工 程 技 术 学 院

School of Engineering and Technology



**生产实习**

**系 部: 工程技术学院**

**专业班级: 计科1603**

**姓 名: 宿孟**

**学 号: 201640806**

**指导教师: 邓强**

**2019年9月制**

|  |  |  |  |
| --- | --- | --- | --- |
| **生产实习报告** | | | |
| 实习任务 | 网上选课系统的设计与实现 | | |
| 指导教师 | 邓强 | 专业班级 | 计科1603 |
| 实习时间 | 2019 | 实习地点 | -- |
| 实习目的 | | | |
| 增强我的实践能力、培养学生提高分析问题和解决问题的能力以及综合运用所学基础知识和基本技能的重要途径，也是我最终完成本科教学不可或缺的阶段。生产实习是学生培养计划的重要组成部分，是在教学过程中是学生获得生产实际知识，巩固和深化所学理论知识，贯彻理论与实际结合的教学原则，加强学生的工程实践训练，培养学生获取知识、运用知识创新能力的重要教学环节;是使学生了解社会，接触实际，进行素质教育的重要途径。 | | | |
| 实习内容及要求 | | | |
| 实习内容：  这次的课题我们小组做的是网上选课系统，采用JSP语言进行开发，通过小组的共同努力，实现了网上选课系统中所需求的各个功能，使得系统满足用户的需求。  实习要求：  第1-2周：资料收集整理，确定项目。 第3-5周：编程语言、数据库、开发环境学习，需求分析，编写项目方案。 第6周：系统设计，规划体系结构图、数据设计E-R图。 第7-10周：程序编写，调试运行。 第11-12周：实习报告编写、完善修改，答辩。 | | | |
| 目录  [一．设计题目 4](#_Toc24969481)  [1.1设计选题 4](#_Toc24969482)  [1.2设计的目的 4](#_Toc24969483)  [1.3的设计背景 4](#_Toc24969484)  [二.需求分析 4](#_Toc24969485)  [2.1功能需求 4](#_Toc24969486)  [2.2 系统的开发运行环境 7](#_Toc24969487)  [三． 程序功能模块图及数据流图 7](#_Toc24969488)  [3.1程序功能模块图 7](#_Toc24969489)  [3.2 数据流图 7](#_Toc24969490)  [四． 总体设计 8](#_Toc24969491)  [4.1 系统体系结构设计 8](#_Toc24969492)  [4.2 系统功能说明 8](#_Toc24969493)  [4.3 数据库设计 9](#_Toc24969494)  [4.4 界面设计 11](#_Toc24969495)  [五． 技术难点及实现 11](#_Toc24969496)  [5.1程序结构 11](#_Toc24969497)  [5.2应用技术及实现 12](#_Toc24969498)  [5.3部分代码 12](#_Toc24969499)  [六．实习总结 15](#_Toc24969500) 一．设计题目1.1设计选题 《网上选课系统》 1.2设计的目的 目的旨在推动软件的规范化，使设计人员遵循统一的概要设计书写规范，  节省制作文档的时间，降低系统实现的风险，做到系统设计资料的规范性和全面性，以利于系统的实现、测试、维护、版本升级等。 1.3的设计背景 随着在校人数的增加，选课是重点问题，拥有一个好的网上选课系统可以简便快速的对选课各种信息进行管理，节省了人力和时间，提高了对课程管理的效率。 二.需求分析2.1功能需求 分为三个用户  管理员、教师、学生   1. 管理员 2. 登录功能   需求描述：通过已知的管理员账号登录，登录成功进入管理员界面，方可执行相应功能。   1. 添加教师信息   需求描述：添加一个新的教师信息，教师工号作为教师身份唯一标识，添加时会对输入框中教师工号是否存在进行判断并提示。   1. 修改教师信息   需求描述：可以选择教师并修改其信息，教师编号不可修改。   1. 删除教师信息   需求描述：删除某一条选中教师的所有信息。   1. 查询教师信息   需求描述：选择查询条件，可查询所有教师、通过教师编号查询教师信息，并以表格的形式显示出来。   1. 添加课程信息   需求描述：添加一个新的课程信息，课程编号作为该课程身份唯一标识，添加时会对输入框中课程编号是否存在进行判断并提示。其中一些属性通过下拉框方便选择。   1. 修改课程信息   需求描述：可对选中的课程进行修改，其中课程编号不可修改   1. 删除课程信息   需求描述：删除选定课程的所有信息。   1. 查询课程信息   需求描述：根据相应条件查询课程，所有课程、根据编号、课程类型、所属学院等进行查询。  10.添加学生信息  需求描述：添加一个新的学生信息，学号作为学生身份唯一标识，添加时会对输入框中学号是否存在进行判断并提示。其中学院、专业等信息通过下拉框方便选择。  11.修改学生信息  需求描述：修改选中学生的信息，其中学号不可修改。  12.删除学生信息  需求描述：删除选中学生的所有信息。  13.查询学生信息  需求描述：通过相应条件查询学生，例如查询所有信息，通过学号、学院、专业、班级等条件查询学生信息。   1. 教师 2. 查询个人信息   需求描述：可以查询自己的所有信息   1. 修改个人信息   需求描述：只能修改自己密码，其他信息都不可修改   1. 选择课程   需求描述：通过查询课程表，查看已有课程，并选择自己需要教学的课程。   1. 添加课程安排   需求描述：通过选择一门已有课程，对该课程进行安排包括上课班级名、上课时间、所在教师、学分学时、总人数等。   1. 修改课程安排   需求描述：修改自己已经安排的课程的信息。   1. 删除课程安排 2. 查询课程安排   需求描述：查询自己的已选教学课程的各种信息   1. 查询选课信息   需求描述：选定自己教学课程，查看该课程的学生学科情况，包括一些学生信息。   1. 打印选课信息   需求描述：选择自己教学的一门课程，并打印出该课程的学生信息。   1. 学生 2. 修改个人信息   需求描述：只能修改自己的密码，其他信息需联系管理员修改。   1. 选课   需求描述：选择一门已有课程，并进入确认，提示学习该课程是否已有教师安排。   1. 查询课程安排   需求描述：查询自己所选课程的信息。   1. 查询选课人数   需求描述：查询课程已选人数。   1. 退选。   需求描述：退选自己已选课程。 2.2 系统的开发运行环境 本系统开发平台: JSP + MYSQL  本系统集成开发环境: Eclipse + Tomcat  本系统运行环境: Windows OS 程序功能模块图及数据流图3.1程序功能模块图  3.2 数据流图  总体设计4.1 系统体系结构设计 界面首先为选择登录身份，选择后进入相应登录界面，登陆成功后的主界面由两块组成，一个为左栏，用于显示功能以及功能操作的，可以手动隐藏和显示；还有主界面，用于操作对应功能以及查看结果等。主界面内含一个div盒子，用于美观。使用户能够在简单、易用、单一、统一的可视化界面下，轻松、方便地访问到各种类型的数据。 4.2 系统功能说明 1.管理员  ①可以对学生信息通过诸多条件进行查询；  ②可以对学生信息进行修改以及删除；  ③添加一个新的学生信息  ④可以对教师信息查询、修改、删除以及添加。  ⑤可对课程进行修改、删除、查询、以及添加。  2.教师  ①可以查询管理员添加的课程信息，并选择自己需要教学的课程。  ②选择自己教学课程后对该课程进行安排，包括上课时间、上课地点、学分、学时、总人数等。  ③可以对自己已选教学课程的课程安排进行修改、删除等操作。  ④可以查询自己教学课程的学生选课信息。  ⑤查看自己信息，并可以修改密码。  3.学生  ①可以查询全部管理员已添加的课程，并可以通过课程编号、课程名、课程所属专业、课程类型等条件查询。  ②可以选择查询出来的课程进行选课，选择后进入确认界面，显示课程名，提示是否已有教师安排此课，若有则显示教师姓名，并有确认和取消两个按钮，确认则选课成功，取消则返回上一界面。  ③查看选课结果，用表格形式显示出课程名、上课教室、上课地点、学分、学时以及代课教师。没有教师安排的课程无法查出，只能在退选中查看单独的课程名。  ④退选已选课程，先查询出自己已选的所有课程，通过课程名选择退选。  ⑤查看个人所有信息，只可以修改密码。 4.3 数据库设计 创建六个表，学生表、教师表、管理员表、课程表、课程安排表、选课表。  各表属性类型约束如下：  管理员表admin   |  |  |  |  | | --- | --- | --- | --- | | 列名 | 类型 | 约束 | 备注 | | Aname | Varchar（12） | 主键 | 账号 | | Apassword |  |  | 密码 |   学生表Student   |  |  |  |  | | --- | --- | --- | --- | | 列名 | 类型 | 约束 | 备注 | | Sid | Char(12) | 非空、主键 | 学号 | | Sname | Varchar（8） | 非空 | 姓名 | | Sidcard | Char(18) | 非空 | 身份证 | | Ssex | Char(2) | 男女 | 性别 | | Spassword | Varchar(12) | 非空 | 密码 | | Sage | Char（2） |  | 年龄 | | Classr | Varchar（15） |  | 班级 | | profession | Varchar（15） |  | 专业 | | College | Varchar（15） |  | 学院 |   教师表Teacher   |  |  |  |  | | --- | --- | --- | --- | | 列名 | 类型 | 约束 | 备注 | | Tid | Char（4） | 主键 | 工号 | | Tname | Varchar(8) |  | 姓名 | | Tpassword | Varchar(12) |  | 密码 | | Tsex | Char(2) | 男女 | 性别 | | Introduction | Varchar（100） |  | 简介 |   课程表course   | 列名 | 类型 | 约束 | 备注 | | --- | --- | --- | --- | | Cid | Char（4） | 主键 | 课程号 | | Cname | varChar(15) |  | 课程名 | | Cintroduction | Varchar（100） |  | 简介 | | Type | Varchar（15） |  | 类型（选修，必修） | | Belongcoll | Varchar（15） |  | 所属学院 | | Belongpro | Varchar（15） |  | 所属专业 |   课程安排   |  |  |  |  | | --- | --- | --- | --- | | 列名 | 类型 | 约束 | 备注 | | Courseclass | Varchar（12） | 主键 | 开课班级 | | Coursetime | Varchar（12） |  | 上课时间 | | Courseweek | Varchar（12） |  | 上课周 | | Cid | Char（4） | 外键 | 课程号 | | Tid | Char（4） | 外键 | 教师编号 | | Classroom | Varchar（6） |  | 上课教室 | | credits | Int（11） |  | 学分 | | period | Int（11） |  | 学时 | | Totalnum | Int（11） |  | 总人数 |   选课表sc   |  |  |  |  | | --- | --- | --- | --- | | 列名 | 类型 | 约束 | 备注 | | Id | varChar（4） | 自动增长、主键 | 用于传值 | | Sid | Char(12) | 外键 | 学生学号 | | Cid | Char（4） | 外键 | 课程编号 |   其中设计了表与表之间的关联属性以及一些约束，例如删除学生信息时，该学生在选课表中的相应信息也会被删除。 4.4 界面设计 登录（以管理员为例）  图片1  管理员登录界面（以管理员为例），左侧栏可收缩  图片2  课程管理界面  图片3 技术难点及实现5.1程序结构 网上选课系统利用MVC设计模式，利用spring mvc框架，将其分为视图层、控制器层、模型层。视图层为jsp文件，分别存放在admin、teacher、student三个文件夹中，分别是管理员界面、教师界面、学生界面，而index和登录界面以及操作成功失败界面直接放在主目录；控制器层为servlet类，用来处理视图层发送的每个请求，存放在以域名翻转命名的包内：net.fuzui.servlet；模型层用来存放javabean类，分为实体bean和业务bean中，分别放在以域名翻转命名包net.fuzui.beans、net.fuzui.dao中，实体bean存放实体类，业务bean中存放数据库连接以及对数据库的操作，由servlet调用。 5.2应用技术及实现 在网上选课系统中，用到基于注解的spring mvc框架，在视图层中的每一次请求都会发送到控制器层，由servlet类通过注解识别请求，处理请求，需要用到数据库时则调用DAO。先通过配置web.xml以及springmvc.xml来应用框架，由@RequestMapping 来处理请求映射，在处理url和表单提交时，会自动根据url或者表单中参数的名字和方法中同名形参进行匹配并赋值。  在添加学生、教师、课程信息时，通过jQuery运用ajax技术，如果添加的学号已存在，在输入框右侧会红字提示该学号已存在，不存在也会有提醒该学号可添加，教师工号、课程编号也如此。  在学生界面查询选课结果时，利用了数据库表与表之间级联，通过选课表、课程表、课程安排表、教师表四表级联的sql语句，查询出课程名、班级名称、上课时间、教室、学分学时以及代课老师信息，并将此信息存放在一个实体类中，方便调用。  在视图层的样式设计中，通过css设置背景色透明度渐变和背景图片透明度来实现整体视觉效果，在该系统中，表比较多，所以将表的样式也焕然一新。总界面分为左侧栏和显示栏，运用js，左侧栏通过点击相应图标可收缩隐藏。  运用了el表达式和jstl标记，是界面更加简约。  在代码编写过程中，通过system.out.println()来调错。 5.3部分代码 Springmvc.xml配置文件  <!--静态资源交给默认的Servlet-->  <mvc:default-servlet-handler/>  <mvc:annotation-driven></mvc:annotation-driven>  <context:component-scan base-package=*"net.fuzui"*></context:component-scan>  <bean class=*"org.springframework.web.servlet.view.InternalResourceViewResolver"*>  <property name=*"prefix"* value=*"/"*></property>  <property name=*"suffix"* value=*".jsp"*></property>  </bean>  下拉列表查询显示，在多数查询中，可以查询全部信息，也可以在下拉列表选择，通过选择条件，并在输入框中输入条件的值来查询。通过@RequestParam来确定表单中所传参数，以课程查询为例：  //  @RequestMapping(value = "/query", method = RequestMethod.***GET***)//get方法  **public** String redirect(@RequestParam("serc") String serc,@RequestParam("condition") String condition,Model model) {  CourseDAO courseDAO = **new** CourseDAO();  ArrayList<Course>courseList = **new** ArrayList<Course>();    **if**(serc.equals("all")) { //查询全部  courseList = courseDAO.getAllCourse(); //调用DAO  model.addAttribute("courseList",courseList);//返回数据到视图层  System.***out***.println(courseList);  **return** "redirect:finalPage";//重定向    }**else** **if**(serc.equals("sid")) {//通过课程编号查询  courseList = courseDAO.getCourseCid(condition);//调用模型层  model.addAttribute("courseList",courseList);  System.***out***.println("sid");    **return** "redirect:finalPage";    }**else** **if**(serc.equals("nam")) {//通过课程名称  courseList = courseDAO.getCourseName(condition); //调用模型层  model.addAttribute("courseList",courseList);  System.***out***.println(courseList);  System.***out***.println("cla");  **return** "redirect:finalPage";    }**else** **if**(serc.equals("col")) {//通过所属专业  courseList = courseDAO.getCourseCol(condition); //调用模型层  model.addAttribute("courseList",courseList);  System.***out***.println(courseList);  System.***out***.println("col");  **return** "redirect:finalPage";    }**else** **if**(serc.equals("type")) {//通过课程类型  courseList = courseDAO.getCourseType(condition); //调用模型层  model.addAttribute("courseList",courseList);  System.***out***.println(courseList);  System.***out***.println("pro");  **return** "redirect:finalPage";    }**else** {  courseList = courseDAO.getAllCourse();  model.addAttribute("courseList",courseList);  System.***out***.println(courseList);  **return** "redirect:finalPage";    }  }  在退出登录时，通过httpSession.removeAttribute("xx")来销毁session值 六．实习总结 这次web设计让我学习到很多。虽然结束了，但这只能是一个开始。今后今后作为程序员，要学习的规范，程序设计语言还有很多。怎样使自己从普通的PG升为SE，在计算机领域，要学的实在太多，仅大学生涯所学实在有限。我们只有对自己有了更高的要求，才能作为动力不断取得新的成绩!。  实习是我们每个大学生必须拥有的一段经历，它使我们在实践中了解社会，让我们学到了很多在课堂上根本就学不到的知识，受益匪浅，也打开了视野，增长了见识，为我们以后进一步走向社会打下坚实的基础。 | | | |

成绩评定表

|  |  |  |
| --- | --- | --- |
| 学生小结 | 经过这次的生产实习，我们小组完成了一个网上选课系统的设计与实现。在这个过程中，我们分工明确，每个人负责一部分的东西，其中我主要负责的是后端逻辑代码架构的编写，并和一部分同学一起完成该模块，正所谓众人拾柴火焰高，在多人讨论的情况下，很多问题都能迎刃而解。并结合了多位同学的建议和想法，做出了相对合理的管理系统。  虽然一个人可以很强，强到所有轮子都是自己造，但众人的力量相比较起来会更加强大，合理的分配每个人适合的工作，将会大大的增加整体项目的进度和完成度，超额不奢望，但基本目标肯定是没有问题的，这也是团队合作工作的一个亮点，也是团队合作的最大优势。 | |
| 教师评语 |  | |
| 成绩 | 平时成绩 | 综合成绩 |
|  |  |
| 指导教师（签字）： | |