

**本科生毕业论文（设计）**

**教务管理系统设计与实现**

**学部（院） 人工智能与大数据学部**

**专 业 软件工程**

**学生姓名 周慧**

**学 号 210111315 　 年级 2019级**

**指导教师 易文美 　 职称 助教**

**2023年 4月28日**

**学位论文原创性声明**

郑重声明：所呈交的学位论文《教务管理系统设计与实现》，是本人在导师的指导下，独立进行研究取得的成果。除文中已经注明引用的内容外，本论文不包括他个人或集体已经发表或撰写过的作品成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律后果，并承诺因本声明而产生的法律结果由本人承担。

学位论文作者：

日期： 年 月 日

**学位论文版权使用授权书**

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权宜宾学院将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于（请在以下相应方框内打“√”）

保 密🞎，在\_\_年解密后适用本授权书。

不保密🞎。

作者（签名）： 指导教师（签名）：

日期： 年 月 日 日期： 年 月 日

# 摘要

教务管理系统是一种用于高校管理教务工作的软件系统。本文介绍了一种教务管理系统设计与实现方法，旨在提高教务管理的效率和准确性，方便教务工作的各个环节的协调与管理。该系统包括教务班级管理、科目管理、教师管理、学生管理和成绩管理等功能模块的设计与开发。设计阶段首先进行需求分析，明确系统的功能需求和业务流程，然后进行系统架构设计和数据库设计。在实现阶段，采用了面向对象分析，Java、Jsp和Script等技术以及Spring框架，确保系统具备高性能和可扩展性。通过合理的模块划分和功能设计，教务管理系统可以实现教务资源的统一管理。实际应用和测试结果表明，该教务管理系统在学校教务管理中取得了显著的效果，提升了教务管理的质量和效率。

关键词：教务管理系统；Java；面向对象

Abstract

The Academic Affairs Management System is a software system used for managing academic affairs in universities. This article introduces a design and implementation method for an educational management system, aimed at improving the efficiency and accuracy of educational management, and facilitating the coordination and management of various aspects of educational work. The system includes the design and development of functional modules such as academic class management, subject management, teacher management, student management, and grade management. In the design phase, the first step is to analyze the requirements, clarify the functional requirements and business processes of the system, and then design the system architecture and database design. In the implementation phase, object-oriented analysis, Java, Jsp, Script and other technologies as well as the Spring Framework are used to ensure the system has high performance and scalability. Through reasonable module division and functional design, the educational management system can achieve unified management of educational resources. The practical application and test results show that the educational management system has achieved significant results in school educational management, improving the quality and efficiency of educational management.

**Keywords:** Academic Administration Management System;Java;Object-oriented

**目录**

[摘要 I](#_Toc28831)

[Abstract II](#_Toc23351)

[第1章 绪论 1](#_Toc16613)

[1.1 研究背景及意义 1](#_Toc7848)

[1.2 国内外研究现状 2](#_Toc7014)

[1.3 研究内容及方法 3](#_Toc23499)

[1.4 相关技术简介 3](#_Toc19637)

[第2章 教务管理系统需求分析 5](#_Toc8665)

[2.1 可行性研究分析 5](#_Toc12340)

[2.1.1 技术可行性 5](#_Toc2940)

[2.1.2 操作可行性 5](#_Toc6028)

[2.1.3 经济可行性 5](#_Toc18127)

[2.2 业务需求 6](#_Toc30493)

[2.2.1 业务范围 6](#_Toc17059)

[2.2.2主要业务流程 6](#_Toc18102)

[2.3 功能需求 7](#_Toc21180)

[2.3.1 用例确定 7](#_Toc16730)

[2.3.2 用例图 8](#_Toc27424)

[2.3.3 用例描述 10](#_Toc17373)

[2.4 非功能需求 13](#_Toc21165)

[第3章 教务管理系统概要设计 15](#_Toc18834)

[3.1 系统总体结构设计 15](#_Toc28563)

[3.1.1设计目标 15](#_Toc16242)

[3.1.2 功能模块结构图 15](#_Toc9314)

[3.2 系统功能设计 15](#_Toc26969)

[3.2.1 架构描述 15](#_Toc10959)

[3.2.2 登录模块时序图 16](#_Toc12879)

[3.2.3 教师管理模块时序图 17](#_Toc349)

[3.2.4 学生管理模块时序图 18](#_Toc18423)

[3.2.5 班级管理模块时序图 18](#_Toc27531)

[3.2.6 科目管理模块时序图 19](#_Toc8858)

[3.2.7 成绩管理模块时序图 19](#_Toc16441)

[3.3 系统数据库设计 20](#_Toc26218)

[3.3.1数据库概念设计 20](#_Toc24667)

[3.3.2数据库逻辑设计 25](#_Toc12268)

[3.3.3数据库的实现 28](#_Toc16445)

[第4章 详细设计与实现 32](#_Toc11438)

[4.1 登录模块详细设计与实现 32](#_Toc32735)

[4.1.1 登录模块详细设计概述 32](#_Toc10930)

[4.1.2 登录方法程序流程图 32](#_Toc13105)

[4.1.3 登录模块界面设计 33](#_Toc23611)

[4.2 教师管理模块详细设计与实现 33](#_Toc16698)

[4.2.1 教师管理模块详细设计概述 33](#_Toc15060)

[4.2.2 教师管理程序流程图 33](#_Toc14704)

[4.2.3 教师管理模块界面设计 34](#_Toc3024)

[4.3 学生管理模块详细设计与实现 35](#_Toc14534)

[4.3.1 学生管理模块详细设计概述 35](#_Toc22933)

[4.3.2 学生管理方法程序流程图 36](#_Toc1087)

[4.3.3 学生管理模块界面设计 36](#_Toc4207)

[4.4 班级管理模块详细设计与实现 37](#_Toc27911)

[4.4.1 班级模块详细设计概述 37](#_Toc8891)

[4.4.2 班级管理方法程序流程图 37](#_Toc29277)

[4.4.3 班级管理模块界面设计 37](#_Toc30632)

[4.5 科目管理模块详细设计与实现 38](#_Toc30685)

[4.5.1 科目管理模块详细设计概述 38](#_Toc16623)

[4.5.2 科目管理方法程序流程图 39](#_Toc23260)

[4.5.3 科目管理模块界面设计 39](#_Toc30491)

[4.6 成绩管理模块详细设计与实现 40](#_Toc18429)

[4.6.1 成绩模块详细设计概述 40](#_Toc31078)

[4.6.2 成绩管理方法程序流程图 40](#_Toc31278)

[4.6.3 成绩管理模块界面设计 41](#_Toc30009)

[第5章 教务管理系统软件测试 42](#_Toc5793)

[5.1 测试概述 42](#_Toc6849)

[5.2 黑盒测试 42](#_Toc13699)

[5.2.1 教师管理模块测试 42](#_Toc3914)

[5.2.2 科目管理方法测试 44](#_Toc25983)

[5.2.3 成绩管理方法测试 45](#_Toc21298)

[5.3 白盒测试 46](#_Toc10323)

[5.3.1 教师管理模块测试 46](#_Toc16564)

[5.3.2 科目管理方法测试 47](#_Toc6392)

[5.3.3 成绩管理方法测试 48](#_Toc12769)

[总结 49](#_Toc12342)

[致谢 50](#_Toc3962)

[参考文献 51](#_Toc659)

第1章 绪论

1.1 研究背景及意义

随着教育事业的发展和数字化技术的普及，学校管理越来越重视信息化的建设[1]。教务管理作为学校管理的重要环节，也得到了越来越多的关注。教务管理的范围包括教师信息、学生信息、院系信息、年级班级信息、科目信息等。传统的教务管理方式往往存在信息不全、数据处理困难、管理效率低下等问题，难以满足学校管理的需求。因此，设计和实现一个高效、安全、易于维护的教务管理系统变得十分重要。

教务管理系统的目的在于提供一个方便、高效、准确的管理平台，以帮助学校实现教务信息管理的自动化和信息化。教务管理系统的设计与实现，可以带来以下几个方面的研究意义：

1. 提高教学管理效率

教务管理系统可以帮助学校更加高效地管理师生信息、课程信息、考试成绩等重要数据。通过对数据的实时监测和分析，学校可以及时发现和解决问题，提高教学管理的效率和准确性。

1. 改善教学质量

教务管理系统可以为学校提供更为科学的教学管理方式。通过对课程、教学质量和成绩等数据的分析，学校可以及时发现和解决教学中存在的问题，提高教学质量和学生学习效果。

1. 增强师生交流互动

教务管理系统可以为学生、教师提供更加便捷的信息交流和互动方式。通过系统内部的信息发布和交流功能，可以及时传递重要信息，方便学生、教师进行沟通和交流。

1. 推动教育信息化建设

教务管理系统的设计与实现，不仅可以满足学校管理需求，还可以推动整个教务信息化建设。学校的教务信息化建设水平的提高，将有利于推动教育现代化进程，提高教育质量和效率。

综上所述，教务管理系统的设计与实现对于提高教育教学管理效率和质量、增强师生交流互动、推动教育信息化建设等方面都具有重要意义。因此，针对教务管理系统的研究和实现是当前教育信息化领域的重要课题之一。

1.2 国内外研究现状

随着信息技术的不断发展，教务管理系统在学校管理中发挥了越来越重要的作用[2]。在国内外，已经有大量的研究针对教务管理系统进行了探讨和研究。

在国内，教务管理系统的研究已经有了相对完善的体系和成果。许多大学和高等院校已经建立了自己的教务管理系统，并在日常管理中得到了广泛应用。国内的学术界和行业界也对教务管理系统进行了大量研究，主要集中在以下几个方面。教务管理系统的研究重点之一是系统架构和技术[3]。国内的研究人员在这方面主要涉及数据库技术、系统设计和开发技术、网络技术等方面，以提高系统的性能和功能。另一个重要的研究方向是教务管理系统的应用和效果。国内研究人员通过对教务管理系统的应用效果进行评估和研究，发现教务管理系统可以更好更便捷地去工作，也能提高学生学习效果，减轻管理人员的工作负担等等。学生信息涉及到个人隐私，所以教务管理系统的保密性等是重中之重[4]。国内的研究人员主要从数据安全、权限管理等方面提高教务管理系统各方面性质。目前国内的教务管理系统主要分为两类：一类是由学校自行研发的定制化系统，另一类是采用开源系统进行二次开发。其中，一些知名的教务管理系统包括鲲鹏教务管理系统、钉钉教育版、学信网、咕咕教育等。这些系统不仅具备基本的学生信息管理功能，还拥有成绩查询、选课、教学进度管理、教师评价、考试安排等多种功能。此外，国内的教务管理系统也开始向智能化和数据分析方向发展，例如学信网采用了大数据技术，为学生提供了全面的学生档案管理和学业发展分析报告。

在国外，教务管理系统的研究也有很多成果，特别是在系统设计、用户体验设计和应用领域方面。在系统设计和开发方面，国外研究人员主要涉及数据库技术、Web技术和移动技术等方面，以提高教务管理系统的性能和功能。同时，一些研究也关注系统设计的可持续性和可维护性等问题。在用户体验设计方面，国外的研究人员加强了需求分析。研究人员通过对教务管理系统用户的使用习惯、反馈意见等进行分析，提高系统的易用性和用户满意度。目前，各大学校普遍采用了教务管理系统，例如美国的 Blackboard、Desire2Learn，加拿大的D2L等[5]。这些教务管理系统涵盖了教学、学生管理、成绩管理、课程表管理、教育资源共享等多个方面，并且不断推出新功能以满足用户的需求。例如，Desire2Learn采用了人工智能技术，通过分析学生的学习数据和行为模式，为学生提供个性化的学习建议和课程推荐[6]。此外，许多国外的教务管理系统还可以和第三方应用程序进行集成，例如和邮件系统、语音消息系统、视频会议系统等进行集成，提高了系统的可用性和用户满意度[7]。

总的来说，教务管理系统在国内外的研究和应用都已经相当成熟，不断地推出新功能和新技术，以适应学生管理的需求和时代的发展[8]。随着人工智能、大数据等技术的不断发展，教务管理系统也将在未来得到更加广泛的应用。

1.3 研究内容及方法

教务管理系统的研究内容通常包括以下方面：

教务管理系统是一种用于高校或学校管理教务工作的软件系统。通过面向对象对教务管理系统进行了研究，内容及方法如下将一一概述。

1. 教务管理系统需求分析：对教务管理工作的需求进行详细分析，包括班级管理、学生管理、教师管理、成绩管理、科目管理等方面。通过与教务部门的沟通、文献调研和实地观察，明确系统所需的功能模块、业务流程和用户需求。
2. 教务管理系统设计：基于需求分析的结果，进行系统的设计工作。包括系统架构设计、数据库设计、用户界面设计等。在设计过程中，可以运用UML建模工具进行系统建模，如用例图、类图、活动图等，以明确系统的结构和交互方式。
3. 教务管理系统开发：根据设计阶段的结果，采用IDEA开发工具和Java、Jsp、Script等技术进行系统的开发。这包括编写代码、实现功能模块、集成测试和性能优化等。
4. 教务管理系统实施与部署：完成系统开发后，进行系统的实施与部署。这包括系统的安装、配置和数据迁移等。
5. 教务管理系统改进：通过用户调查、系统日志分析方法，收集反馈信息，发现问题和改进空间，进一步完善系统。

总之，通过这些研究内容的深入探讨，可以为学校管理提供更加高效和便利的信息化管理手段。

1.4 相关技术简介

B/S结构:即 Browser/Server(浏览器/服务器)结构，是随着Internet 技术的兴起，对C/S结构的一种变化或者改进的结构。在这种结构下，用户界面完全通过 WWW浏览器实现，一部分事务逻辑在前端实现，但是主要事务逻辑在服务器端实现。B/S结构，主要是利用了不断成熟的WWW浏览器技术，结合浏览器的多种 Script语言(VBScript，JavaScript...)和 ActiveX技术，用通用浏览器就实现了原来需要复杂专用软件才能实现的强大功能，并节约了开发成本，是一种全新的软件系统构造技术[9]。随着Windows 98/Windows 2000将浏览器技术植入操作系统内部，这种结构更成为当今应用软件的首选体系结构。显然B/S结构应用程序相对于传统的C/S结构应用程序将是巨大的进步[10]。

在前端技术方面，该系统采用了响应式设计，可以适应不同大小的屏幕，提高了用户体验。同时，该系统主要采用超文本标记语言、层叠样式表、JS技术来实现。超文本标记语言和层叠样式表用于界面的设计和布局，JS实现了前端页面的动态效果。

在后端技术方面，该系统采用了Spring框架和SpringMVC架构，Spring框架提供了丰富的功能模块和组件，如事务管理、数据访问、安全性等，可以帮助开发人员快速构建一个可靠的系统。SpringMVC则是一种基于MVC架构的Web框架，它通过将请求映射到控制器并返回视图，简化了Web开发的复杂性[11]。

总之，基于B/S结构的教务管理系统利用现代的前后端技术和框架，实现了快速、灵活、高效的教务信息管理功能，提高了教务信息化管理水平。

第2章 教务管理系统需求分析

2.1 可行性研究分析

2.1.1 技术可行性

软件方面，操作系统采用的是Windows10，开发工具采用的是IDEA，数据库采用MySQL Workbench 8.0 CE，这些软件在教务管理系统的开发中已经被大量应用。教务管理系统基于B/S架构，采用分布式的技术架构[12]。前端采用HTML、CSS、JS技术，后端采用JAVA技术，这些技术都已经比较成熟，可以有效地支持教务管理系统的开发和应用。

2.1.2 操作可行性

该系统界面友好且易于操作，采用简洁而直观的操作界面，信息清晰明了，能够方便用户快速找到所需功能，不需要太多的培训即可开始使用。硬件设备方面，该系统所需要的硬件设备只需要服务器系统和PC机一台以及网络配件，市场有售且性价比高，可以满足系统开发的硬件所需条件。教务管理系统存储着大量的学生信息、课程信息等数据，设置严格的权限控制，设置访问密码等手段，保证了数据的完整性和安全性[13]。教务管理系统针对不同用户的需求，设立了不同的功能模块，例如学生可以进行信息修改、成绩查询等操作，老师可以进行学生、成绩录入等操作，管理员可以进行系统维护、数据备份等操作，满足了不同用户的需求。所以教务管理系统的操作可行性非常高，可以方便学校和用户进行信息化管理和服务，提高教育教学的效率和质量。

2.1.3 经济可行性

教务管理系统是高校追求信息化建设的必然选择，其建设的经济可行性是系统实施的必要条件。建设教务管理系统的投资相比于传统管理方式，虽然有一定的成本，但对学校的管理决策带来的效益将远远超过成本带来的负担。教务管理系统的建设可以大幅度提升学校的管理效率，降低教学管理和操作成本。教务管理系统的研发与实施需要一定的资金投入，包括软硬件设施、技术人员培训等方面。不过，随着科技的不断进步和技术的普及，系统的建造和维护成本也在逐步降低。所以务管理系统在经济可行性方面具备一定的优势，可以带来企业和用户之间的双赢，既能提高效率，还能节约成本，符合市场需要。

2.2 业务需求

2.2.1 业务范围

在教务管理系统中，业务范围是指该系统需要涉及到的各项业务及其处理范围。需要实现对学生教师、学院专业、科目信息等的录入、修改、查询和删除等操作。一般来说，范围包括以下几个方面，教师管理是指对学校教师的管理。它包括班级分配、课程分配等信息。教务管理系统需要实现对教师信息的基础操作。权限管理能够确保只有授权人员可以访问和修改学生以及教师信息，从而保护学生和教师信息的安全性和完整性。年级管理是对年级信息的管理，在学院和专业下添加年级。班级管理是对班级信息的管理，需要选择专业等信息，再输入班号，自动生成班级编号。科目管理是对科目信息的管理，可以通过选择相应的学院、专业、年级去添加科目，也可以通过选择学院、专业、年级等去获取相关信息。学生管理是对学生的基础信息进行管理，包括可以对学生信息进行基础操作以及转班。成绩管理是指对学生各门课程的分数进行管理，包括分数基础操作。教务管理系统需要实现对学生分数的统计等功能。

综上所述，教务管理系统涵盖了学校教学业务的主要方面，系统灵活，能够满足不同学校的不同需求。

2.2.2主要业务流程

在使用教务管理系统时，系统的管理员需要将教师的信息添加到系统当中，并且给教师分配角色权限，然后由教师来添加学生信息到系统当中，由系统管理员给学生添加相应权限，不同角色管理员分配不同的权。管理员负责教师信息、学生信息、院信息系、专业信息、年级信息、班级信息、科目信息、成绩信息的管理以及权限的分配。若教师修改学生成绩，则需要提交申请至管理员，由管理员审核，管理员审核通过以后学生的成绩才可以被修改。若教师要修改自身信息，则需要提交申请至管理员，管理员审核通过则教师信息修改成功。教师负责录入学生信息到系统当中以及修改学生成绩提交申请，教师可以修改自己的信息，需要管理审核通过后才可以被修改。学生只能查看科目和科任教师信息以及成绩，学生若要修改信息，则需要将修改的信息提交至教师，由教师审核通过方可修改。

本系统的主要业务流程图如图2.1所示。



图2.1 系统主要业务流程图

2.3 功能需求

2.3.1 用例确定

在信息管理系统的初始开发阶段当中，一般是采用UML统一建模语言对信息管理系统的用例来进行建模，再根据建模转化成相应的的开发语言代码。经过对信息管理系统的需求数据分析，本文采用统一建模语言UML对本系统开发为背景的系统用例进行模型构建。通过分析用户的需求，确定系统的外部实体有系统管理员，教师和学生。

通过分析系统的外部实体，并且结合外部实体对信息管理系统功能的需求，便可以确定系统的用例。最后可以确定管理员对应用例的最高权限管理人员，教师对应用例为教师，学生用例对应用例为学生。

综上所述，系统的总用例如图2.2所示。



图2.2 系统总用例图

2.3.2 用例图

1. 系统用户登录用例图

系统用户需要登录系统后才能够操作系统，使用相关模块的功能。管理员、教师、学生都需要输入自己的账号、登录密码和验证码，账号、密码和验证码都输入无误后才能够成功登录系统进行相关功能模块操作。图2.3为系统的用户登录用例图。



图2.3 系统用户登录用例图

1. 管理员用例图

管理员用例包括教师管理、学生管理、学院管理、专业管理、年级管理、班级管理、科目管理、成绩管理、权限管理。管理员用户登录需要输入管理员账号、密码和验证码进行登录，账号、密码、验证码都输入无误后才可以进入管理员模块进行相关操作。图2.4为系统的管理员用例图。



图2.4 管理员用例图

1. 教师功能模块用例图

教师界面包括教师管理、学生管理、成绩管理。教师用户登录需要输入教师账号、密码和验证码进行登录，账号、密码、验证码都输入无误后才可以进入教师界面进行相关操作。图2.5为系统的教师功能模块用例图。

  
图2.5 教师功能模块用例图

1. 学生功能模块用例图

学生界面包括学生管理、科目管理、成绩管理。学生用户登录需要输入学生账号、密码和验证码进行登录，账号、密码、验证码都输入无误后才可以进入学生界面进行学生信息修改、科目查询和成绩查询。图2.6为系统的学生功能模块用例图。



图2.6 学生功能模块用例图

2.3.3 用例描述

（1）用户登录相关的需求用例描述如表2.1所示

表2.1 用户登录用例表

|  |  |  |
| --- | --- | --- |
| 描述项 | 说明 | |
| 用例名称 | 用户登录 | |
| 标识符 | 201 | |
| 用例描述 | 描述了用户登录系统需要的整个过程。用户需要输入正确的账号、密码和验证码方可登录。 | |
| 参与者表 | | 管理员、教师、学生 |
| 前置条件 | | 用户需要输入正确的用户名、密码和验证码 |
| 后置条件 | | 系统给出登录成功的提示 |
| 基本操作流程 | | 系统用户输入用户名、密码和验证码，进行登录验证，验证成功后进入系统。 |
| 异常流程 | | 用户未注册，登录失败。用户名错误，登录失败。密码错误，登录失败验证码错误，登录失败。网络异常，登录失败。服务端口未启动，登录失败。 |

（2）教师管理的需求用例描述如表2.2所示

表2.2 教师管理用例表

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 教师管理 |
| 标识符 | 202 |
| 用例描述 | 描述了教师管理需要的整个过程。 |
| 参与者表 | 管理员、教师 |
| 前置条件 | 必须使用教师或者管理员账号登录进行操作 |
| 后置条件 | 系统给出操作成功的提示 |
| 基本操作流程 | 进入教师管理模块，选择管理教师子模块，可以对教师信息进行删除、修改、和查询以及对教师分配班级。选择添加教师用户子模块，可以添加新的教师用户。 |
| 异常流程 | 用户名未注册，登录失败。用户名和密码错误，登录失败。验证码错误，登录失败。非教师用户登录，无相关菜单权限。 |

（3）学生管理的需求用例描述如表2.3所示

表2.3 学生管理例表

|  |  |  |
| --- | --- | --- |
| 描述项 | 说明 | |
| 用例名称 | 学生管理 | |
| 标识符 | 203 | |
| 用例描述 | 描述了学生乏力需要的整个过程。 | |
| 参与者表 | 管理员、教师、学生 | |
| 前置条件 | | 用户需要输入正确的用户名和密码以及验证码 |
| 后置条件 | | 系统给出操作成功的提示 |
| 基本操作流程 | | 进入学生管理模块，选择管理学生子模块，可以对学生信息进行删除、修改、和查寻以及转班。选择添加学生用户子模块，可以添加新的学生用户。 |
| 异常流程 | | 用户名未注册，用户名和密码错误，登录失败。验证码错误，登录失败。  信息填写格式不正确，操作失败。 |

（4）班级管理的需求用例描述如表2.4所示

表2.4 班级管理用例表

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 班级管理 |
| 标识符 | 204 |
| 用例描述 | 描述了班级管理需要的整个过程。 |
| 参与者表 | 管理员、教师 |
| 前置条件 | 教师或者管理员账号登录进行操作 |
| 后置条件 | 系统给出操作成功的提示 |
| 基本操作流程 | 进入班级管理模块，选择管理班级子模块，可以对班级信息进行删除、修改、和查寻以及转级。选择添加班级信息子模块，可以添加新的班级。 |
| 异常流程 | 用户名未注册，登录失败。用户名和密码错误，登录失败。验证码错误，登录失败。非管理员用户登录，无相关菜单权限。 |

（5）科目管理的需求用例描述如表2.5所示

表2.5 科目管理用例表

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 科目管理 |
| 标识符 | 205 |
| 用例描述 | 描述了科目管理需要的整个过程。 |
| 参与者表 | 管理员、教师、学生 |
| 前置条件 | 用户需要输入正确的用户名和密码以及验证码 |
| 后置条件 | 系统给出操作成功的提示 |
| 基本操作流程 | 进入科目管理模块，选择管理科目子模块，可以对科目信息进行添加、搜索和删除以及分配年级教师。选择查看科任老师子模块，可以查看科任老师信息。 |
| 异常流程 | 用户名未注册，登录失败。用户名和密码错误，登录失败。验证码错误，登录失败。信息填写有误，操作失败。 |

（6）成绩管理的需求用例描述如表2.6所示

表2.6 成绩管理用例表

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 成绩管理 |
| 标识符 | 205 |
| 用例描述 | 描述了成绩管理需要的整个过程。 |
| 参与者表 | 管理员、教师、学生 |
| 前置条件 | 用户需要输入正确的用户名和密码以及验证码 |
| 后置条件 | 系统给出操作成功的提示 |
| 基本操作流程 | 进入成绩管理模块，选择管理成绩子模块，可以对成绩信息进行查询、修改和统计以及排序。选择添加成绩子模块，可以添加学生成绩。选择成绩审核子模块，可以对教师提交的成绩修改进行审核。 |
| 异常流程 | 用户名未注册，登录失败。用户名和密码错误，登录失败。验证码错误，登录失败。输入的信息不正确，操作失败。 |

2.4 非功能需求

非功能需求是指软件系统除了功能需求之外的其他需求，描述了系统应该具备的性能、可靠性、安全性、交互性等方面的要求。以下是关于教务管理系统的非功能需求：

1. 性能：教务管理系统应该具备良好的性能，能够处理大量的数据和并发请求。系统应该能够快速响应用户的操作，并保持良好的响应时间，避免因系统延迟而导致用户等待或无法正常使用系统。
2. 可扩展性：系统应该具备可扩展性，能够满足未来业务发展的需求。系统应该能够支持增加新的功能模块、处理更多的用户和数据量，而不需要对系统进行较大的修改。
3. 可靠性：教务管理系统应该具备高可靠性，能够持续稳定地运行。系统应该具备容错机制，能够处理异常情况并及时恢复。系统应该具备数据备份和恢复功能，以防止数据丢失或损坏。
4. 安全性：系统应该具备良好的安全性，保护用户的隐私和数据安全。系统应该采用合适的身份认证和权限控制机制，确保只有授权用户可以访问系统的敏感信息和功能。系统应该具备防止恶意攻击和数据泄露的能力。
5. 交互性：系统应该具备良好的可用性，易于学习和使用。用户界面应该直观友好，符合用户的使用习惯。系统应该提供清晰的操作指南和帮助文档，以便用户能够快速上手和解决问题。
6. 可维护性：系统应该具备良好的可维护性，方便进行系统的更新和维护工作。系统的代码应该具备良好的结构和文档，便于开发人员进行修改和扩展。系统应该支持日志记录和错误监控，以便及时发现和解决问题。
7. 可移植性：系统应该具备良好的可移植性，能够在不同的操作系统和硬件环境中运行。系统的实现应该遵循开放的标准和规范，尽量减少对特定平台的依赖。

第3章 教务管理系统概要设计

3.1 系统总体结构设计

3.1.1设计目标

系统有管理员、教师、学生三种角色并为不同的角色分配不同权限，以此来管理教师信息、学生信息、学院信息、专业信息、年级信息、班级信息、科目信息、成绩信息。通过不同的模块来规范教师信息、学生信息、学院信息、专业信息、年级信息、班级信息、科目信息、成绩信息，提高工作效率，方便信息提取查阅。

3.1.2 功能模块结构图

通过对本系统的业务和用户需求的分析，本系统设计一共有十大功能模块，分别为教师管理、学生管理、学院管理、专业管理、年级管理、班级管理、科目管理、成绩管理、权限管理。每一个模块都包含了多个的子功能模块，每个模块只有对应的用户才能操作和使用。如图3.1为系统功能设计结构图。



图3.1 系统功能模块图

3.2 系统功能设计

3.2.1 架构描述

本系统是基于B/S的动态网页信息管理系统，本系统的开发包括后台数据库的建立、连接和维护以及前台网络页面的开发两大方面。由于Java的跨平台性、安全性等诸多优势，本系统前端采用HTML、CSS、JavaScript、jQuery和LayUI，后端采用SpringBoot+MyBatis，并使用MySQL作为数据库引擎。系统整体架构图如图3.2所示。



图3.2 系统架构图

3.2.2 登录模块时序图

在登录模块中，系统用户通过登录系统可以完成相关操作，每位用户在登录时都需要进入登录界面当中，登录界面会去判断当前用户是否勾选了记住密码，如上次登录时勾选了记住密码则会自动录入密码进行登录，若没有勾选则需要用户自己输入自己的账户和密码，并填写正确的验证码，均填写无误后，数据会提交到后台，根据用户名和密码在数据库中进行匹配，匹配成功后，才能登录成功。系统分为三种用户，分别是学生、教师和管理员，不同用户对应不同的权限，登录后有不同的功能模块。登录模块时序图如图3.3所示。



图3.3 登录模块时序图

3.2.3 教师管理模块时序图

教师管理模块是只有管理员和教师用户有权限进行操作的模块，管理员模块的教师管理模块分为管理教师子模块和添加教师用户子模块，管理员可以在管理教师子模块中查询、修改、删除教师的信息，也可以为教师分配年级、班级、科目。可以在添加教师子模块下添加教师用户。教师在教师管理模块教师只能修改自身信息提交管理员审核。这里仅以添加教师为例添加时序图，教师管理模块添加时序图如图3.4所示



图3.4 教师管理模块添加时序图

3.2.4 学生管理模块时序图

在学生管理模块中，教师和管理员拥有相同的权限，都可以在管理学生子模块中查询、修改和删除学生信息以及给学生转班，可以在添加学生子模块中添加学生信息。学生用户权限就只能查看和修改自己的信息。这里仅以查询学生信息为例添加时序图，学生查询信息时序图如图3.5所示。



图3.5 学生查询信息时序图

3.2.5 班级管理模块时序图

班级管理模块是只有管理员和教师有权限进行操作的模块，班级管理模块的管理分为管理班级子模块和添加班级信息子模块，管理员可以在管理班级子模块中查询、修改和删除班级的信息以及给班级转级。可以在添加班级信息子模块下添加班级信息。这里仅以删除班级信息为例添加时序图，删除班级时序图如图3.6所示。



图3.6 删除班级时序图

3.2.6 科目管理模块时序图

科目管理模块是只有管理员和教师用户有权限进行操作的模块，管理员或教师用户可以在管理科目子菜单下查看、添加、删除科目信息以及给科目分配年级、班级和教师。这里仅以科目添加为例添加时序图，科目添加时序图如图3.7所示。



图3.7 科目添加时序图

3.2.7 成绩管理模块时序图

成绩管理模块不同角色界面权限不同，管理员拥有最大权限，可以查看学生科目成绩、成绩排名、成绩统计图和录入学生成绩以及学生成绩修改审核。教师用户界面可以查看、录入学生成绩，修改学生成绩需要向管理员提交申请。学生用户界面只能查看成绩。这里仅以成绩修改为例添加时序图，成绩修改时序图如图3.6所示。



图3.8 成绩修改时序图

3.3 系统数据库设计

3.3.1数据库概念设计

概念设计是数据库设计的核心环节[14]。通过对用户需求进行综合、归纳与抽象形成一个独立具体DBMS的概念模型[15]。以下是本系统主要实体的列举和对应的E-R图。

1. 用户实体

学生用户实体包括姓名、账号、密码、性别、年龄、学院名称、专业名称、所在年级和所在班级、联系电话、QQ号、家庭地址。图3.9为学生用户实体属性图。



图3.9 学生用户实体属性图

教师用户实体包括姓名、账号、密码、性别、年龄、学历、学院名称、专业名称、擅长科目、QQ号码、邮箱地址、家庭地址、个人简介。图3.10为教师用户实体属性图。



图3.10 教师用户实体属性图

管理员用户实体包括账号、ID、密码。图3.11为管理员用户实体属性图。



图3.11 管理员用户实体属性图

1. 角色实体

角色实体包括用户ID、用户账号、用户密码、和用户状态ID、用户名。图3.12为角色实体属性图。



图3.12 角色实体属性图

1. 学院实体

学院实体包括学院ID、学院名、状态、备注。图3.12为学院实体属性图。



图3.13 学院实体属性图

1. 专业实体

专业实体包括ID、学院名、专业名、状态、专业备注。图3.14为专业实体属性图。



图3.14 专业实体属性图

1. 年级实体

年级实体包括年级ID、年级编号、年级名、创建人、创建时间。图3.15为年级实体属性图。



图3.15 年级实体属性图

1. 班级实体

班级实体包括班级ID、编号、班级名、创建人、创建时间、年级ID、科目列表。图3.16为班级实体属性图。



图3.16 班级实体属性图

1. 成绩实体

成绩实体包括成绩ID、科目ID、学生ID、成绩、年级名称、班级名称、学生学号、学生姓名、提交修改教师ID、状态、修改成绩值。图3.17为成绩实体属性图。



图3.17 成绩实体属性图

1. 科目实体

科目实体包括科目ID、科目名称、创建人、创建时间、年级ID、年级名、学院ID。图3.18为科目实体属性图。



图3.18 科目实体属性图

通过对以上系统当中主要实体E-R图的分析，得到系统的主要E-R图。图3.19为系统主要E-R模型图。



图3.19 系统E-R图

3.3.2数据库逻辑设计

数据库物理设计，包括选择数据库产品，确定数据库实体属性（字段）、数据类型、长度、精度确定、DBMS页面大小等[16]。数据库逻辑设计决定了数据库及其应用的整体性能，调优位置。如果数据库逻辑设计不好，则所有调优方法对于提高数据库性能的效果都是有限的[17]。根据从上节中系统主要E-R模型图的分析，根据从概念模型向关系模型转换的原则（即，实体转换成关系模型，联系也要转换成关系模型），具体如下。

根据实体转换成关系模式原则，可以得到以下的9个关系模式，管理员由后台直接录入数据库，所以以下无管理员关系模式。有下划线的是码。

1. 学生用户：学生ID、姓名、账号、密码、性别、年龄、学院名称、专业名称、所在年级、所在班级、联系电话、QQ号、家庭地址。
2. 教师用户：教师ID、姓名、账号、密码、性别、年龄、学历、学院名称、专业名称、擅长科目、QQ号码、邮箱地址、家庭地址、个人简介。
3. 角色：用户ID、用户账号、用户密码、和用户状态ID、用户名
4. 专业：专业ID、学院名、专业名、状态、专业备注
5. 年级：年级ID、年级编号、年级名、创建人、创建时间
6. 班级：班级ID、编号、班级名、创建人、创建时间、科目列表
7. 成绩：成绩ID、成绩、年级名称、班级名称、学生学号、学生姓名、提交修改教师ID、状态、修改成绩值
8. 科目：科目ID、科目名称、创建人、创建时间、年级名
9. 学院：学院ID、学院名称、状态、备注

根据每个联系完成转换，有下划线的是码，有下划波浪线是外码。  
（一）一对一（1:1）

1、成绩科目联系

一个学生的一个科目只能有一个成绩，得到关系模式：成绩（成绩ID、科目ID、学生ID、成绩、年级名称、班级名称、学生学号、学生姓名、提交修改教师ID、状态、修改成绩值）

1. 一对多（1:n）

1、学生成绩联系

一个学生有不同的科目，所以一个学生对应多个成绩，得到关系模式：成绩（学生ID、科目ID、成绩ID、成绩、年级名称、班级名称、学生学号、学生姓名、提交修改教师ID、状态、修改成绩值）。

2、科目年级联系

一个年级可以有多个科目，得到关系模式：年级（年级ID、科目名称、创建人、创建时间、科目ID、年级名、学院ID）

3、专业学院联系

一个学院可以有多个专业，得到关系模式：学院（学院ID、学院名、专业ID、专业名、状态、专业备注）

4、科目学院联系

一个学院可以拥有多个科目，得到关系模式：学院（学院ID、科目名称、创建人、创建时间、年级ID、年级名、科目ID）

5、班级年级联系

一个年级可以有多个班级，得到关系模式：年级（年级ID、编号、班级名、创建人、创建时间、班级ID、科目列表）  
（三）多对多（n：m）

1、教师学生联系

一个教师可以教多个学生，一个学生也可以有多个教师教。取各实体的码得到关系模式：用户角色（教师ID，学生ID）

2、教师学院联系

一个教师可以在多个学院任教，一个学院也可以有多个教师任教。取各实体的码得到关系模式：教师（教师ID，学院ID，学院名称）

3、教师专业联系n:m

一个教师可以教多个专业，一个专业也可以有多个教师任教。取各实体的码得到关系模式：教师（教师ID，专业ID，专业名称）

4、教师班级联系n:m

一个教师可以教多个班级，一个班级有多个教师任教。取各实体的码得到关系模式：教师（教师ID，班级ID）

5、教师科目联系n:m

一个教师可以教多个科目，一个科目也可以有多个教师。取各实体的码得到关系模式：教师（教师ID，科目ID）

6、教师年级联系n:m

一个教师可以任教多个年级，一个年级也可以有多个教师任教。取各实体的码得到关系模式：教师（教师ID，年级ID）

7、学生科目联系n:m

一个学生要学习多个科目，一个科目有多个学生学习。取各实体的码得到关系模式：学生（学生ID，科目ID）

8、专业年级联系n:m

一个专业有多个年级，一个年级也有多个专业。取各实体的码得到关系模式：专业（专业ID，年级ID）

9、科目年级联系n:m

一个科目有不同年级的学生一起上，一个年级有不同的科目。取各实体的码得到关系模式：科目（科目ID，年级ID，年级名）

（四）所以最终得到的关系模式有以下9个：

（1）学生用户：学生ID、姓名、账号、密码、性别、年龄、学院名称、专业名称、所在年级、所在班级、联系电话、QQ号、家庭地址。

（2）教师用户：教师ID、姓名、账号、密码、性别、年龄、学历、学院名称、专业名称、擅长科目、QQ号码、邮箱地址、家庭地址、个人简介。

（3）角色：用户ID、用户账号、用户密码、用户状态ID、用户名

（4）专业：专业ID、学院名、专业名、状态、专业备注

（5）年级：年级ID、年级编号、年级名、创建人、创建时间

（6）班级：班级ID、编号、班级名、创建人、创建时间、年级ID、科目列表

（7）成绩：成绩ID、科目ID、学生ID、成绩、年级名称、班级名称、学生学号、学生姓名、提交修改教师ID、状态、修改成绩值

（8）科目：科目ID、科目名称、创建人、创建时间、年级ID、年级名、学院ID

（9）学院：学院ID、学院名称、状态、备注

3.3.3数据库的实现

本系统数据库管理系统是MySQL关系型数据库系统。以下是本系统数据库所需要创建的数据库表。

1. 角色（Character）的字段包括状态ID、角色。角色表如表3.1所示。

表3.1 角色表 （Character）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 是否可为空 | 描述 |
| StateID | Int | 11 | 是 | 否 | 状态ID |
| Info | Varchar | 45 | 否 | 否 | 角色 |

1. 班级（Class）主要的字段包括班号、班级名称等。班级表如表3.2所示。

表3.2 班级（Class）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 是否可为空 | 描述 |
| ID | int | 11 | 是 | 否 | 班级ID |
| ClassCode | Varchar | 45 | 否 | 否 | 班级号 |
| CassName | Varchar | 45 | 否 | 否 | 班级名 |
| CreateMessage | Varchar | 45 | 否 | 否 | 创建人 |
| CreateTime | Datetime | 19 | 否 | 否 | 创建时间 |
| GradeID | Int | 11 | 否 | 否 | 年级ID |
| cID | Int | 11 | 否 | 否 | 学院ID |
| mID | Int | 11 | 否 | 否 | 专业ID |

1. 学院（College）主要字段包括学院ID、名字等。学院表如表3.3所示。

表3.3 学院表（College）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 是否可为空 | 描述 |
| CID | Int | 11 | 是 | 否 | 学院ID |
| Cname | Varchar | 255 | 否 | 是 | 学院名称 |
| Cremark | Varchar | 255 | 否 | 是 | 学院描述 |
| Cstatus | Varchar | 255 | 否 | 是 | 学院状态 |

1. 年级（Grade）中主要的字段包括年级ID、年级编号、年级名称、创建人、创建时间、学院ID、专业ID。年级表如表3.4所示。

表3.4 年级表（Grade）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 是否可为空 | 描述 |
| ID | Int | 11 | 是 | 否 | 年级ID |
| GradeCode | Int | 11 | 否 | 是 | 年级编号 |
| GradeName | Varchar | 45 | 否 | 是 | 年级名称 |
| CreateMessage | Varchar | 45 | 否 | 是 | 创建人 |
| reateTime | Datetime | 19 | 否 | 是 | 创建时间 |
| CID | Int | 11 | 否 | 是 | 学院ID |
| MID | Int | 11 | 否 | 是 | 专业ID |

1. 登录（Login）中主要的字段包括账号ID、账号、密码、状态ID。登录表如表3.5所示。

表3.5 登录表（Login）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 是否可为空 | 描述 |
| ID | Int | 11 | 是 | 否 | 账号ID |
| Code | Varchae | 45 | 否 | 是 | 账号 |
| Pwd | Varchar | 255 | 否 | 是 | 密码 |
| StateID | Int | 11 | 否 | 是 | 账号角色ID |

1. 专业（Major）中主要的字段包括专业ID、名称等。专业表如表3.6所示。

表3.6 专业表（Major）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 是否可为空 | 描述 |
| MID | Int | 11 | 是 | 否 | 专业ID |
| Mname | Varchar | 255 | 否 | 是 | 专业名称 |
| Mstatus | Varchar | 255 | 否 | 是 | 专业状态 |
| Mremark | Varchar | 255 | 否 | 是 | 专业描述 |
| CID | Int | 11 | 否 | 是 | 学院ID |

1. 成绩（Result）主要字段包括成绩ID、成绩等。成绩表如表3.7所示。

表3.7 成绩表（Result）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 是否可为空 | 描述 |
| ID | Int | 11 | 是 | 否 | 成绩ID |
| SubID | Int | 11 | 否 | 是 | 科目ID |
| sID | Int | 11 | 否 | 是 | 学生ID |
| Time | Datetime | 19 | 否 | 是 | 时间 |
| Result | Double | 22 | 否 | 是 | 成绩 |
| ActualResult | Double | 22 | 否 | 是 | 修改成绩 |
| TID | Varchar | 255 | 否 | 是 | 教师ID |
| Status | Varchar | 255 | 否 | 是 | 状态 |

1. 学生（Student）中主要的字段包括ID、账号、姓名、密码、年龄、性别、QQ、电话、地址等。学生表如表3.8所示。

表3.8 学生表（Student）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 是否可为空 | 描述 |
| ID | Int | 11 | 是 | 否 | 学生ID |
| Code | Varchar | 45 | 否 | 是 | 学生账号 |
| Name | Varchar | 45 | 否 | 是 | 学生姓名 |
| Age | Int | 11 | 否 | 是 | 学生年纪 |
| Sex | Char | 2 | 否 | 是 | 学生性别 |
| Qq | Varchar | 45 | 否 | 是 | 学生qq |
| Phone | Varchar | 45 | 否 | 是 | 电话号码 |
| Address | Varchar | 100 | 否 | 是 | 家庭住址 |
| ClassID | Int | 11 | 否 | 是 | 班级ID |
| cID | Int | 11 | 否 | 是 | 学院ID |
| mID | Int | 11 | 否 | 是 | 专业ID |

1. 科目（Subject）中主要的字段包括创建人、创建时间、年级ID、年级名等。科目表如表3.9所示。

表3.9 科目表（Subject）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 是否可为空 | 描述 |
| ID | Int | 11 | 是 | 否 | 科目ID |
| SubjectCode | Varchar | 45 | 否 | 是 | 科目编号 |
| SubjectName | Varchar | 45 | 否 | 是 | 科目名称 |
| CreateMessage | Varchar | 45 | 否 | 是 | 创建人 |
| CreateTime | Datetime | 19 | 否 | 是 | 创建时间 |
| GradeID | Int | 11 | 否 | 是 | 年级ID |
| CID | Int | 11 | 否 | 是 | 学院ID |
| MID | Int | 11 | 否 | 是 | 专业ID |

1. 教师（Teacher）主要字段包括学历、邮箱等。教师表如表3.10所示。

表3.10 教师表（Teacher）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 是否可为空 | 描述 |
| ID | Int | 11 | 是 | 否 | 教师ID |
| Code | Varchar | 45 | 否 | 否 | 账号 |
| Name | Varchar | 45 | 否 | 否 | 名字 |
| Sex | Char | 2 | 否 | 否 | 性别 |
| Age | Int | 11 | 否 | 否 | 年纪 |
| Education | Varchar | 100 | 否 | 否 | 学历 |
| GoodAt | Varchar | 100 | 否 | 否 | 擅长 |
| Phone | Varchar | 45 | 否 | 否 | 电话 |
| Qq | Varchar | 45 | 否 | 否 | Qq |
| Email | Varchar | 45 | 否 | 否 | 邮箱 |
| Address | Varchar | 100 | 否 | 否 | 住址 |
| Introduction | Varchar | 100 | 否 | 否 | 介绍 |
| CID | Int | 11 | 否 | 否 | 学院ID |
| MID | Int | 11 | 否 | 否 | 专业ID |

第4章 详细设计与实现

4.1 登录模块详细设计与实现

4.1.1 登录模块详细设计概述

登录模块分为三种角色登录，分别是学生账号登录、教师账号登录和管理员账号登录。在用户登录过程中，主要通过调用后台接口LoginController类下的Login()方法验证用户输入的账号、密码对应的状态ID，输入随机生成的四位数验证码，点击登录若账号输入无误返回登录信息，进入相应的登录界面。

4.1.2 登录方法程序流程图

通过对上述登录管理模块设计的具体描述，输入账号、密码、验证码添加Login()方法进行程序设计。Login()方法通过接受前端调用指令，接受对输入信息的验证通过后的参数，连接数据库，在数据库登录表中执行查询语句，验证通过，返回登录信息，登录相应界面。图4.1为登录方法程序流程图。



图4.1 登录方法流程图

4.1.3 登录模块界面设计

登录的角色不同，所对应的登录进入的界面功能也不同。图4.2为登录界面设计。

  
图4.2 登录界面设计

4.2 教师管理模块详细设计与实现

4.2.1 教师管理模块详细设计概述

教师管理分为管理教师和添加教师用户两个子功能，管理教师界面中可以通过选择学院名称、专业名称、账号、姓名点击查询，通过调用后台SelectController类下的Select()方法，传TableName参数为Teacher，查询教师信息列表，另外可对教师列表进行修改、删除和分配班级的操作。

添加教师用户在表单页面中输入相关信息后，提交后台，通过调用后台InsertController类的insert方法，传TableName参数为Teacher，即可将新增的教师用户上传到数据库中，教师用户信息添加成功。

4.2.2 教师管理程序流程图

根据上述对本模块的详细设计概述，添加Select()方法、Update()方法、Delete()方法、Insert()方法，Select()方法、Update()方法、Delete()方法、Insert()方法会将相应的信息进行传递。在数据库教师表执行相关操作，执行成功则返回相应信息到前端界面。图4.3为教师添加程序流程图。



图4.3 教师添加程序流程图

4.2.3 教师管理模块界面设计

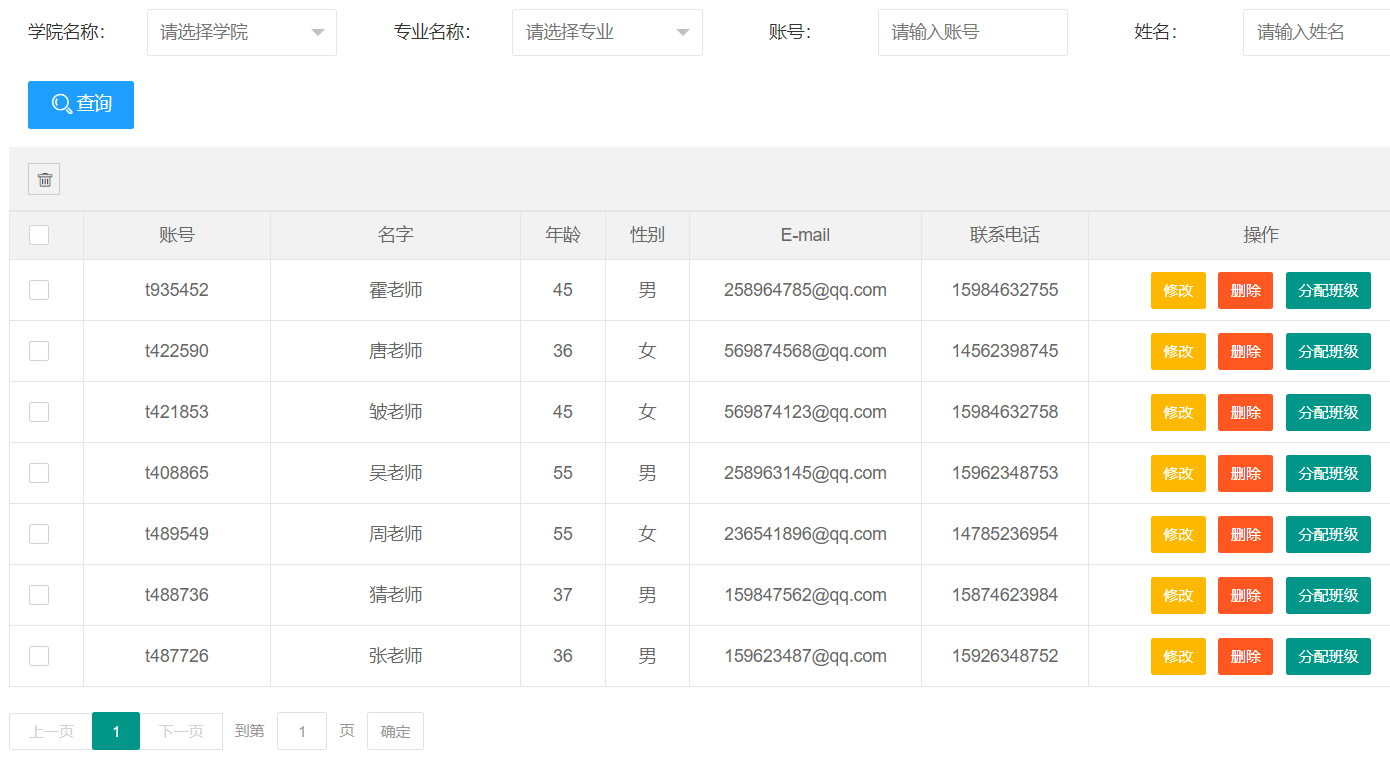
管理教师界面可以通过选择相关信息对教师信息进行相关操作。图4.4为管理教师界面设计。图4.5为添加教师用户界面设计。  


图4.4 管理教师界面设计



图4.5 添加教师用户界面设计

4.3 学生管理模块详细设计与实现

4.3.1 学生管理模块详细设计概述

本模块分为学生查询和添加学生用户两个子功能，学生查询界面中可以通过选择姓名、学号、学院、专业点击查询，通过调用后台SelectController类下的Select()方法，传TableName参数为Student，查询学生信息列表，另外可对学生列表进行修改、删除和转班的操作。

添加学生用户在表单页面中输入相关信息后，提交后台，通过调用后台InsertController类的insert方法，传TableName参数为Student，即可将新增的学生用户上传到数据库中，学生用户信息添加成功。

4.3.2 学生管理方法程序流程图

|  |
| --- |
| 根据上述对本模块内容的详细设计概述，通过添加Select()方法、Update()方法、Delete()方法、Insert()方法，这些方法会将相应的信息进行传递，在数据库学生表执行相关操作，执行成功则返回相应信息到前端界面。图4.6为学生查询程序流程图。 |

  
图4.6 学生查询程序流程图

4.3.3 学生管理模块界面设计

管理学生界面可以通过输入姓名、学号，选择学院名称、专业名称进行搜索，搜索显示该学生信息，可以对该学生信息进行操作。图4.7为管理学生界面设计。

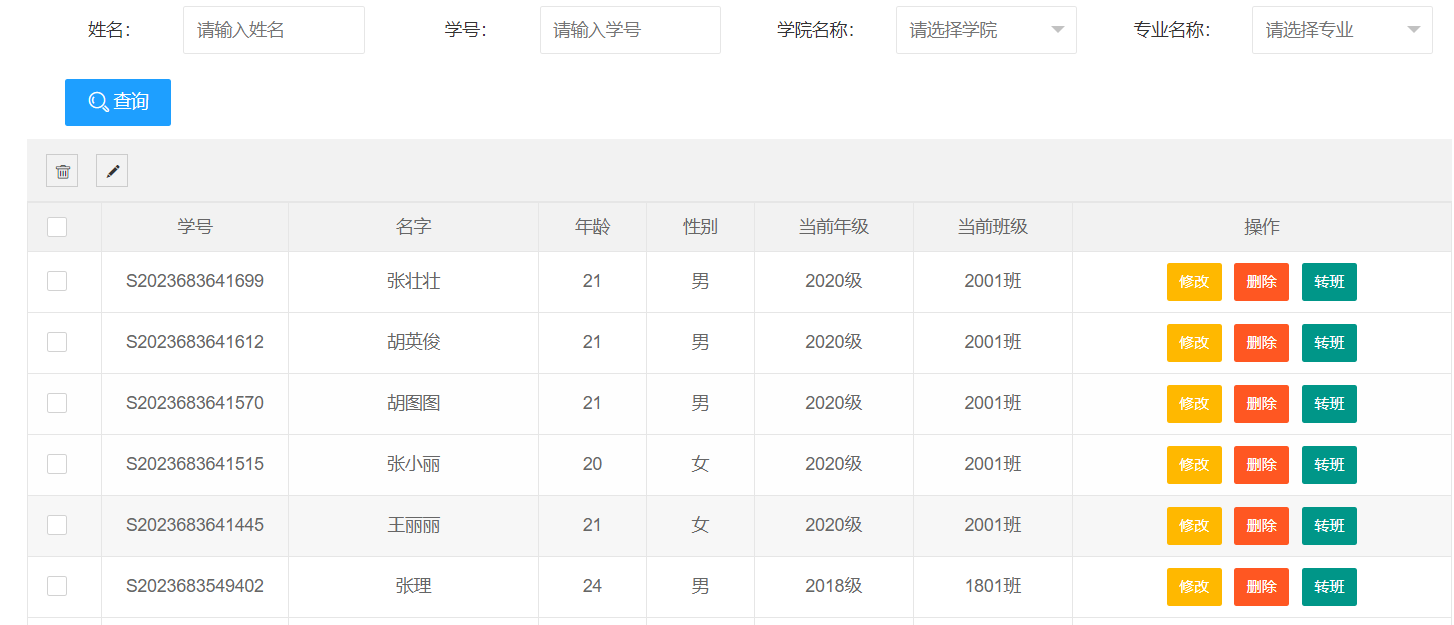


图4.7 管理学生界面设计

4.4 班级管理模块详细设计与实现

4.4.1 班级模块详细设计概述

本模块分为管理班级和添加班级信息用户两个子功能，第一个界面中可以通过选择相关信息点击查询，通过调用后台SelectController类下的Select()方法，传TableName参数为Class，查询班级信息列表，另外可对班级列表进行修改、删除和转级的操作。

添加班级用户在表单页面中输入相关信息后，点击提交，通过调用后台InsertController类的Insert方法，传TableName参数为Class，即可将新增的班级信息上传到数据库中，班级信息添加成功。

4.4.2 班级管理方法程序流程图

|  |
| --- |
| 根据上述对本模块的详细设计概述，通过添加Delete()方法。将相应的数据进行传输，在数据库班级表执行删除，执行成功则返回相应信息到前端界面。图4.8为班级删除程序流程图。 |



图4.8 班级删除程序流程图

4.4.3 班级管理模块界面设计

管理班级界面可以通过选择相关信息对班级信息进行基础操作，点击转级可以给学生转年级。图4.9为管理班级界面设计。图4.10为删除班级信息界面设计。

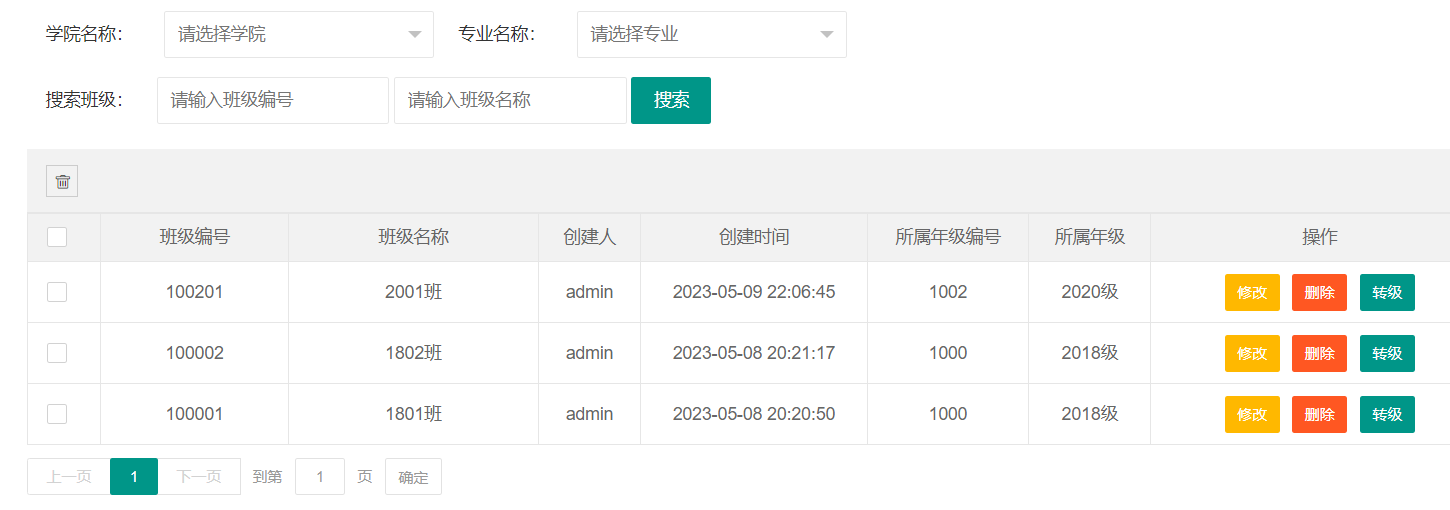


图4.9管理班级界面设计

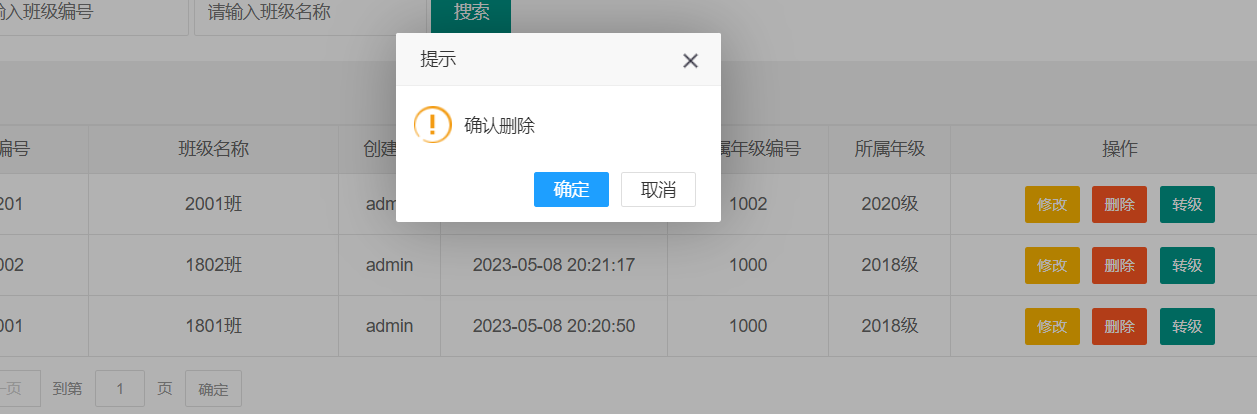


图4.10删除班级信息界面设计

4.5 科目管理模块详细设计与实现

4.5.1 科目管理模块详细设计概述

科目管理分为管理科目和查看科任教师信息两个子功能，管理科目界面中可以通过选择学院名称、专业名称，输入科目编号和科目名称点击搜索，通过调用后台SelectController类下的Select()方法，传TableName参数为Subject，查询科目信息列表，另外可对科目列表进行基础操作。通过选择学院名称、专业名称，输入科目编号和科目名称点击添加科目，通过调用后台增加控制类的增加方法，传TableName参数为Subject，即可将新增的科目信息上传到数据库中，科目信息添加成功。

查看科任教师信息界面中可以通过学院名称、专业名称，输入科目编号和科目名称点击搜索，通过调用后台SelectController类下的Subject()方法，传TableName参数为TeacherClass，查询科任教师信息列表，

4.5.2 科目管理方法程序流程图

|  |
| --- |
| 根据上述对本模块的详细设计概述，添加Select()方法、Delete()方法、Insert()方法，这些方法将相应的数据进行传输，在数据库科目表执行相关操作，执行成功则返回相应信息到前端界面。图4.11为科目添加程序流程图。 |



图4.11科目添加流程图

4.5.3 科目管理模块界面设计

管理科目界面可以通过选择相关信息，输入相应信息进行对应科目信息的查询，可以点击相关按钮进行相关操作。图4.12为管理科目界面设计。



图4.12 管理科目界面设计

4.6 成绩管理模块详细设计与实现

4.6.1 成绩模块详细设计概述

成绩管理分为管理成绩、查看成绩、成绩审核三个子功能，管理成绩界面中可以通过选择年级、班级、科目，输入学号和姓名点击相关按钮，通过调用后台SelectController类下的查询方法，传TableName参数为Grade，查询学生成绩信息列表，另外可对学生成绩进行修改和修改审核。

录入成绩在表单页面中选择年级、班级和科目点击查询再手动输入成绩，点击一键保存，通过调用后台InsertController类的Insert方法，传TableName参数为Grade，即可将新增的成绩信息上传到数据库中，学生成绩信息添加成功。

修改成绩审核界面中可以选择年级、班级、科目，输入学号和姓名点击查询，通过调用后台SelectController类的select方法，传TableName参数为Grade，即可将查看的教师提交的的成绩修改申请。

4.6.2 成绩管理方法程序流程图

|  |
| --- |
| 根据上述对本模块的详细设计概述，添加Select()方法、Update()方法将相应信息进行传输，在数据库成绩表执行查询和修改，执行成功则返回相应信息到前端界面。图4.13为成绩审核程序流程图。 |



图4.13成绩审核程序流程图

4.6.3 成绩管理模块界面设计

查看成绩界面可以通过选择年级、班级、科目，输入学号和姓名点击查询可以查看成绩列表，会自动生成成绩排名和统计。图4.14为查看成绩界面设计。

修改成绩审核界面可以通过选择相关信息点击按钮，就可以看到教师提交的成绩修改申请，点击审核按钮点击同意即可修改成绩。图4.15为 成绩审核界面设计。



图4.14 查看成绩界面设计

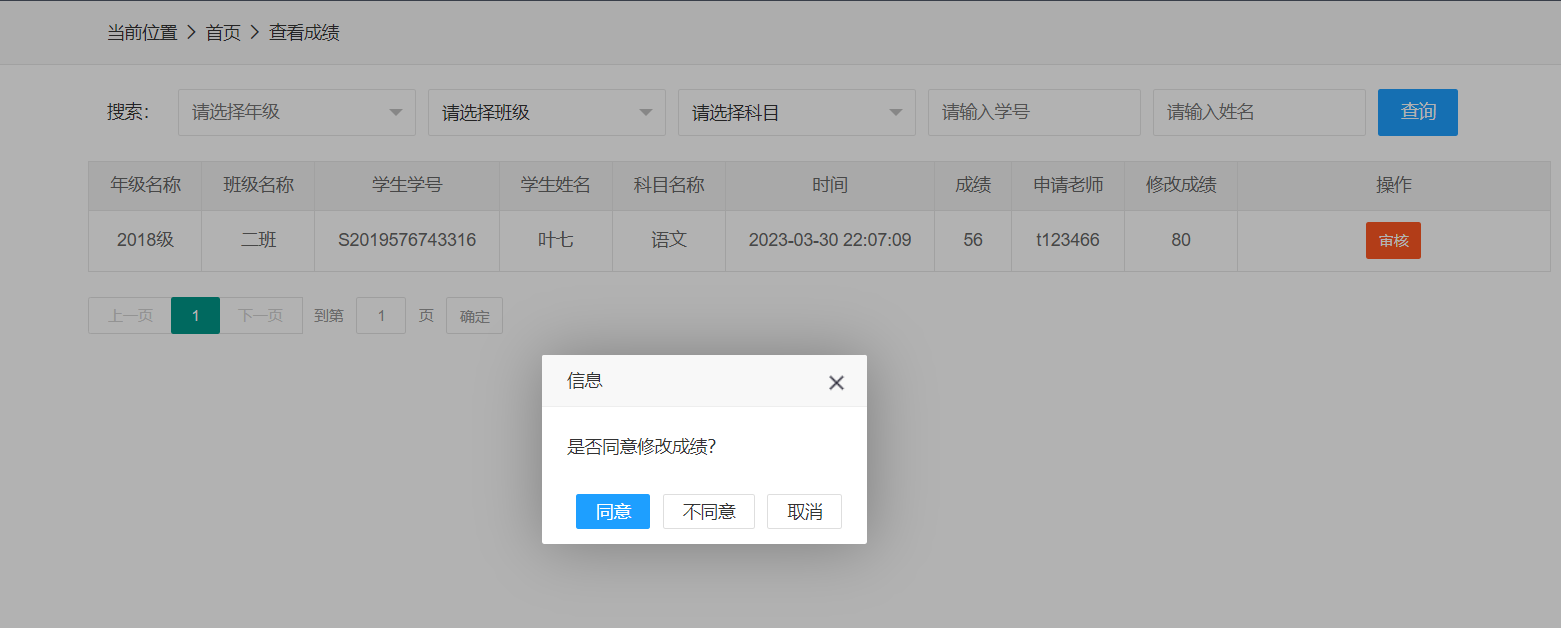


图4.15 成绩审核界面设计

第5章 教务管理系统软件测试

5.1 测试概述

本测试针对教务管理系统进行测试，主要分为黑盒测试和白盒测试[18]。

黑盒测试是检测每个功能是否都能正常使用，在测试中，把程序看作一个不能打开的黑盒子，在完全不考虑程序内部结构和内部特性的情况下，在程序接口进行测试，该方法只检查程序功能是否按照需求规格说明书的规定正常使用，程序是否能适当地接收输入数据而产生正确的输出信息[19]。

白盒测试白盒测试又称结构测试、透明盒测试、逻辑驱动测试或基于代码的测试[20]。

5.2 黑盒测试

5.2.1 教师管理模块测试

教师管理模块功能包括管理教师和添加教师用户两个子功能模块，管理教师模块主要是对教师信息的添加功能点进行黑盒测试，采用等价类划分测试方法。查询采用了模糊查询，输入部分文字或者选择任意一个选项都可以查询出相关教师信息。添加教师用户功能模块，通过输入姓名不能为空，密码必须包含数字和字母6-16位，性别默认为男，年龄不可空，学历不可为空，学院名称不可为空，专业名称不可为空，擅长科目不可为空，联系电话不可为空，QQ号码不可为空，邮箱地址不可为空，家庭地址和个人简介可不填，账号自动生成，点击提交便可以完成教师用户的添加。本测试仅根据教师用户添加功能进行黑盒测试，以下是添加教师用户的等价类情况如表5.1所示。添加教师用户测试数据如表5.2所示，添加教师用户的相关测试用例及结果如表5.3所示。

表5.1 添加教师用户等价类

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试内容 | 有效等价类 | 编号 | 无效等价类 | 编号 |
| 姓名 | 输入名字 | 1 | 未输入 | 2 |
| 性别 | 选择性别 | 3 | 未选择 | 4 |
| 密码 | 输入数字加字母6-16位 | 5 | 只输入数字或字母 | 6 |
| 输入小于六位数 | 7 |
| 输入大于16位数 | 8 |

续表5.1 添加教师用户等价类

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试内容 | 有效等价类 | 编号 | 无效等价类 | 编号 |
| 年龄 | 输入数字 | 9 | 输入文字 | 10 |
| 学历 | 选择 | 11 | 不选择 | 12 |
| 学院名称 | 选择 | 13 | 不选择 | 14 |
| 专业名称 | 选择 | 15 | 不选择 | 16 |
| 擅长科目 | 输入文字 | 17 | 不输入 | 18 |
| 联系电话 | 输入11位数字 | 19 | 不输入 | 20 |
| QQ号码 | 输入6-13位数字 | 21 | 不输入 | 22 |
| 邮箱地址 | 输入正确的邮箱格式 | 23 | 输入为空 | 24 |
| 不输入正确的邮箱格式 | 25 |
| 家庭地址 | 输入 | 25 | 不输入 | 26 |
| 个人简介 | 输入 | 27 | 不输入 | 28 |

表5.2 添加教师用户测试数据

|  |  |
| --- | --- |
| 用例编号 | 测试数据 |
| 1.1 | （张三，男，123456z，55，博士，人工智能与大数据，软件工程，计算机网络，15962348751,258963258，258963258@qq.com，四川眉山，优秀讲师） |
| 1.2 | （ ，男，123456z，55，博士，人工智能与大数据，软件工程，计算机网络，15962348751,258963258，258963258@qq.com，四川眉山，优秀讲师） |
| 1.3 | （张三，男，123456，55，博士，人工智能与大数据，软件工程，计算机网络，15962348751,258963258，258963258@qq.com，四川眉山，优秀讲师） |
| 1.4 | （张三，男，123z，55，博士，人工智能与大数据，软件工程，计算机网络，15962348751,258963258，258963258@qq.com，四川眉山，优秀讲师） |
| 1.5 | （张三，男，123456z1111111111，55，博士，人工智能与大数据，软件工程，计算机网络，15962348751,258963258，258963258@qq.com，四川眉山，优秀讲师） |
| 1.6 | （张三，男，123456z，55， ，人工智能与大数据，软件工程，计算机网络，15962348751,258963258，258963258@qq.com，四川眉山，优秀讲师） |

续表5.2 添加教师用户测试数据

|  |  |
| --- | --- |
| 用例编号 | 测试数据 |
| 1.7 | （张三，男，123456z，55，博士，人工智能与大数据，软件工程，计算机网络，15962348751,258963258，258963258@qq.com，四川眉山， ） |

表5.3 添加教师用户用例设计及结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | 覆盖范围 | 预期输出 | 实际结果 | 测试结果 |
| 1.1 | 1,3,5,9,11,13,15,17,19,21,23,25,27 | 录入教师信息 | 录入教师信息 | 通过 |
| 1.2 | 2,3,5,9,11,13,15,17,19,21,23,25,27 | 未录入教师信息 | 未录入教师信息 | 不通过 |
| 1.3 | 1,3,6,9,11,13,15,17,19,21,23,25,27 | 未录入教师信息 | 未录入教师信息 | 不通过 |
| 1.4 | 1,3,7,9,11,13,15,17,19,21,23,25,27 | 未录入教师信息 | 未录入教师信息 | 不通过 |
| 1.5 | 1,3,8,9,11,13,15,17,19,21,23,25,27 | 未录入教师信息 | 未录入教师信息 | 不通过 |
| 1.6 | 1,3,5,9,12,13,15,17,19,21,23,25,27 | 未录入教师信息 | 未录入教师信息 | 不通过 |
| 1.7 | 1,3,5,9,11,13,15,17,19,21,23，25 | 录入教师信息 | 录入教师信息 | 通过 |

5.2.2 科目管理方法测试

科目管理模块功能包括对管理科目模块和显示科任教师信息两个子功能模块，管理科目模块主要是对科目添加进行黑盒测试，采用等价类划分测试方法。添加科目功能模块，通过输入科目名称，选择学院和专业以及年级，点击提交便可以完成科目的添加。本测试仅根据科目添加功能进行黑盒测试，以下是添加科目的等价类情况如表5.4所示。添加科目测试数据如表5.5所示，添加科目的相关测试用例及结果如表5.6所示。

表5.4 添加科目功能等价类

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试内容 | 有效等价类 | 编号 | 无效等价类 | 编号 |
| 科目名称 | 输入名称 | 1 | 未输入 | 2 |
| 学院 | 已选用 | 3 | 未选用 | 4 |
| 专业 | 已选用 | 5 | 未选用 | 6 |
| 年级 | 已选用 | 7 | 未选用 | 8 |

表5.5 添加科目测试数据

|  |  |
| --- | --- |
| 用例编号 | 测试数据 |
| 1.1 | （软件工程，人工智能与大数据，软件工程，2018级） |
| 1.2 | （ ，人工智能与大数据，软件工程，2018级） |
| 1.3 | （软件工程， ，软件工程，2018级） |
| 1.4 | （软件工程，人工智能与大数据， ，2018级） |
| 1.5 | （软件工程，人工智能与大数据，软件工程， ） |

表5.6 添加科目功能用例设计及结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | 覆盖范围 | 预期输出 | 实际结果 | 测试结果 |
| 1.1 | 1,,3,5,7 | 录入科目信息 | 录入科目信息 | 通过 |
| 1.2 | 2,,3,5,7 | 未录入科目信息 | 未录入科目信息 | 不通过 |
| 1.3 | 1,,4,5,7 | 未录入科目信息 | 未录入科目信息 | 不通过 |
| 1.4 | 1,,3,6,7 | 未录入科目信息 | 未录入科目信息 | 不通过 |
| 1.5 | 1,,3,5,8 | 未录入科目信息 | 未录入科目信息 | 不通过 |

5.2.3 成绩管理方法测试

成绩管理模块功能包括查看、录入、审核三个功能模块，成绩管理模块主要是对录入成绩进行黑盒测试，采用等价类划分测试方法。录入成绩功能模块，通过输入成绩，点击一键保存便可以完成成绩录入。本测试仅根据录入成绩功能进行黑盒测试，录入成绩的等价类情况如表5.7所示。录入成绩的相关测试用例及结果如表5.8所示。

表5.7 录入成绩功能等价类

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试内容 | 有效等价类 | 编号 | 无效等价类 | 编号 |
| 成绩录入 | 数字 | 1 | 符号 | 2 |
| 文字 | 3 |
| 字母 | 4 |

表5.8 录入成绩功能用例设计及结果

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 用例编号 | 覆盖范围 | 测试数据 | 预期输出 | 实际结果 | 测试结果 |
| 1.1 | 1 | 88 | 录入成绩 | 录入成绩 | 通过 |
| 1.2 | 2 | ！ | 未录入成绩 | 未录入成绩 | 不通过 |
| 1.3 | 3 | 星 | 未录入成绩 | 未录入成绩 | 不通过 |
| 1.4 | 4 | z | 未录入成绩 | 未录入成绩 | 不通过 |

5.3 白盒测试

5.3.1 教师管理模块测试

本模块对添加教师用户功能进行白盒测试。如图5.1所示。

通过基本路径的测试方法，可以确定环路复杂度V(G) = 2。

根据环路复杂度可以得到以下几条路径，教师管理白盒测试如表5.9所示。

表5.9 教师管理白盒测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 路径 | 测试用例 | 期望结果 | 运行结果 |
| 1 | 1-2-3-4-5-6 | 均填写合法的信息，点击提交按钮。 | 教师用户添加成功。 | 教师用户添加成功。 |
| 2 | 1-2-4-5-6 | 有内容没有填写合法的信息，点击提交按钮。 | 教师用户添加失败。 | 教师用户添加失败。 |



图5.1 添加教师用户方法程序流程图（左）及控制流图（右）

5.3.2 科目管理方法测试

本模块对添加科目功能进行白盒测试.。如图5.2所示。

通过基本路径的测试方法，可以确定环路复杂度V(G) = 2。

根据环路复杂度可以得到以下几条路径，科目管理白盒测试如下表5.10。

5.10 科目管理白盒测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 路径 | 测试用例 | 期望结果 | 运行结果 |
| 1 | 1-2-3-4-5 | 均填写合法的信息，点击提交按钮。 | 科目创建成功。 | 科目创建成功。 |
| 2 | 1-2-3-5 | 未选择填写合法的信息，点击提交按钮。 | 科目创建失败。 | 科目创建失败。 |



图5.2 添加科目方法程序流程图（左）及控制流图（右）

5.3.3 成绩管理方法测试

本模块对录入成绩功能进行白盒测试。如图5.3所示。

通过基本路径的测试方法，可以确定环路复杂度V(G) = 2。

根据环路复杂度可以得到以下几条路径，成绩管理白盒测试如下表5.11.

表5.11 成绩管理白盒测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 路径 | 测试用例 | 期望结果 | 运行结果 |
| 1 | 1-2-3-4-5 | 均填写合法的成绩信息，点击保存按钮。 | 成绩录入成功。 | 成绩录入成功。 |
| 2 | 1-2-3-5 | 未填写合法的成绩信息，点击保存按钮。 | 成绩录入失败。 | 成绩录入失败。 |



图5.3 录入成绩方法程序流程图（左）及控制流图（右）

总结

本文旨在描述一个基于Spring Boot和MyBatis的教务管理系统的设计与实现过程。该系统的前端采用JSP、jQuery和Layui技术。

在本项目中，我们首先进行了需求分析和系统设计。通过与教务部门的合作，我们确定了系统的主要功能模块，包括学生管理、课程管理、教师管理、成绩管理等。我们设计了相应的数据库结构，用于存储学生、课程、教师和成绩等信息。

在后端开发过程中，我们选择了Spring Boot框架作为系统的基础框架，并结合MyBatis技术实现了数据访问层。使用Spring Boot的优势，我们快速搭建了一个稳定的、可扩展的后端架构。通过MyBatis的ORM（对象关系映射）功能，我们能够方便地将数据库表映射为Java对象，简化了数据访问的操作。

同时，我们还实现了一些重要的业务逻辑。例如，学生管理模块中，我们设计了学生信息的增加、删除、查询和修改功能，通过与前端的交互实现了学生信息的展示和操作。科目管理模块中，我们实现了科目信息的增删改查，并设计了选课功能，使学生能够在系统中选择自己的课程。教师管理模块中，我们设计了教师信息的管理和查询功能，实现了对教师信息的维护。

在前端开发方面，我们选择了JSP作为页面的模板技术，并利用jQuery和Layui框架实现了页面的动态效果和用户交互。通过使用这些前端技术，我们开发了用户友好的界面，使用户能够方便地操作系统并获取所需的信息。

在项目实施过程中，我们遇到了一些挑战和困难。例如，系统的安全性和权限管理是一个重要的考虑因素。为了确保系统的安全性，我们实施了身份认证和授权机制，只有经过认证的用户才能访问系统的敏感功能。此外，我们还优化了系统的性能，通过合理的数据库设计和索引优化，提升了系统的查询和响应速度。

综上所述，通过本项目的设计与实现，我们成功地开发了一个基于Spring Boot和MyBatis的教务管理系统。该系统具备了学生管理、课程管理、教师管理和成绩管理等功能，并且在前端使用了JSP、jQuery和Layui等技术，提供了良好的用户体验。该系统的设计与实现为教务部门的工作提供了便利，并具有一定的实用价值。未来，我们可以进一步完善系统的功能，提升系统的性能和安全性，以满足不断变化的教务管理需求。

致谢

通过设计和实现教务管理系统，我不仅获得了关于教务管理的深入了解，还学到了许多软件设计和开发的技能。这些收获将对我未来的学习和职业发展产生积极的影响，并为我在实际工作中能够更好地应对挑战提供了坚实的基础。

本文是在我的指导教师易文美教师的指导下完成的，在本系统项目的开发和论文的创作过程中，教师给予了我很多具体的指导和帮助，她以孜孜不倦为人师表的工作态度，为我们在学习和生活中树立了榜样。在论文的创作过程中教师非常的耐心为我指导了我论文的诸多错处和不合逻辑之处，让我能顺利地完成论文创作，在论文完稿之际，特向易文美教师表示诚挚的感谢。

同时还要感谢这两年学习中所有的代课教师和我的辅导员，他们在学习上给予了我很大的帮助，我本身理论基础薄弱，在这两年各位教师的孜孜教导下，进步显著，受益匪浅。也感谢辅导员对我生活的关心以及耐心的教导让我成为了更好的人。

除此之外，我还要感谢在学习上给予了我帮助的同学，在我遇到问题时耐心且及时的为我解疑答惑，让我的学习之路更加顺畅，提高了我的学习效率，我在此表示诚挚的感谢！

然后我非常感谢我们的学校——宜宾学院，感谢学校给我们提供了这样舒适美丽的学习环境，在这两年的学习中，我的眼界也得到了很大的开拓，深深感受到本科学府的紧张激烈严肃的学习氛围，在这样的氛围下我一改了往日的懒惰习性，紧跟大家紧凑的学习节奏，自身收益匪浅，所以在此，也向我们的学校表示真诚的感谢，也祝福宜宾学院能够在未来发展得越来越好。

在这两年的学习最后，我要感谢我的母亲成女士，感谢她的支持和鼓励，在我难过时，耐心开导；在我抑郁苦闷时，给予我最大的鼓励；在我迷茫不知所措时，给我指引方向。非常感谢成女士这二十几年对我辛勤的付出和耐心的陪伴，我想说：“我爱你，妈妈！”。这个世界因为它的参差才会丰富多彩，我相信人以温柔待世界，世界也终会以温柔回馈到我们身上，希望大家的过去都能有处安放，祝福所有人的未来都能安然无恙！

参考文献

1. 吕冬梅. 新时代下高职院校教务管理系统功能优化研究[J]. 今日财富, 2016(17): 143-144.
2. 何中辰. 新时期出版信息管理系统设计之思考[J]. 编辑学刊,2022(5): 100-105.
3. 张忠华,王冀鲁,朱向荣，等. 基于区块链技术的学分银行系统的设计与实现——以北京外国语大学的“外语教育创新”项目为例[J]. 现代教育技术, 2022,32(9): 100-108.
4. Paul P.K.,Bhuimali A.,Aithal P.S.,Tiwary K.S.,Saavedra R.,Mewada S.. Emerging IT and Computing Gradients in Information Sciences[J]. International Journal of Applied Science and Engineering,2021,9(1).
5. 涂国章. 探索大数据在高校学生管理中的应用[J]. 黑龙江教师发展学院学报, 2023,42(4): 19-22.
6. 姜强,赵蔚,李勇帆，等 基于大数据学习分析的在线学习绩效预警因素及干预对策的实证研究[J]. 电化教育研究, 2017,38(1): 62-69.
7. Bin W,Chengyu J,Zhifei W. Design and Development of Student Achievement Management System Based on JSP[J]. Journal of Social Science and Humanities,2022,4(6).
8. Wei Huiting. Optical Hybrid Network Structure Based on Cloud Computing and Big Data Technology[J]. Journal of Sensors,2022,2022.
9. 李珂. 论综合性大学舞蹈教学资源管理模块的划分[J]. 艺术评鉴, 2020(6): 130-131.
10. 齐贵超,朱贵秋,李慧亭，等. 师范类专业认证信息化辅助系统设计与实现[J/OL].中国教育技术装备: 1-6[2023-04-19].http://kns.cnki.net/kcms/detail/11.4754.T.20230112.1341.001.html.
11. 朱振文. 基于Web的教学管理平台的设计与实现[J]. 甘肃科技纵横, 2020,49(8): 10-14+20.
12. 郑飞. 基于J2EE的在线答疑系统的设计[J]. 信息技术, 2012,36(12): 161-163.
13. 刘宇明. 教育信息化背景下中学生物学教师的信息技术素养探析[J]. 教师,2022(34): 84-86.
14. 杨宝龙. 基于J2EE的多用户学术主页系统研究与实现[D].西安: 西安电子科技大学, 2020.DOI: 10.27389/d.cnki.gxadu.2020.000719.
15. 王昊,刘友华. 信息系统分析与设计[M]. 南京: 南京大学出版社,2021.
16. 焦怡迪. 天津市学生体质健康大数据平台的设计与实现[D]. 天津: 天津体育学院, 2021.DOI: 10.27364/d.cnki.gttyy.2021.000006..
17. 段伟涛. 基于Web的化学实验室信息管理系统的设计与实现[D]. 西安: 西安电子科技大学, 2021.DOI:10.27389/d.cnki.gxadu.2021.000589.
18. 黄静斌. 基于多种目标类型的视频监控系统应用方案[D]. 浙江: 浙江工商大学, 2019.
19. 杨鹏飞. 社区安防可视化管理系统的设计与实现[D]. 河北: 河北工程大学, 2020: 10.27104/d.cnki.ghbjy.2020.000240.
20. 黄梦歌. 基于社会网络分析的电影推荐系统的设计与实现[D]. 西北民族大学, 2022:10.27408/d.cnki.gxmzc.2022.000122.