

**本科毕业设计（论文）**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **题目：** | | |  | | --- | | **基于Vue + SpringBoot 教考分离系统** | | |
|  | | **的设计与实现** | |

|  |  |  |  |
| --- | --- | --- | --- |
| **学生姓名** | **汪思朋** | **学　号** | 201831082129 |
| **教学院系** | **计算机科学学院** | | |
| **专业年级** | **计算机科学与技术2018级** | | |
| **指导教师** | **谯英** | **职　称** | **副教授** |
| **单　　位** | **计算机科学学院** | | |

Southwest Petroleum University

Graduation Thesis



**The System for separation of teaching and testing**

**Grade: 2018**

**Name: Wang Sipeng**

**Speciality: Computer Science**

**Instructor: Qiao Yin**

**School of Computer Science**

**2022-5**

**摘要**

学生的学习成绩是衡量教学质量的一个重要指标。而学习成绩又与教学质量、考试质量、评价方式等因素有密切关系。一直以来，高校中许多课程由授课教师自己授课，自己出题验收，自己阅卷。由此导致了一系列的问题．降低了学校所获取的教学反馈信息的可信度，诱发了部分教师教学、出卷及评分的主观性和随意性，阻碍了学生学习积极性的提高．甚至使部分学生平时不上课，考试时寄托于教师考前复习的“透明度”，临考搞突击，考后脑空空，不能真正培养出知识能力强、专业水平高的学生。教考分离的实施能够更有效地控制考试中的人为因素，最大限度地排除考试的主观随意性和试题的可被猜测性，使试卷的质量更符合教学大纲的要求，更客观地反映教学情况，促进教师钻研教学，切实转变教风:学生转变学风，实现动机与效果的统一、教与学的统一、学与考的统一。确保教学质量稳步提高都具有积极引导作用．从而间接地监控教师的日常教学，提高教学质量。

这个系统的设计和综合了本科计算机科学与技术的各类知识，综合了理论知识和流行技术，还综合了理论知识和业务应用等一些技术的应用。能够体现出设计的专业性，通过此设计，对认识专业在行业内的应用情况和对系统搭建业务流程具有重大意义和作用。

本文对教考分离进行了深刻的分析与研究。在参考学习了超星、慕课网、中国大学Mooc等学习平台后，自主设计了数据库，前台页面样式，以及大致的交互逻辑。

接下来，本文对教考分离系统的各个模块做出详细的介绍，包括流程、用例、功能模块的使用。

关键词: Vue; Spring Boot; MySQL; Java; JavaScript;

## 1 绪论

### 选题的目的和研究意义

什么是教考分离?教考分离考试制度是一种现代教学管理手段。所谓“教考分离”指的是将学院教师的教学工作和考试制度进行分离，即教学与考试分离进行，将过去某一课程由任课教师自己命题、自己评分的做法，改为由教学管理部门组织教学经验较为丰富的非任课老师依据教学大纲命题，或从规范、标准的试题库中筛选、组合出符合要求的试卷．并统一组织考试，统一评阅试卷。教考分离是实现公平竞争的有力措施,是实现教风和学风根本好转的有效办法．是确保教学质量的有效手段之一，能使教学的检查和评定更趋科学、有效和可靠。

为了实现上述的功能，我对超星系统进行了仔细地分析，着重挑选了其中的用户管理功能、试题录入功能、组卷功能进行重点实现。

这个系统的设计和综合了本科计算机科学与技术的各类知识，综合了理论知识和流行技术，还综合了理论知识和业务应用等一些技术的应用。能够体现出设计的专业性，通过此设计，对认识专业在行业内的应用情况和对系统搭建业务流程具有重大意义和作用。

### 相关现状研究

教考分离”在我国基础教育领域极为普遍。小学的学期统测，初高中的学业水平测试均属于“教考分离”，即不由任课教师出题、阅卷、评价学生，而由地区教育部门统一命题、阅卷，评价学生的学习效果与教师的教学效果。

“教考分离”考试制度之所以被我国不少高校所采纳，是因为其本身具备了传统考试制度所没有的优越性。

其优越性主要表现在以下几点

1. 有利于全面贯彻教学大纲

教考合一的考试制度下，教师受教学大纲的约束较少，从而使教学大纲得不到全面贯彻，教学质量难以得到保证。

1. 有利于调整学生的学风

教考合一的考试制度下，学生在考前则要求任课教师划定复习范围，降低试题难度，考试时怀着侥幸心理作弊，考后如考试成绩不理想就找老师求助，这种种现象都滋长了学生的惰性心理，严重破坏了学校学风考风的建设。

1. 有利于强化考试命题的严肃性

在教考合一的考试制度下，题量、题型、知识的覆盖面存在着一定的盲目性和随意性，所以学生的知识面接受存在一定的缺陷性，从而影响教学质量和学习质量。

1. 有利于规范教学管理

在教考合一考试制度下，经常出现考试方式随意，试卷类型多样的现象，从而给教学管理的评估和归档工作带来诸多不便。实施教考分离后，试卷都是从试题库或试卷库中抽取，并由课程负责人或专业教师对试题的分值分布、难易程度、试题类型、知识面覆盖进行统一制定。

### 本文工作

本文说明了基于Vue + SpringBoot 等技术实现的教考分离系统的原理。详细说明了此项工作的技术选型考虑、重点实现功能分析、可行性分析、重难点解决过程分析以及自己对此项工作的一些思考。

### 本文的章节安排

本文对各个章节安排如下:

第一章： 本文的绪论，主要介绍了教考分离系统的产生背景，其国内外现状以及此系统的好处。还有就是此系统的大致开发工作。

第二章：教考分离系统的技术选型研究，为什么要用Vue 和 SpringBoot 框架，web端和后端的相关开发技术介绍。

第三章：教考分离系统的具体实现以及过程分析。按照实际的工程化思路，构建完整的应用流程。

第四章：教考分离系统的测试。

## 技术选型的分析与研究

### 2.1 前端技术选型分析

#### 2.1.1 HTML、CSS、JavaScript 介绍

|  |
| --- |
| 1. HTML  超文本标记语言（英语：HyperText Markup Language，简称：HTML）是一种用于创建网页的标准标记语言。HTML 运行在浏览器上，由浏览器来解析。HTML 可以看作一个程序版的 Word，我们在Word中进行文字的编写、图片的排版以及一些超链接的引入等，都是使用office直接提供的GUI进行操作，而HTML 就是命令式操作。  比如向Word中插入一个图片，需要从本地文件中复制或者引入，批量插入的时候就比较麻烦，而HTML用一个 <img src=’’> 标签就可以完成这些事情，非常的方便。  2. CSS  上文说到HTML 可以像word那样插入各种各样的内容，有的时候 word还需要排版、整理文字或者其他内容的样式，在word中可以通过开始菜单中的控制面板进行控制，而在HTML 中，CSS就是起了一个这样的作用，比如我给HTML某段内容设置文字颜色为红色，那么可以用CSS { color: red }, 进行控制。  3. JavaScript  上述的HTML和CSS已经能够覆盖 word 的大部分使用场景了，为什么还要使用 JavaScript呢？  其实大部分时候，我们在使用word的时候，都是使用的一个静态的页面，也就是只有只读功能，但是我们的系统还需要进行一个复杂的操作，是HTML 和 CSS 不能实现的，这时候就需要JavaScript来实现了。  JavaScript是一个运行在浏览器运行时的一个动态解释型语言。这也就决定了JavaScript只能运行在浏览器之上，各个厂家的浏览器内核不同，为了统一JavaScript的运行时标准，欧洲计算机协会指定了ECMAScript 规范。保证各个厂家的浏览器都实行同一套标准。此次系统中JavaScript主要用来进行 IO操作，网络操作，渲染操作。 |

#### 2.1.2 Vue 介绍

|  |
| --- |
| 上文说到 HTML、CSS、JavaScript已经能够实现此系统的功能。但是实现起来相当地繁琐。比如我想要获取一个输入框中的内容，我需要先获取这个输入框的Dom，然后取得其中的值，一个注册页面，有很多的输入框，挨个取获取内容太麻烦了。再比如我要去渲染一个列表，我首先要通过网络请求拿到后端的数据，然后将数据与HTML进行拼接，如果是一个比较复杂的列表，拼接的过程十分麻烦，而且极易出错，不方便修改也不易维护。  这个时候，就需要去寻找一个JavaScript框架来解决上述问题，那么我选用的是Vue框架。Vue是一个数据驱动的MVVM框架。对比于传统的 MVC框架，其中的VM(view model)可以理解为MVC框架中的 controller, view就是HTML模板，model就是JavaScript中需要渲染到HTML上的变量, VM相当于一个中间人，帮助我们把 view和model绑定起来，将model的数据渲染到view上，同时也让view的改变映射到model中，利用设计模式中的【发布/订阅模式】实现了双向绑定。 |

### 2.2 后端技术选型分析

#### 2.2.1 Java 介绍

|  |
| --- |
| Java是一门面向对象的编程语言，不仅吸收了C++ 语言的各种优点，还摒弃了C++里难以理解的多继承、指针等概念。因此Java语言具有功能强大和简单易用两个特征。因为在实习工作的时候，接触到的后端同学都是以写Java为主，对Java也就比较熟悉了，同时，Java生态比较成熟，有众多的工具包和解决方案，在我遇到问题的时候，能够快速去社区通过问问题的方式解决。 |

#### 2.2.2 Java Web介绍

|  |
| --- |
| 1. JavaEE  JavaEE 全称 Java Platform, Enterprise Edition，它是对 JavaSE(Java Platform, Standard Edition) 的扩展，加入了面向企业开发（实际上就是网络和 Web 有关开发）的支持，包括 Servlet，WebSocket，EL，EJB 等。简单理解，JavaEE 就是 JavaSE + 更多的 jar 包，这些 jar 包命名以 javax 开头，例如 javax.servlet, javax.websocket 等。 2. Servlet 和 Servlet Container 在 JavaEE 的诸多组件中，做 Web 开发一定躲不开的是 Servlet。Servlet 是一套用于处理 HTTP 请求的 API 标准。我们可以基于 Servlet 实现 HTTP 请求的处理。但是 JavaEE 当中只提供了 Servlet 的标准，要真正运行 Servlet，需要使用 Servlet Container。  如果 Servlet 是电器，Servlet Container 就是电源插座。这层抽象让 Servlet 可以跑在任何一个 Container 当中，隔绝了对 Runtime 环境的依赖。  JavaEE 本身没有提供 Servlet Container，比较常用支持 Servlet Container 的 Server 软件有 Apache Tomcat，Glassfish，JBoss，Jetty 等等。  JavaEE 开发完全可以不使用任何第三方框架，仅仅通过 Servlet 和 JSP 技术，在 JSP 之上构建 View，数据库，业务逻辑等，辅以 EL 表达式等技术简化开发，  3. Java Bean  Java Bean 是一个Java的规范,规定了   1. 所有属性为private(2)提供默认构造方法(3)提供getter和setter(4)实现serializable接口   4. Spring IoC  控制反转 (下文统一简称为 IoC) 把传统模式中需要自己通过 new 实例化构造函数，或者通过工厂模式实例化的任务交给容器。通俗的来理解，就是本来当需要某个类（构造函数）的某个方法时，自己需要主动实例化变为被动，不需要再考虑如何实例化其他依赖的类，只需要依赖注入 (Dependency Injection, 下文统一简称为 DI), DI 是 IoC 的一种实现方式。所谓依赖注入就是由 IoC 容器在运行期间，动态地将某种依赖关系注入到对象之中。所以 IoC 和 DI 是从不同的角度的描述的同一件事情，就是通过引入 IoC 容器，利用依赖注入的方式，实现对象之间的解耦。 |

#### 2.2.3 Spring Boot 介绍

|  |
| --- |
| SpringBoot 内置了一个 Tomcat 服务器，也就是内嵌了一个 Servlet Container，因此直接运行工程就可以看到效果，不需要再进行额外的部署，也无需进行 Servlet 的有关配置。同时 SpringBoot 还会帮我们做许多配置工作。通过 @SpringBootApplication 这个 Annotation，它会帮我们打开 @EnableWebMvc，以及 @ComponentScan 功能，这意味着 SpringBoot 会和 SpringMvc 找到同一个 package 下的 @Controller，@Configuration 等类，进行自动配置。  简单来说，Spring Boot 就是帮我在Java Web 开发中简化了很多配置。 |

#### 2.2.4 MySql 介绍

|  |
| --- |
| MySQL是一个关系型数据库，并与 PHP、Java 等主流编程语言紧密结合。它简单易用并且有免费的社区版供我使用，体积小，安装方便，易于在本机和服务器进行调试。 |

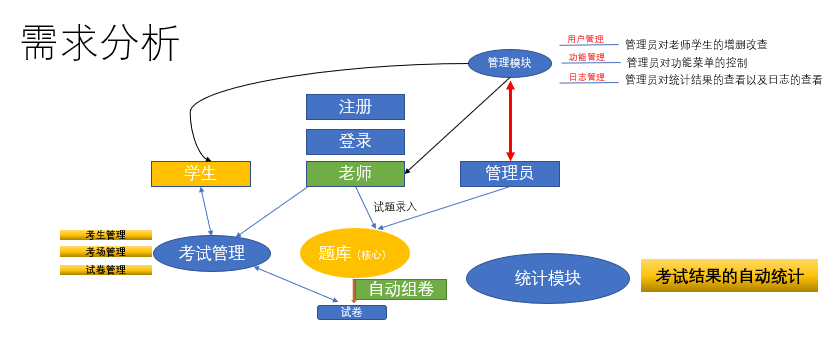
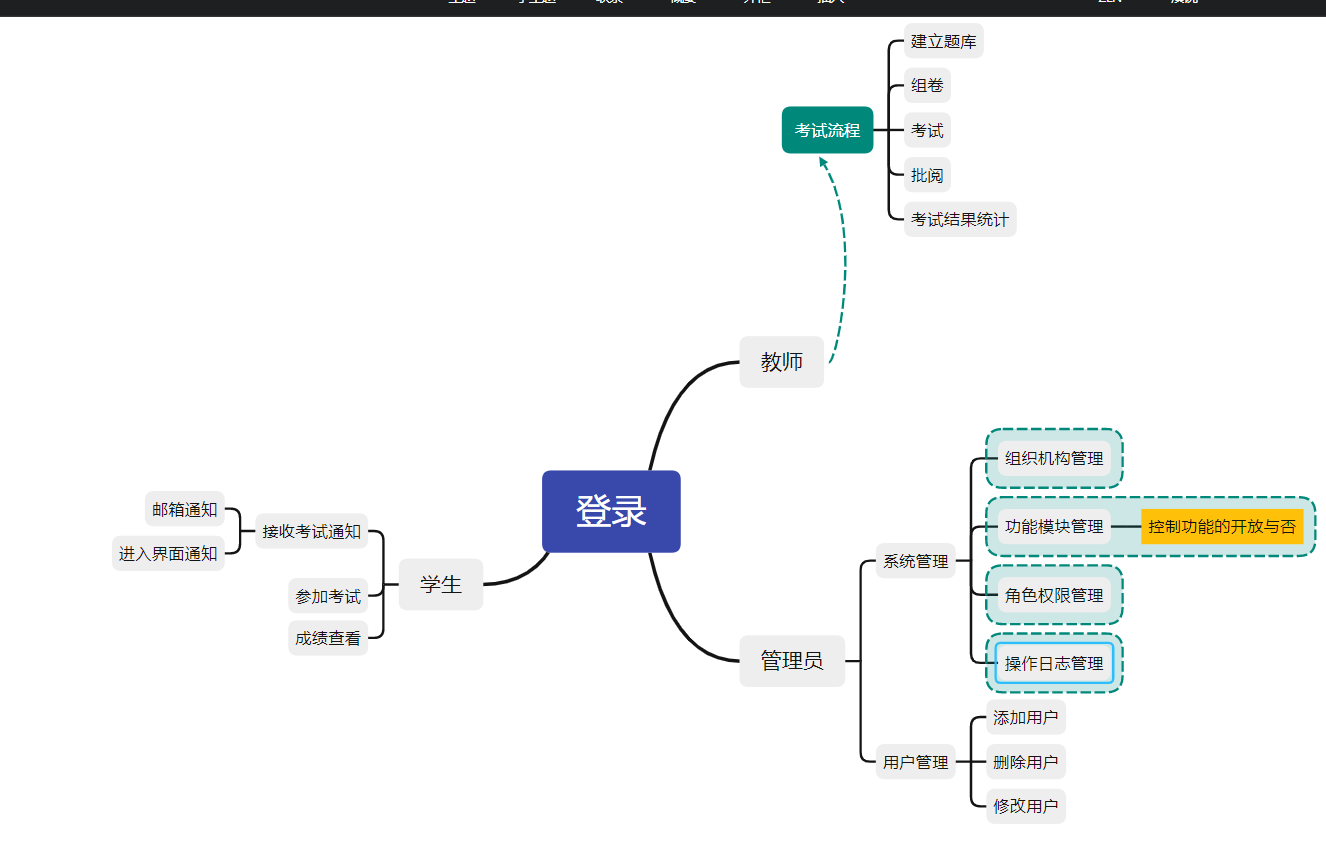
### 2.3 本章总结

整个系统的后台部分，采用传统的Java Web 技术路线进行开发，Java、MySQL、springboot、maven等等都需要我去从零开始学习，其实自己在后端方面也有着广泛的涉猎，用Node写过后台，大概的工程化的东西与Java Web相差不大，所需要学习的就是语言范式、工程结构、代码组织等配置类的东西。

前端部分，因为我有着一段相关的实习经历，所以实现起来不会太难。

## 4总体设计

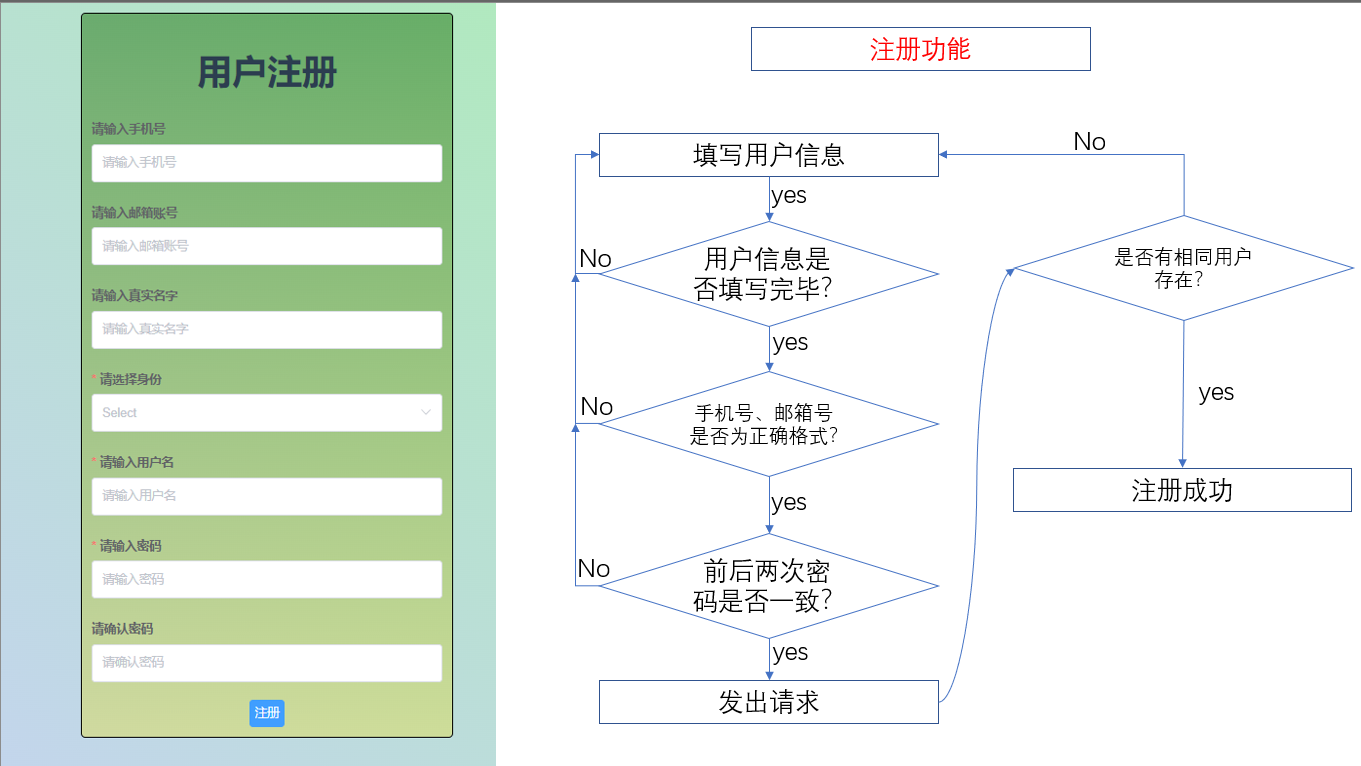
### 4.1 需求分析



### 4.2 各功能模块流程分析

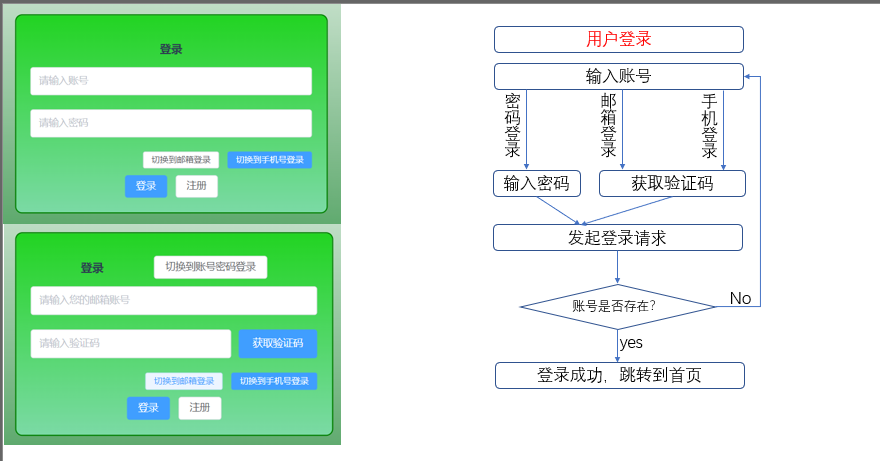
#### 4.2.1 用户注册模块

|  |
| --- |
| 首先用户会在前端的页面进行基本信息的填写，手机号和邮箱使用正则表达式对用户的输入进行验证，同时也会验证表单内容是否填写完整，在输入密码的时候，为了防止用户随意输入，会对密码进行二次验证，当以上三个条件同时通过的时候，才会通过HTTP去向后端发起请求，然后后端会去数据库中根据用户名进行查询，如果查询到此用户存在于数据库，会返回前端一个用户已经存在的信息，此次注册失败，用户需要重新填写用户名或者身份信息。 |



#### 4.2.2 用户登录模块

|  |
| --- |
| 在注册的时候，填写了用户名、手机号和邮箱号，所以用户不仅可以通过账号密码进行登录，也可以通过手机号、邮箱号进行登录。但是由于个人没有手机发短信的申请权限，暂时不能用手机号进行登录。  用户可以在这几种登录模式来回切换，当切换到邮箱登录的时候，发送完验证码，为了防止用户频繁点击导致服务器压力过大，会给发送按钮一个loading的状态，限制用户在60s之内不能重新获取验证码，后端也做了兜底限制处理。 |



### 4.3 重难点实现分析

#### 4.3.1 登录模块的实现分析