**《专业技能训练—数据库应用系统开发》**

**报告**

**课程设计题目：学生选课管理信息系统**

|  |  |
| --- | --- |
| **班　　级：** | **软件工程2班** |
| **姓　　名：** | **庞观艺** |
| **学　　号：** | **202124113234** |
| **任课教师：** | **李泓波** |
| **开始时间：** | **2023年 10 月 10 日** |
| **结束时间：** | **2023年 10 月 12 日** |
| **成　　绩：** |  |

[第 1 章 绪论 3](#_Toc153116959)

[1.1概述 3](#_Toc153116960)

[1.2项目背景 4](#_Toc153116961)

[1.3编写目的 4](#_Toc153116962)

[1.4软件定义、 4](#_Toc153116963)

[1.5开发环境等内容 4](#_Toc153116964)

[第 2 章 开发环境及相关要求 5](#_Toc153116965)

[2.1 Java语言 6](#_Toc153116966)

[2.2 spring框架 6](#_Toc153116967)

[2.3 Mybatis框架 6](#_Toc153116968)

[2.4 MySql数据库 6](#_Toc153116969)

[2.5 硬件、软件要求 6](#_Toc153116970)

[第3章 需求分析 6](#_Toc153116971)

[3.1 安全性要求 6](#_Toc153116972)

[3.2 完整性要求 7](#_Toc153116973)

[第4章 概念结构设计 7](#_Toc153116974)

[数据库设计表 7](#_Toc153116975)

[第 5 章 专业技能训练内容 10](#_Toc153116976)

[5.1业务描述 10](#_Toc153116977)

[第 6 章 系统设计 11](#_Toc153116978)

[6.1 系统总体设计 12](#_Toc153116979)

[6.1.1 系统结构设计 12](#_Toc153116980)

[6.1.2 模块划分 12](#_Toc153116981)

[6.1.3 数据流程设计 12](#_Toc153116982)

[6.2 系统角色设计 12](#_Toc153116983)

[6.2.1 管理员 12](#_Toc153116984)

[6.2.2 教师 13](#_Toc153116985)

[6.2.3 学生 13](#_Toc153116986)

[6.3 功能模块设计 13](#_Toc153116987)

[6.3.1 管理员信息 13](#_Toc153116988)

[6.3.2 教师信息 13](#_Toc153116989)

[6.3.3 学生信息 14](#_Toc153116990)

[6.3.4 学院信息 14](#_Toc153116991)

[6.3.5 专业信息 14](#_Toc153116992)

[6.3.6 课程信息 14](#_Toc153116993)

[6.3.7 选课信息 14](#_Toc153116994)

[6.3.8 其他功能模块 14](#_Toc153116995)

[第 7 章 系统功能的实现 15](#_Toc153116996)

[7.1 主要功能的实现 15](#_Toc153116997)

[7.2．主要模块的设计以及代码 15](#_Toc153116998)

[7.2.1 权限限制 15](#_Toc153116999)

[7.2.2 密码安全性 15](#_Toc153117000)

[7.2.3 验证码校验 16](#_Toc153117001)

[第 8 章 设计开发 17](#_Toc153117002)

[8.1 设计思路 17](#_Toc153117003)

[8.2 具体开发过程 17](#_Toc153117004)

[第 9 章 个人总结 18](#_Toc153117005)

# 第 1 章 绪论

## 1.1概述

1.1 概述

在现代信息技术的背景下，JavaWeb数据库应用正逐渐成为计算机科学与技术领域中的研究和实践热点。本结课作业旨在通过实际项目的开发，深入理解和掌握JavaWeb数据库应用的相关技术和方法。

随着信息化时代的来临，学校管理和信息处理的需求日益增长。为了更好地满足学校、老师和学生的需求，我们选择了一个具有实际应用意义的主题——学生选课信息管理系统。该系统不仅涉及前沿的JavaWeb技术，还包含数据库设计与管理、用户权限控制、界面美化等多个方面，对于提高学校管理效率和提升教学服务质量都具有积极的意义。

通过本次结课作业，我们旨在培养学生对JavaWeb数据库应用的兴趣和深度理解，锻炼学生分析和解决实际问题的能力。通过参与学生选课信息管理系统的设计与实现，学生将有机会综合运用所学的Java、前端技术、数据库管理等知识，提高团队协作和项目管理的能力，为将来从事相关领域的工作奠定坚实基础。

此次结课作业将涉及系统的需求分析、系统设计、编码实现、测试调试等多个阶段，通过项目实践，学生将更全面地理解JavaWeb数据库应用的开发流程和方法，为将来进一步深造或从事相关工作打下扎实的基础。

## 1.2项目背景

随着信息技术的不断发展，学校管理日益复杂，学生选课、教师开课等管理任务需要更加高效和便捷的解决方案。本结课作业立足于JavaWeb数据库应用，旨在开发一个全面的学生选课信息管理系统，以满足学校、老师和学生在选课流程、教学安排等方面的需求。

JavaWeb数据库的结课作业。开发一个学生选课信息管理系统，方便学校，老师，和学生进行选课，开课等等，以此提示相关课程安排实践。

通过此项目，我们旨在促使学生充分参与到系统设计和实现的全过程中，通过实际操作加深对JavaWeb数据库应用的理解，提高团队协作和项目管理的能力。此外，通过满足学校管理的实际需求，我们也将为学校提供一种更加先进、便捷的教务管理解决方案，推动学校管理信息化的进程。

## 1.3编写目的

本结课作业旨在通过学生选课信息管理系统的开发，实现以下目标：

1. 提高数据库应用系统开发能力：学生将深入参与数据库应用系统的设计、开发、测试等全流程，提升对数据库操作、系统架构设计等方面的实际技能。

2. 增强个人技能素养：参与项目将培养学生解决实际问题、协作沟通和团队协调的能力，提高在工作场景中的适应力。

3. 综合应用多领域知识：学生将整合JavaWeb技术、前端技术、数据库设计等多个领域的知识，形成对整体技术体系的更全面认识。

4. 激发创新精神：通过解决实际问题，培养学生创新思维，提高问题解决的创造性，为未来职业发展打下基础。提高数据库应用系统开发能力，增强个人的技能素养。

## 1.4软件定义、

学生选课信息管理系统是一种基于JavaWeb技术的数据库应用系统，旨在实现学生选课过程的自动化管理和信息化处理。通过该系统，学校、老师和学生可以方便地进行选课操作、查看课程信息、管理学生选课记录等功能，从而提高选课流程的效率和准确性。

## 1.5开发环境等内容

1.5 开发环境

为了完成学生选课信息管理系统的开发，我们选择了以下技术栈，采用Java Server Pages（JSP）作为前端视图层，Bootstrap用于前端界面设计，Spring MVC负责系统的后端控制器和业务逻辑，而MyBatis则用于数据库访问与操作，实现了前后端不分离的开发模式。

1.5.1 前端技术 Java Server Pages (JSP)

描述： JSP是一种Java技术，允许在HTML页面中嵌入Java代码。它使得开发人员能够更轻松地创建动态Web页面，与Java后端交互实现数据展示和用户交互。

用途： JSP用于构建系统的前端视图，实现与用户的交互和界面展示。

1.5.2 前端框架 Bootstrap

描述： Bootstrap是一个流行的前端框架，基于HTML、CSS和JavaScript，提供了一套美观而灵活的组件和样式，用于构建响应式的、现代化的Web界面。

用途： Bootstrap用于设计系统的前端界面，增强用户体验，确保页面在不同设备上的友好显示。

1.5.3 后端框架 Spring MVC

描述： Spring MVC是Spring框架的一部分，用于构建基于模型-视图-控制器（MVC）的Web应用程序。它提供了强大的控制器层和灵活的配置，使得开发者能够轻松管理系统的后端逻辑。

用途： Spring MVC负责处理前端请求、控制系统流程，并将处理结果传递给前端展示。

1.5.4 数据访问 MyBatis

描述： MyBatis是一个持久层框架，简化了数据库操作，通过XML或注解配置，将Java对象与数据库表进行映射，提供了便捷的数据库访问方式。

用途： MyBatis用于处理系统与数据库的交互，执行SQL语句，实现数据的增删改查。

采用这一技术栈，我们能够充分发挥各项技术的优势，实现系统的高效开发与维护，同时保持前后端不分离的模式，为学生选课信息管理系统的开发提供了坚实的基础。

# 第 2 章 开发环境及相关要求

**开发环境及相关要求**

* 数据库管理系统选用Microsoft SQL Server 2000/2005、MySQL等。
* 数据库访问方式可选用数据源（ODBC、JDBC）、ADO等。
* 系统开发可根据个人喜好和熟练程度选择任意一种可视化的软件开发工具（如Visual Studio 6 [Visual Basic、Visual C、Visual Foxpro]、Visual Studio.Net、ASP、Java、C、PHP等）。
* 系统界面可采用WinForm或WebForm等形式。

## 2.1 Java语言

采用Java语言

## 2.2 spring框架

采用spring框架，实现前后端不分离模式下进行开发。

## 2.3 Mybatis框架

使用Mybatis框架，高效调用MySql语言，实现后端数据库与前端交互

## 2.4 MySql数据库

使用MySql数据库，

## 2.5 硬件、软件要求

1.Idea编译器

2.MySql 数据库

3.Navicat 数据库(MySql数据库可视化)

# 第3章 需求分析

在系统的安全性与完整性要求，确保学生选课信息管理系统在设计和实施中能够保护数据的安全性，维护系统的完整性。

## 3.1 安全性要求

学生选课信息管理系统需要具备一定的安全机制，以确保敏感信息不受未授权用户的访问，保障数据的机密性、完整性和可用性。具体安全性要求包括：

3.1.1用户身份验证：实施严格的用户身份验证机制，确保只有经过授权的用户才能访问系统，防止非法入侵。

3.1.2数据传输加密：对系统与数据库之间的数据传输采用加密协议，如SSL/TLS，以防止数据在传输过程中被窃取或篡改。

3.1.3访问控制：设定不同用户角色的访问权限，确保用户只能访问其权限范围内的信息，以维护数据的隐私和机密性。

3.1.4日志记录：记录用户的操作日志，包括登录、数据修改等，以便于事后审计，追踪潜在的安全问题。

3.1.5防止SQL注入：使用参数化查询等方式，防范SQL注入攻击，保障数据库查询的安全性。

## 3.2 完整性要求

为了保障系统中的数据完整性，确保数据的一致性和准确性，学生选课信息管理系统需要具备以下完整性要求：

3.2.1数据验证：对用户输入的数据进行合法性验证，防止无效或错误数据的录入，确保数据的准确性。

3.2.2事务管理：使用数据库事务，确保在复杂的操作过程中，数据的一致性得到维护，避免因为异常导致数据不一致。

3.2.3关系约束：在数据库层面设置关系约束，如主键、外键等，以维护数据的关联性，避免不符合业务逻辑的数据出现。

3.2.4备份与恢复：定期对数据库进行备份，并确保能够及时有效地进行数据恢复，以应对意外数据丢失或损坏的情况。

3.2.5版本控制：对系统进行版本控制，确保系统的升级和维护不影响数据的完整性，避免版本切换带来的问题。

# 第4章 概念结构设计

数据库设计表

1. 课程信息表：

图1 课程信息表

1. 学生信息表：



图2 学生信息表

3．教师信息表：

图3 教师信息表

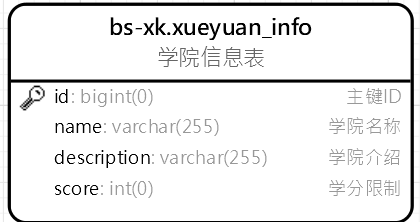
4.学院信息表：

图4 学院信息表

5.选课信息表：

图5 选课信息表

6.管理员信息表：

图6 管理员信息表

# 第 5 章 专业技能训练内容

## 5.1业务描述

设计本系统，模拟学生选课的部分管理功能。学生入校注册后需统一记录学生个人基本信息，对于面向学生开设的相关课程需要记录每门课程的基本信息，每个任课教师规定其可主讲三门课程，学生选课时系统将相应的选课信息记录入库，考试结束后需在相应的选课记录中补上考试成绩。简化的系统数据库（SCDB）中主要包含以下五个数据表（设计时可根据实际业务需要适当追加其它的数据表），各表及其相应字段分别描述如下：

1．学生信息表（student）

学号sno：char(9)

姓名sname：nvarchar(8)

性别ssex：nchar(1) 男　或　女

年龄sage：int(4) 14≤sage≤24

系别sdept：nvarchar(30)

2.课程信息表（course）

课程号cno：char(6)

课程名cname：nvarchar(50)

先行课编号cpno：char(6)

学分ccredit：smallint(2)

3.教师信息表（teacher）

教工号tno：char(8)

姓名tname：nvarchar(8)

性别tsex：nchar(1) 男　或　女

年龄tage：int(4) 24≤sage≤60

学历teb：nvarchar(10) 学士、硕士、博士

职称tpt：nvarchar(10) 助教、讲师、副教授、教授

主讲课程一cno1: char(6)

主讲课程二cno2: char(6)

主讲课程三cno3: char(6)

4.院系信息表（department）

系编号dno：char(3)

系名dname：varchar(30)

系主任dmanager：char(8)

5．选课信息表（sct）

学号sno：char(9)

课程号cno：char(6)

教工号tno：char(8)

成绩grade：int(4)

# 第 6 章 系统设计

## 6.1 系统总体设计

在系统总体设计阶段，我们将关注整个学生选课信息管理系统的架构和主要模块之间的关系。这包括系统的层次结构、模块划分、数据流程等，以确保系统能够高效、可维护和可扩展。

系统总体设计的主要步骤包括：

### 6.1.1 系统结构设计

设计系统的整体结构，包括前端、后端、数据库等各个组成部分的关系和协作方式。

### 6.1.2 模块划分

将系统划分为若干个功能模块，每个模块负责实现系统的一个具体功能或一组相关功能。

### 6.1.3 数据流程设计

定义系统中数据的流动和交互方式，明确数据在各个模块之间的传递路径，确保数据的一致性和准确性。

## 6.2 系统角色设计

在系统设计阶段，我们明确了系统中的三大主要角色：管理员、教师和学生。每个角色有不同的权限和操作范围，以保障系统的安全性和数据的隐私性。

### 6.2.1 管理员

管理员拥有最高权限，可以进行系统中所有模块的增删改查操作。其主要职责包括对用户信息、学院信息、专业信息、课程信息等进行全面管理，确保系统的正常运行和数据的完整性。

### 6.2.2 教师

教师角色具有管理与教学相关的权限，可以查看学院信息、专业信息，并且对自己的课程信息进行管理，包括查看选课学生信息等。

### 6.2.3 学生

学生角色主要负责选课操作，可以查看学院、专业信息，对已有的课程进行选课，并能够在选课信息模块对已选的课程进行取消。此外，学生也具有个人信息管理的权限，包括登录注册、修改密码等操作。

## 6.3 功能模块设计

在系统设计中，明确了系统的各个功能模块，包括但不限于管理员信息、教师信息、学生信息、学院信息、专业信息、课程信息、选课信息等。每个模块都承担着特定的功能，确保系统能够满足用户需求。

### 6.3.1 管理员信息

属性：用户名、密码、性别、年龄、手机号

操作：增加管理员、删除管理员、修改管理员信息、查询管理员信息

### 6.3.2 教师信息

属性：用户名、密码、性别、年龄、职称、所属专业

操作：增加教师、删除教师、修改教师信息、查询教师信息

### 6.3.3 学生信息

属性：用户名、密码、性别、年龄、学号、总学分、所属学院

操作：增加学生、删除学生、修改学生信息、查询学生信息

### 6.3.4 学院信息

属性：学院名称、学院介绍、学院学分限制

操作：增加学院、删除学院、修改学院信息、查询学院信息

### 6.3.5 专业信息

属性：专业名称、系名、所属学院

操作：增加专业、删除专业、修改专业信息、查询专业信息

### 6.3.6 课程信息

属性：课程名称、介绍、学分、所属专业、上课教师、开班人数、上课时段、上课地点、已选人数

操作：增加课程、删除课程、修改课程信息、查询课程信息

### 6.3.7 选课信息

属性：课程名称、介绍、学分、所属专业、上课教师、开班人数、上课时段、上课地点、学生姓名、课程状态

操作：查询选课信息、取消选课

### 6.3.8 其他功能模块

登录注册、修改密码、个人信息管理、退出登录

相关信息的导出Excel文件

密码正则表达式验证

验证码防止爬虫验证

通过以上功能模块的设计，系统将能够提供全面而丰富的管理和操作功能，满足管理员、教师和学生在学生选课信息管理系统中的各类需求。

# 第 7 章 系统功能的实现

## 7.1 主要功能的实现

在设计时，我注重权限的限制，好比如，在选课功能的实现时，并不需要管理员对其进行操作，所以我就在前端页面上设计了限制，我只需要判断当前用户类型即可：如果当前用户不是教师或学生，就不会显示出选课功能的界面。从而减少后端开发的难度。

在登录界面时，用户输入密码，我会设计正则表达式，功能效果显著，可以帮助用户输入更安全的密码，从而降低网络安全威胁。再者，我添加了验证码的校验，这样可以有效防止恶意用户进行爬虫攻击。这样也很好地提高网页的安全性。

权限设置，我的做法：设置一个开关，打开开关，赋予权限。简单的权限设定，只需改一下前端页面判断就行。

## 7.2．主要模块的设计以及代码

### 7.2.1 权限限制

<li v-if="user.level === 1">管理员信息</li>

### 7.2.2 密码安全性

login() {  
 *//先判断密码正则表达式  
 // 首先验证密码是否符合正则表达式* const passwordRegex = /^(?=.\*[A-Za-z])(?=.\*\d)[A-Za-z\d]{8,}$/;  
  
 if (!passwordRegex.test(this.user.password)) {  
 this.passwordError = "密码格式错误，至少包含8个字符，包括至少一个字母和一个数字。";  
 this.user.password = ""; *// 清空密码框* this.clickImg();  
 return; *// 不继续登录逻辑* }

### 7.2.3 验证码校验

*///验证码*@RequestMapping("/captcha")  
public void captcha(HttpServletRequest request, HttpServletResponse response) throws Exception {  
 *//png类型* SpecCaptcha captcha = new SpecCaptcha(135, 33, 4);  
 captcha.setCharType(SpecCaptcha.*TYPE\_ONLY\_NUMBER*);  
 CaptchaUtil.*out*(captcha, request, response);  
  
 *//算术类型  
 /\*ArithmeticCaptcha captcha = new ArithmeticCaptcha(134, 35);  
 captcha.setLen(3); //几位数运算公式  
 captcha.getArithmeticString(); ///公式如 ：3+1=？  
 captcha.text(); //获得结果：4  
 CaptchaUtil.out(captcha, request, response);\*/*}

# 第 8 章 设计开发

## 8.1 设计思路

在设计与开发过程中，我采用了以下步骤和思路：

### **8.1.1. 数据库表设计**

首先，创建数据库表，确保表结构能够满足系统需求，包括用户信息、课程信息、学院信息等。

### 8.1.2. 实体类映射

数据库表相对应的实体类，使用ORM框架（如MyBatis）实现实体类与数据库表的映射关系。

### 8.1.3. DAO层

创建DAO层，包含对数据库的操作，使用MyBatis等框架执行SQL语句，进行数据库的增删改查操作。

### 8.1.4. Service层

创建Service层，实现业务逻辑，调用DAO层完成数据库操作，同时处理业务逻辑，如权限验证、数据处理等。

### 8.1.5. Controller层

创建Controller层，负责接收前端请求，调用相应的Service层处理业务逻辑，并返回结果给前端。使用SpringBoot框架中的注解来处理HTTP请求，如`@GetMapping`、`@PostMapping`等。

### 8.1.6. 前端页面修改

根据业务需求修改前端HTML页面，添加相应的交互逻辑，使用axios等工具发送请求到后端，实现前后端的数据交互。

## 8.2 具体开发过程

以实现分页查询为例，展示了一般业务功能的开发过程：

### 8.2.1.前端页面

在前端页面中添加一个分页查询的按钮或其他交互元素，使用axios发送请求到后端。

### 8.2.2. Controller层

在后端的Controller层中，使用SpringBoot注解（如`@GetMapping`）创建接口，接收前端请求，调用相应的Service层方法处理业务逻辑。

```java

@GetMapping("/courses")

public List<Course> getCourses(@RequestParam int page, @RequestParam int pageSize) {

return courseService.getCoursesByPage(page, pageSize);

}

```

### 8.2.3. Service层

在Service层中，实现分页查询的具体逻辑，调用DAO层方法从数据库中获取数据。

```java

public List<Course> getCoursesByPage(int page, int pageSize) {

int offset = (page - 1) \* pageSize;

return courseDao.getCoursesByPage(offset, pageSize);

}

```

### 8.2.4. DAO层

在DAO层中，使用MyBatis执行相应的SQL语句，从数据库中查询数据。

```java

@Select("SELECT \* FROM courses LIMIT #{offset}, #{pageSize}")

List<Course> getCoursesByPage(@Param("offset") int offset, @Param("pageSize") int pageSize);

```

通过以上步骤，实现了一个简单的分页查询功能。具体的开发过程也适用于其他业务功能，只需根据需求添加相应的前端页面、Controller接口、Service方法和DAO方法即可。

# 第 9 章 个人总结

在本次项目的开发过程中，我积极运用了现代的开发框架，尤其是SpringBoot框架，简化了开发流程，提高了效率。然而，在使用框架的同时，我也深感框架只是一种工具，理解底层的原理和核心概念仍然至关重要。

在系统设计上，我注重了权限的控制、密码的安全性和验证码的校验，以确保系统的安全性。同时，采用了前后端分离的架构，通过API请求实现前后端的数据交互，提高了系统的灵活性和可维护性。

再者，开发过程使用了一些开发框架，如简单的使用到了SpringBoot框架。然而，个人认为所谓的开发框架就是把一些增删改查给你封装一下，让你可以简单跑一下。你也可以自己写SQL语句，根据需求造轮子跑就行。

个人总结中提到了对框架的看法，认为框架是为了简化操作，但强调了底层原理的理解。这体现了我对于开发的态度，既注重实际操作，又强调理论知识的深入理解，以更好地应对复杂的开发场景。在未来的学习和工作中，我将继续保持这种学习态度，不断提升自己的技能水平。