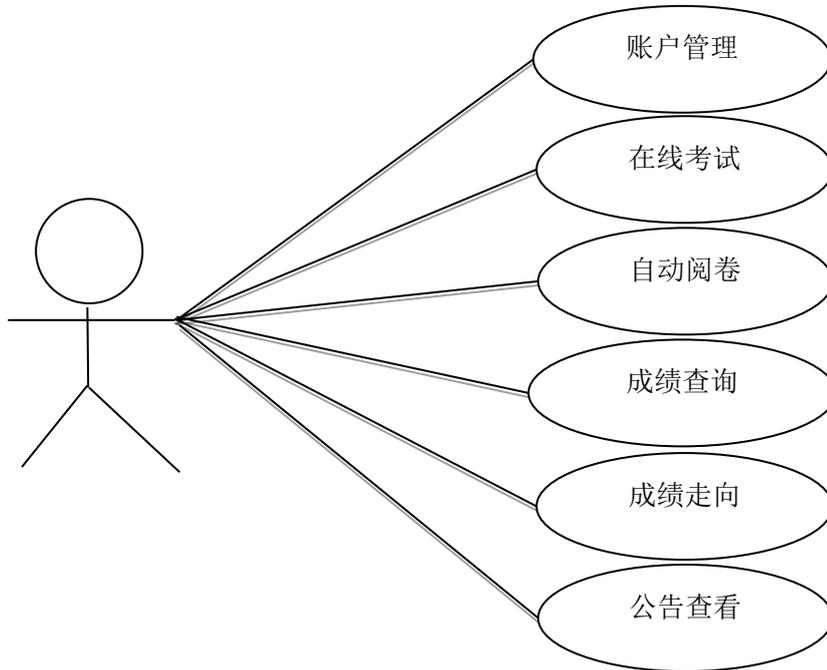


图 3.1 学生用例图

图 3.2 教师用例图

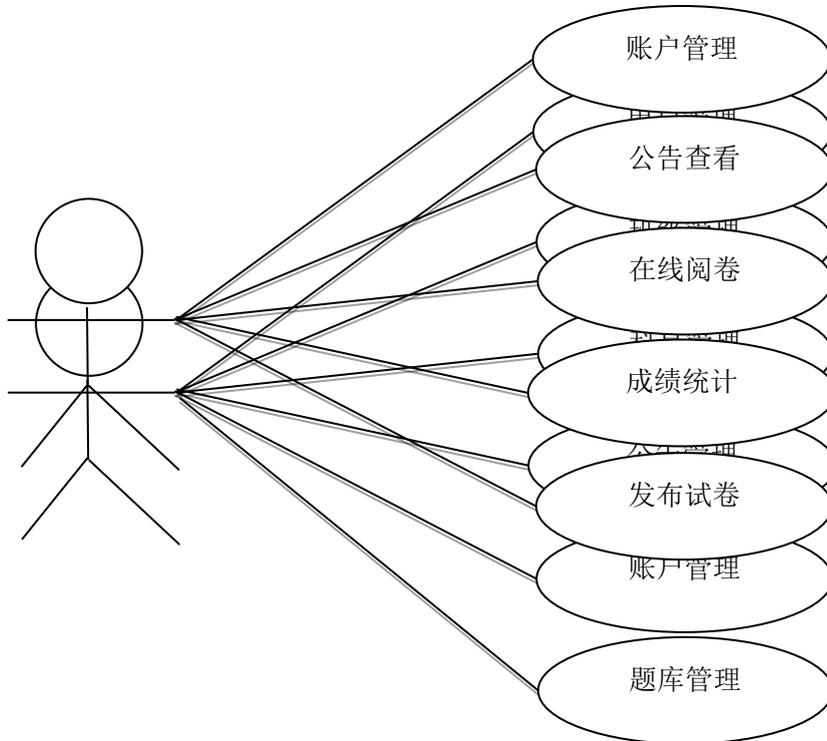


图 3.3 管理员用例图

第 4 章 系统设计

4.1 功能模块设计

在该在线考试系统中，每个学生申请注册后可以查看考试信息，随后可以进行在线答题，试卷作答完成后可以看到自己的相关成绩信息，有考试时可以查看相关公告，以便按时考试，还可以在面中进行个人账户信息编辑。

在该在线考试系统中，老师可以组卷并发布试卷，学生作答后可以进行阅卷，并且可以查看学生作答后的成绩统计信息，老师查看管理员发布的公告以及进行个人账户信息的编辑修改。

在该在线考试系统中，管理员可以管理题库，基础信息中学生、老师、管理员自身和班级信息编辑，可以发布修改公告，可以进行个人账户信息编辑。模块概况如下图 4.1

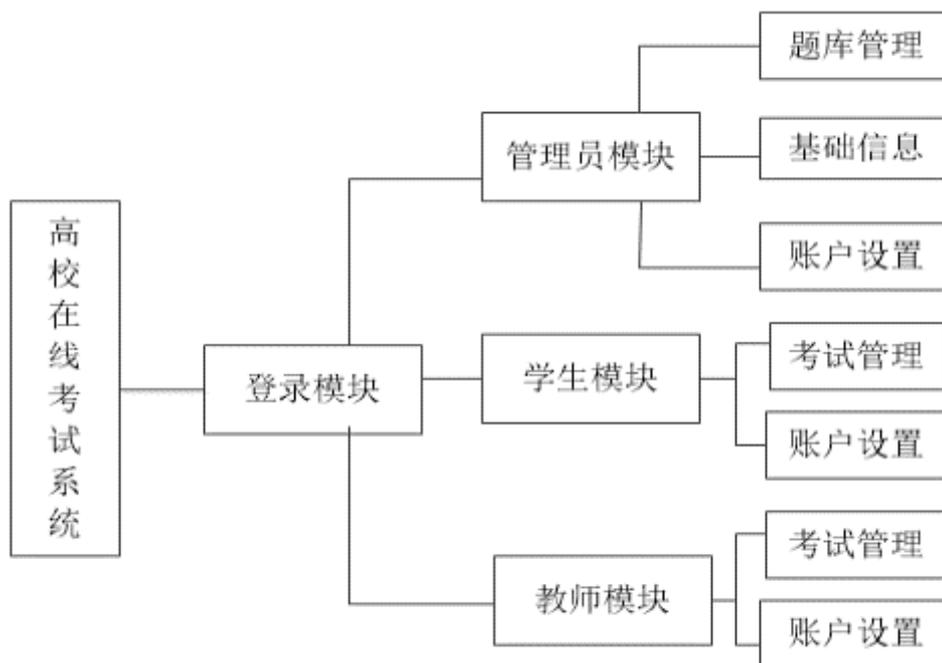


图 4.1 系统功能模块图

4.2 数据库实体设计

按照功能模块进行开发后，要找出和其功能模块匹配的对象。为了直观表达对象间的关系，通常可以用属性图来概况，将其变换为更为直观的图形化模型。

(1) 成绩信息实体属性图，如图 4.2 所示：

图 4.2 成绩信息实体图

(2) 学生实体属性图，如图 4.3 所示：

图 4.3 学生实体图

(3) 单选题实体属性图，如图 4.4 所示：

图 4.4 单选题实体图

(4) 试卷实体属性图，如图 4.5 所示：

图 4.5 试卷实体图

(5) 科目信息实体属性图，如图 4.6 所示：

图 4.6 科目信息实体图

4.3 数据库表设计

紧接着对数据库表格进行设计，三范式标准必须遵守，如此一来后面的工作就将变得更容易：

表 4.1 成绩信息表(chengji)

字段名称	字段意义	字段类型	字段长度	是否主键	能否为空
ID	编号	int	11	是	否
yonghuming	学生	varchar	255	否	是
danxuant	单选题	int	11	否	是
panduanti	判断题	varchar	50	否	是
tiankongti	填空题	varchar	50	否	是
jiandati	简答题	varchar	50	否	是
zf	分数	int	11	否	是

addtime	时间	timestamp	null	否	是
bianhao	编号	varchar	255	否	是
kemu	科目	varchar	255	否	是

表 4.2 单选题信息表(duanxuant)

字段名称	字段意义	字段类型	字段长度	是否主键	能否为空
id	编号	Int	11	是	否
kemu	课程	varchar	50	否	是
timu	题目名称	varchar	50	否	是
xuanxiangA	选项 A	varchar	50	否	是
xuanxiangB	选项 B	varchar	50	否	是
xuanxiangC	选项 C	varchar	50	否	是
xuanxiangD	选项 D	varchar	50	否	是
daan	正确答案	varchar	50	否	是

表 4.3 判断题信息表(judge)

字段名称	字段意义	字段类型	字段长度	是否主键	能否为空
id	编号	int	255	是	否
question	课题名称	varchar	255	否	是
answer	正确答案	int	10	否	是
kemu	所属课程	varchar	255	否	是

表 4.4 填空题信息表(tiankong)

字段名称	字段意义	字段类型	字段长度	是否主键	能否为空
id	编号	int	11	是	否
diffract	小数答案	double	0	否	是
kmid	正确答案	int	11	否	是
title	标题	longtext	0	否	是

表 4.5 简答题信息表(tiankong)

字段名称	字段意义	字段类型	字段长度	是否主键	能否为空
id	编号	int	11	是	否
diffract	小数答案	double	0	否	是
kmid	正确答案	int	11	否	是

title	标题	longtext	0	否	是
-------	----	----------	---	---	---

表 4.6 管理员信息表(users)

字段名称	字段意义	字段类型	字段长度	是否主键	能否为空
id	编号	Int	11	是	否
password	密码	varchar	255	否	是
sex	性别	varchar	255	否	是
createtime	创建时间	datetime	0	否	是
nickname	昵称	varchar	255	否	是
xiangpian	照片	varchar	255	否	是
realname	真名	varchar	255	否	是
username	姓名	varchar	255	否	是
email	邮箱	varchar	255	否	是
tel	电话	varchar	255	否	是

表 4.7 教师信息表(teacher)

字段名称	字段意义	字段类型	字段长度	是否主键	能否为空
id	编号	Int	11	是	否
password	密码	varchar	500	否	是
sex	性别	varchar	500	否	是
jiguan	籍贯	varchar	500	否	是
xueli	学历	varchar	500	否	是
photo	照片	varchar	500	否	是
nation	民族	varchar	500	否	是
name	姓名	varchar	500	否	是
email	邮箱	varchar	500	否	是
tel	电话	varchar	500	否	是
zhicheng	职称	varchar	500	否	是
major	专业	varchar	500	否	是

表 4.8 学生信息表(student)

字段名称	字段意义	字段类型	字段长度	是否主键	能否为空
id	编号	int	11	是	否

yonghuming	学号	varchar	50	否	是
mima	密码	varchar	50	否	是
xingming	姓名	varchar	50	否	是
dianhua	手机	varchar	50	否	是
xingbie	性别	varchar	50	否	是
youxiang	邮箱	varchar	50	否	是
shenfenzheng	身份证	varchar	50	否	是
dizhi	地址	varchar	50	否	是
zhaopian	照片	blob		否	是

表 4.9 科目信息表(kecheng)

字段名称	字段意义	字段类型	字段长度	是否主键	能否为空
id	编号	int	11	是	否
kemu	课程名称	varchar	50	否	是

表 4.10 试卷信息表(shijuan)

字段名称	字段意义	字段类型	字段长度	是否主键	能否为空
id	编号	int	11	是	否
tiankongti	填空题	varchar	500	否	是
bianhao	试卷编号	varchar	50	否	是
danxuantishu	单选题数量	varchar	50	否	是
danxuanfenshu	单选题分值	varchar	50	否	是
panduanfenshu	判断题分值	varchar	50	否	是
tiankongfenzhi	填空题分值	varchar	50	否	是
tiankongtishu	填空题数量	varchar	50	否	是
panduantishu	判断题分值	varchar	50	否	是
addtime	时间	timestamp	50	否	是
danxuantishi	单选题	varchar	500	否	是
kemu	课程名称	varchar	255	否	是
panduantishi	判断题	varchar	500	否	是

第 5 章 系统实现

在系统设计完成后，接下来就是具体实施步骤。整体来看，这个项目被分成了以下几个层次：前端界面、数据库、后端程序。现在，最常用的就是 MVC 模式进行开发，接下来就是详细阐述的前端界面以及后端功能模块如何实现。

5.1 系统总体的逻辑结构图

我们看到从总体上对整个项目逻辑设计，先画出系统的 MVC 结构图。MVC 模型上 Model 层是系统的核心层，它包含了项目的骨干逻辑代码及数据库，Model 层通过 View 层展现具体例子，处理数据与用户相互交流是 Controller 重要手段，前端与后端形成链接归功于 MVC 模式，这很重要，控制器的作用是中介，它将前端请求分别给出，处理数据运用模型层。在这个系统中，SpringMVC 充当主要控制作用，对 Servlet 包装就用到了其中的 DispatchServlet。如下图 5.1 所示。

图 5.1 系统 MVC 结构图

5.2 登录模块的实现

在系统中，ajax 被提交，输入用户名，密码和验证码进入系统。如图 5.2 所示。

图 5.2 用户登录

实现该功能的关键代码如下：

```
handleLogin() {
this.$refs.loginForm.validate(valid => {
  if (valid) {
    console.log("loginForm.code=",this.loginForm.code);
    if(this.code.toLowerCase()!==this.loginForm.code.toLowerCase()){
      this.$message.error("验证码不正确");
      return;
    }
  }
}
```

```
this.loading = true;  
let url = "admin/login";
```

```

        this.$http.post(url, {
            username: this.loginForm.username,
            password: this.loginForm.password,
            usertype: this.loginForm.usertype
        }).then(res => {
            console.log(res.data);
            let data=res.data;
            if(data.stateCode<0){
                this.$message.error(data.des);
                this.loading=false;
                return;
            }
            if (this.loginForm.usertype==1)
window.location="{pageContext.request.contextPath}/admin/index.jsp";
                if (this.loginForm.usertype==2)
window.location="{pageContext.request.contextPath}/student/index.jsp";
                if (this.loginForm.usertype==3)
window.location="{pageContext.request.contextPath}/teacher/index.jsp";
            });
        }
    })

```

5.3 管理员功能模块的实现

5.3.1 学生管理

管理员可以针对学生进行管理，把需要进行考试的学生加入到系统中，也可以对系统中存储的学生信息，包括学号、姓名、性别等进行删除和修改的操作，其界面分别如图 5.3-5.6 所示：

图 5.3 新增学生

图 5.4 编辑学生

实现该功能的关键代码如下：

```
async submitHandler() {  
    const validRes = this.myValidator.valid(this);  
    console.log("valRes", validRes);  
    if (!validRes)  
        return;  
    console.log("birthday=",this.student.birthday);  
    let util = new VueUtil(this);  
    let params = {...this.student};  
    let {data: res} = await util.http.post("admin/student/update", params);  
    if (res.stateCode <= 0) {  
        util.alert(res.des, '系统提示', {  
            confirmButtonText: '确定'  
        });  
        return;  
    }  
    util.message({  
        message: "更新成功",  
        type: 'success',  
        duration: 2000  
    });  
    window.location.href = this.hostHead + "/student/accountinfo.jsp";  
},  
},
```

```
created() {  
    this.load();  
}
```

图 5.5 删除学生

实现该功能的关键代码如下：

```
async deleteRec() {  
    let url = "admin/student/delete";  
    let util = new VueUtil(this);  
    let hasChecked = this.listStudent.some(c => {  
        return c.rowSelected == true;  
    });  
    if (!hasChecked) {  
        util.alert('请选择需要删除的记录', '系统提示', {  
            confirmButtonText: '确定'  
        });  
        return;  
    }  
    let ids = this.listStudent.filter(c => c.rowSelected).map(c => c.id);  
    let res = util.confirm('是否要删除数据?', '系统提示', {  
        confirmButtonText: '确定',  
        cancelButtonText: '取消',  
        type: 'warning',  
    });  
}
```

```

    }).then(() => {
        util.http.post(url, {ids}, {emulateJSON: false}).then(res => {
            if (res.data != null && res.data.stateCode > 0) {
                this.search();
            }
        });
    }).catch(() => {
    });
},
selectedAllHandler() {
    console.log(this.selectedAll);
    if (this.listStudent != null) {
        this.listStudent.forEach(c => {
            c.rowSelected = !this.selectedAll;
        });
    }
}
},

```

图 5.6 查找学生信息

实现该功能的关键代码如下：

```

async search() {
    let url = "admin/student/list";
    let param = {

```

```
        currentpageIndex: this.pageindex,
        pagesize: this.pagesize,
        "name": this.name,
    };
    if (this.bjid!=null)
        param.bjid=this.bjid;
    let util = new VueUtil(this);
    console.log("this.pageindex=" + this.pageindex);
    let res = await util.http.post(url, param);
    if (res.data != null) {
        let pageInfo = res.data.data;
        this.total = pageInfo.total;
        this.listStudent = pageInfo.list;
        console.log(pageInfo);
    }
}
```

5.3.2 教师管理

管理员可以针对参与批阅试卷工作的老师的信息：工号、姓名、性别等进行编辑操作。包括在系统中查找，增加，删除和修改老师的信息。点击相应按钮即可进行操作。如图 5.7 和 5.8 所示：

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：

<https://d.book118.com/958106103071006050>