****

**滁州学院3**

**本 科 生 毕 业 设 计**

**（申请学士学位）**

**论 文 题 目** 基于MVC的课程教学网站设计与实现

**作 者 姓 名** 宋秋雨

**所学专业名称** 计算机科学与技术

**指 导 教 师** 王正山

**2014年05月29日**

**学 生： （签字）**

**学 号：2010211082**

**论文答辩日期：2014 年05月28日**

**指 导 教 师 ： （签字）**

**滁州学院本科毕业设计（论文）原创性声明**

本人郑重声明：所呈交的设计（论文）是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写的成果。本人完全意识到本声明的法律后果由本人承担。

作者签名： 年 月 日

滁州学院本科毕业设计

目 录

[1 引言 2](#_Toc387329816)

[1.1 选题背景 2](#_Toc387329817)

[1.2 选题意义 2](#_Toc387329818)

[1.3 主要工作 2](#_Toc387329819)

[1.4 组织结构 3](#_Toc387329820)

[2 相关知识 3](#_Toc387329821)

[2.1建模技术 3](#_Toc387329822)

[2.1.1 UML 3](#_Toc387329823)

[2.3 Struts2框架 3](#_Toc387329824)

[2.4 Tomcat服务器概述 3](#_Toc387329825)

[2.5 MySQL数据库 4](#_Toc387329826)

[2.5.1 MySQL 简介 4](#_Toc387329827)

[2.5.2 MySQL 数据库的优点 4](#_Toc387329828)

[2.6 测试技术 4](#_Toc387329829)

[2.6.1 黑盒测试 4](#_Toc387329830)

[2.6.2 白盒测试 4](#_Toc387329831)

[3 系统需求分析 5](#_Toc387329832)

[3.1 用例图 5](#_Toc387329833)

[3.1.1 系统前台功能用例图 5](#_Toc387329834)

[3.1.2 系统后台功能用例图 5](#_Toc387329835)

[3.2 活动图 6](#_Toc387329836)

[3.2.1 登录活动图 6](#_Toc387329837)

[3.2.2 新增课程活动图 7](#_Toc387329838)

[3.3 顺序图 7](#_Toc387329839)

[3.3.1 登录顺序图 7](#_Toc387329840)

[3.3.2 注册顺序图 8](#_Toc387329841)

[4 系统设计 8](#_Toc387329842)

[4.1 课程教学网站设计指导思想和原则 8](#_Toc387329843)

[4.2 课程教学网站总体功能概述 8](#_Toc387329844)

[4.3 系统功能结构 9](#_Toc387329845)

[4.4 系统数据库设计 9](#_Toc387329846)

[4.4.1 概念结构设计 9](#_Toc387329847)

[4.4.2 系统E-R图 10](#_Toc387329848)

[4.4.3 逻辑结构设计 10](#_Toc387329849)

[5 系统实现 12](#_Toc387329850)

[5.1 系统的软件结构 12](#_Toc387329851)

[5.2 登录/注册模块的实现 13](#_Toc387329852)

[5.3 课程管理模块的实现 14](#_Toc387329853)

[5.3.1 添加课程的实现 14](#_Toc387329854)

[5.3.2 维护课程信息的实现 16](#_Toc387329855)

[5.4 用户管理模块的实现 18](#_Toc387329856)

[5.4.1 维护教师用户信息的实现 18](#_Toc387329857)

[5.5 评论管理模块的实现 19](#_Toc387329858)

[5.6 投票模块的实现 19](#_Toc387329859)

[5.7 上传下载模块的实现 19](#_Toc387329860)

[6 系统测试 19](#_Toc387329861)

[6.1 系统测试的方法与步骤 19](#_Toc387329862)

[6.2 模块测试 20](#_Toc387329863)

[6.2.1 登录/注册测试 20](#_Toc387329864)

[6.2.2 用户管理模块测试 21](#_Toc387329865)

[6.2.3 课程管理模块测试 21](#_Toc387329866)

[6.2.4 评论管理模块测试 22](#_Toc387329867)

[6.2.5 投票模块测试 23](#_Toc387329868)

[6.2.6 上传下载模块测试 23](#_Toc387329869)

[结束语 23](#_Toc387329870)

[参考文献 23](#_Toc387329871)

基于MVC的课程教学网站的设计与实现

摘要：随着网络在社会中使用的越来越广泛，现代技术特别是信息技术在教育教学中得到了广泛的应用，课程教学网站的建设也越来越受到人们的重视。鉴于课程教学网站在课程建设中的重要作用，我国各高校均投入了大量人力物力进行课程教学网站的开发，许多软件公司推出了一些精品课程开发系统。但这些开发系统，虽然功能强大，但购买价格较高，且一般不提供源代码，给系统升级和二次开发造成困难。因此，开发通用性强、易于管理的课程教学网站是十分必要的。

关键词：课程教学；MVC；struts2；JSP技术；数据库；Web技术；

Design And Implementation Of Teaching Website Based On MVC Course

**Abstract**：With the use of Internet in the society more and more widely in modern technology, especially information technology has been widely applied in education and teaching, the construction of teaching website is attracting more and more attention. In view of the course website plays an important role in the course of construction, the development of our colleges and universities have invested a lot of manpower curriculum teaching website, many software companies have launched a number of excellent courses development system. But these development system, although powerful, but the purchase price is higher, and generally do not provide the source code, to create difficulties for system upgrade and development two times. Therefore, the development of strong versatility, easy teaching website management is very necessary.

**Key words**：Teaching; MVC; Struts2; JSP; database; Web Technology;

# 1 引言

## 1.1 选题背景

基于MVC[1]的框架模式，是目前使用最广泛的框架模式，它包括三个核心部件：[模型](http://baike.baidu.com/view/96500.htm)、[视图](http://baike.baidu.com/view/71981.htm)、[控制器](http://baike.baidu.com/view/122229.htm)。这三个部件是相互独立，改变其中一个不会影响其它两个，这样可以减少代码的复制及代码的维护量，也易于维护。而现在应用广泛的Struts2[2-3]是Java Web MVC框架中不争的王者，它能更好的组织MVC框架，使得系统的开发效率事半功倍,并且通过UML建模对系统进行分析，通过对Java EE编程技术[4-6]和数据库技术[7]的理解，并利用软件工程规范[8]来开发项目，是非常必要的。而本系统网站正式应用基于MVC模式的struts2主流框架。

课程教学网站[9]可以方便学生对跨学科课程的学习，也可以方便老师让自己的课程知识传播给更多的学生。学生可以收藏自己喜欢的课程，也可以评论课程的某些内容，老师也可以回复评论；学生还可以对某些知识点进行投票，老师可以根据票数反映的数据对这些知识点进行修改。而且，老师还可以上传一个对应的课件或作业供学生下载学习。这些教学方式是传统教学方式所无法比拟的。

## 1.2 选题意义

通过课程教学网站的开发，可以使学生通过网络学习到一些精品课程的知识和原理方法，并通过在线的一些关联例题来练习和巩固学到的理论知识；可以使老师将一些专业知识向更多的学生展示，让更多的学生来充实自己的知识库。通过互动的学习模式，更好的为师生提供交流的空间，以帮助学生轻松理解课程的内容；结合丰富的学习资源，打破教材的局限性，有效地帮助学生开拓视野，了解更多的知识。

通过该系统的开发，让我能将自己所学的知识点串联起来,加深对MVC架构、Java EE编程技术和数据库技术的理解，提高自身的综合应用能力，以及提高自己解决实际问题的动手能力，为将来从事软件开发奠定坚实的基础。

## 1.3 主要工作

本系统主要是根据用户来分为前台页面和后台页面，并以此来设计相关的功能，前台用户主要有学生和教师。前台的主要功能有不同的用户可以注册，用户登录，退出、个人信息的维护、浏览信息；学生可以浏览某门课程的信息，收藏课程，留言或投票并提出学习中遇到的问题；教师可以维护所任课程的信息，统计学生的投票信息并根据投票信息对相应内容进行改进，回复留言并解答学生提出的问题等功能。后台用户主要事管理员，管理员主要功能有管理前台注册的用户（包括教师和学生）信息、管理教师所添加的课程信息以及管理个人信息（主要是密码修改）等功能。

实现上述功能主要设计到的技术：基于MVC模式的struts2框架。

## 1.4 组织结构

摘要：主要介绍下课程教学网站的现状；引言主要介绍选择本毕业设计题目的背景、意义、以及主要的课程教学网站的主要功能；2相关知识：主要是介绍下在本设计本系统网站中主要是用到的技术；

# 2 相关知识

## 2.1建模技术

### 2.1.1 UML

UML[10]是一种基于面向对象的可视化建模语言，它为开发人员提供了一套统一的标准建模符号。UML图包括用例图、类图、时序图、状态图、活动图、组件图和部署图，其中用例图描述系统功能和参与者之间的关系，时序图主要表示详细的流程，显示了流程中不同对象之间的调用关系，活动图表示在处理某个活动时，两个或者更多类对象之间的过程控制流。

UML适用于以面向对象技术来描述的任何系统，而且适用于系统开发的不同阶段，从需求规格描述到系统完成后的测试与维护。UML的主要作用有：

（1）通过对问题进行说明和可视化描述，帮助理解问题，建立文档。

（2）通过UML图的描述，可以获取和交流有关应用问题求解知识。

（3）通过对解决方案说明和可视化描述，辅助构建系统并建立文档。

## 2.3 Struts2框架

Struts 2[11-12]是Struts的下一代产品，是在 struts 1和WebWork的技术基础上进行了合并的全新的Struts 2框架。其全新的Struts 2的体系结构与Struts 1的体系结构差别巨大。Struts 2以WebWork为核心，采用拦截器的机制来处理用户的请求，这样的设计也使得业务逻辑控制器能够与ServletAPI完全脱离开，所以Struts 2可以理解为WebWork的更新产品。虽然从Struts 1到Struts 2有着太大的变化，但是相对于WebWork，Struts 2的变化很小。

## 2.4 Tomcat服务器概述

Tomcat服务器是开放源代码的Web 应用服务器。它是由Apache、Oracle 和其他一些公司及个人共同开发的，是Apache软件基金会的项目中的一个核心项目。因为有Oracle的支持，Tomcat能具有最新的Servlet和JSP规范。由于免费的Tomcat 性能稳定、技术先进，到了部分软件开发商的肯定和Java 爱好者的偏爱，成为目前比较流行的Web 应用服务器。Web执行流程中的Tomcat服务器，它在运行时占用系统资源小扩展性好并支持负载平衡与邮件服务，同时在并发访问用户少和中小型系统运中普遍使用，是开发和调试JSP程序的首选。综上本系统将使用Tomcat作为Web服务器。

## 2.5 MySQL数据库

### 2.5.1 MySQL 简介

MySQL是一个适合中小型网站的数据库管理[系统](http://baike.baidu.com/view/25302.htm)[13]。它的[体积](http://baike.baidu.com/view/274417.htm)小、速度快、灵活性强、丰富的应用编程接口、总体拥有成本低，尤其是开放源码这一特点，受到了许多用户的喜爱并且都会选择MySQL作为网站[数据库](http://baike.baidu.com/view/1088.htm)。MySQL是一个快速、多用户、多线程和健壮的SQL数据库服务器。虽然比其他大型数据库（例如Oracle、SQL Server）相比功能稍弱，但是MySQL的小巧灵活成本低等特性使得一些中小型企业更加偏向于MySQL数据库。MySQL主要目标是快速、健壮和易用。在Web开发的过程中，数据库占有很重要的位置，必须有可靠的安全性和易于管理便于操作等特性。

### 2.5.2 MySQL 数据库的优点

1. 减少记录编档的时间
2. 减小记录检索时间
3. 灵活的查找序列
4. 灵活的输出格式
5. 多个用户可以同时访问记录

## 2.6 测试技术

### 2.6.1 黑盒测试

黑盒测试[14]也称[功能测试](http://baike.baidu.com/view/651435.htm)，它是通过测试来检测每个功能是否都能正常使用。在测试中，把[程序](http://baike.baidu.com/view/17674.htm)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](http://baike.baidu.com/view/2809361.htm)进行测试，它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。黑盒测试着眼于[程序](http://baike.baidu.com/view/17674.htm)外部结构，不考虑内部[逻辑结构](http://baike.baidu.com/view/540423.htm)，主要针对[软件界面](http://baike.baidu.com/view/43207.htm)和软件功能进行测试。黑盒测试是以用户的角度，从输入数据与输出数据的对应关系出发进行测试的。很明显，如果外部特性本身设计有问题或规格说明的规定有误，用黑盒测试方法是发现不了的。具体的黑盒[测试用例设计](http://baike.baidu.com/view/1270753.htm)方法包括等价类划分法、边界值分析法、错误推测法、[因果图法](http://baike.baidu.com/view/4532272.htm)、判定[表驱动](http://baike.baidu.com/view/3707325.htm)法、正交试验设计法、功能图法、[场景](http://baike.baidu.com/view/222559.htm)法等。

### 2.6.2 白盒测试

白盒测试也称结构测试或逻辑驱动测试，它是按照[程序](http://baike.baidu.com/view/17674.htm)内部的结构测试程序，通过测试来检测产品内部动作是否按照设计规格说明书的规定正常进行，检验程序中的每条通路是否都能按预定要求正确工作。 这一方法是把测试对象看作一个打开的盒子，测试人员依据程序内部[逻辑结构](http://baike.baidu.com/view/540423.htm)相关信息，设计或选择[测试用例](http://baike.baidu.com/view/106882.htm)，对程序所有逻辑路径进行测试，通过在不同点检查程序的状态，确定实际的状态是否与预期的状态一致。

白盒测试的测试方法有代码检查法、静态结构分析法、静态质量度量法、[逻辑覆盖](http://baike.baidu.com/view/178784.htm)法、基本[路径测试](http://baike.baidu.com/view/4533384.htm)法、[域测试](http://baike.baidu.com/view/2427471.htm)、符号测试、[路径覆盖](http://baike.baidu.com/view/2193850.htm)和程序变异。

白盒测试法的覆盖标准有逻辑覆盖、循环覆盖和基本路径测试。其中逻辑覆盖包括语句覆盖、判定覆盖、条件覆盖、判定/条件覆盖、条件组合覆盖和路径覆盖。

# 3 系统需求分析

## 3.1 用例图

用例图主要用来描述系统与外部参与者之间的交互。用例图包含系统边界、用例、参与者和关联等符合组成。系统边界主要是由一个矩形表示，内部包含多个用例。用例是系统对外的一个可见功能，在UML中用椭圆形表示。参与者是使用系统的人员、设备和外部子系统，在UML中用一个人形符号表示。参与者与用例之间通过关联建立关系。

### 3.1.1 系统前台功能用例图

系统前台有两类用户，分别为学生和教师，学生和教师的功能用例图如下：

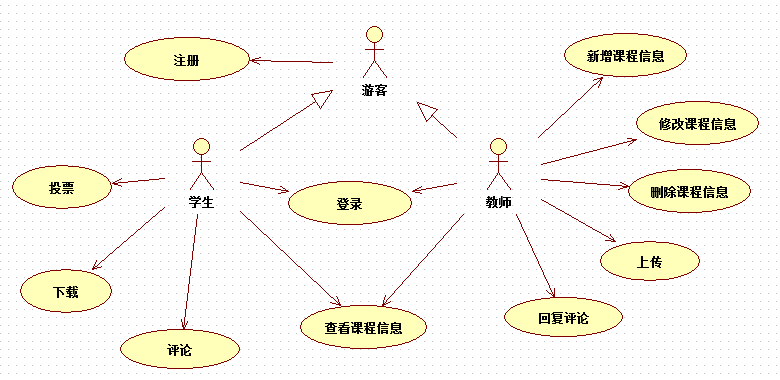


图3-1前台用户用例图

### 3.1.2 系统后台功能用例图

系统后台的用户只有管理员，管理员的功能用例图如下：

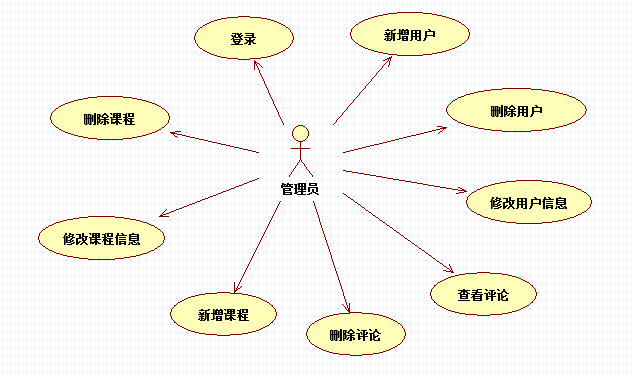


图3-2管理员用例图

## 3.2 活动图

### 3.2.1 登录活动图

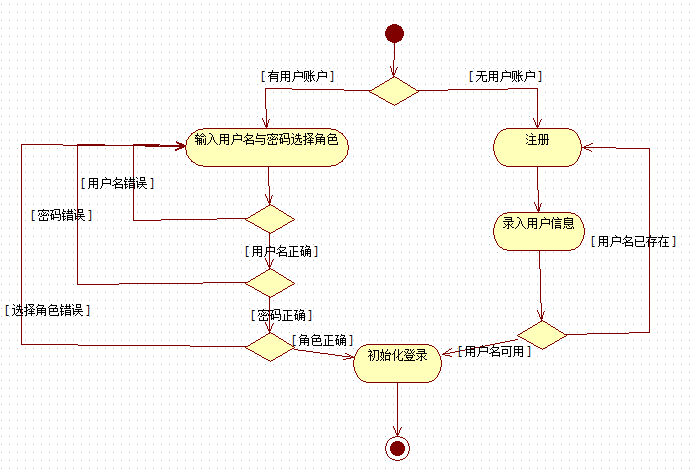


图3-3登录活动图

### 3.2.2 新增课程活动图

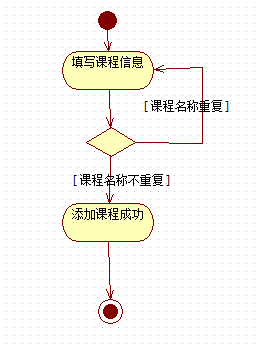


图3-4 新增课程活动图

## 3.3 顺序图

### 3.3.1 登录顺序图

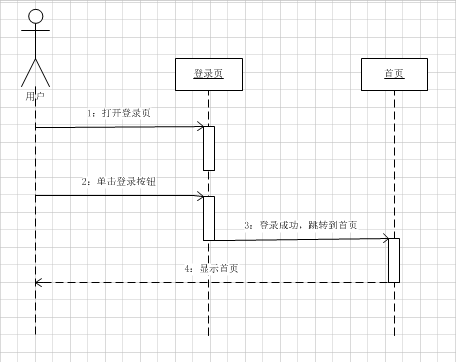


图3-5登录顺序图

### 3.3.2 注册顺序图

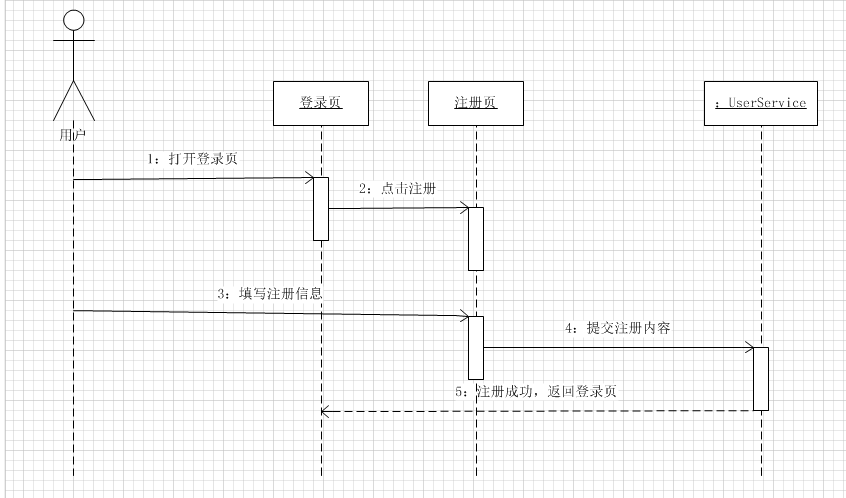


图3-6注册顺序图

# 4 系统设计

## 4.1 课程教学网站设计指导思想和原则

（1）利用软件开发现有软硬件环境，及先进的管理系统开发方案，从而到达充分利用现在资源提高系统开发水平和应用效果的目的。

（2）系统应该符合软件工程开发的理论，开发方法等开发依据。

（3）系统应满足课程网上学习的需要，并达到操作过程中的直观，方便，实用，安全，准确等要求。

（4）系统应具备数据库维护功能，及时根据用户需求进行数据库的各种操作。

（5）系统采用原型，实用模块化程序设计方法，便于系统功能的各种组合和修改，以及系统的测试与维护。

（6）课程教学网站的设计应适合校园的发展。

（7）课程教学网站的设计应当遵循数据库设计规范。

## 4.2 课程教学网站总体功能概述

课程教学网站包含六个模块分别是：登录/注册、课程管理、用户管理、评论管理、投票、上传下载。

登录/注册模块：学生/教师和管理员都必须登录才能进入系统，用户登录时在后台判断用户的权限类型，分为学生、教师和管理员，学生可以浏览课程，收藏课程、对课程进行评论和投票；教师可以添加课程、维护课程、回复评论；管理员可以管理用户信息，课程信息。

课程管理模块：教师/管理员通过验证界面进入系统网站后，进入到课程管理模块，可以对课程进行添加，删除、修改课程信息、查询、查看等一系列的操作。

用户管理模块：管理员通过验证界面进入系统网站后，进入到用户管理模块，可以对用户进行添加，删除、修改课程信息、查询、查看等一系列的操作。

评论管理模块：学生可以添加评论；教师可以回复评论；管理员可以对评论进行查看、删除操作。

投票模块：学生登陆后，可以对课程信息进行投票。

上传下载模块：教师可以上传一些课件；学生可以下载那些课件。

## 4.3 系统功能结构

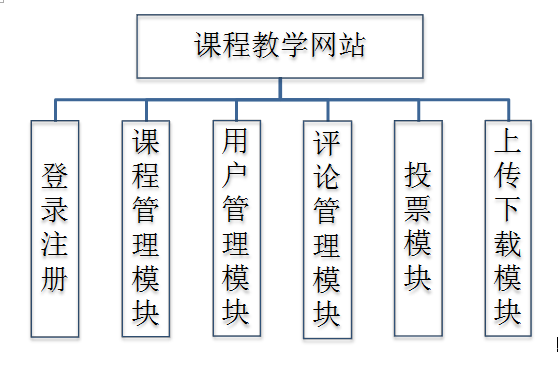


图4-1系统功能结构图

## 4.4 系统数据库设计

### 4.4.1 概念结构设计

（1）学生

学生属性有：学号、姓名、密码、所属院系、电话、email、权限、职业。

（2）教师

教师属性有：职工号、姓名、密码、所属院系、电话、email、权限、职业、职称。

（3）管理员

管理员属性有：用户名、密码、权限。

（3）课程

课程属性有：课程名称、简称、课程属性、英文名称、课程代码、课时、学分、课程简介。

### 4.4.2 系统E-R图



图4-2 系统E-R图

### 4.4.3 逻辑结构设计

根据上述功能模块的需求，设计如下主要数据库表:

表4-1用户表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段英文名称** | **字段中文名称** | **字段类型** | **字段约束** | **是否可为空** |
| **id** | 用户编号 | Int（5） | 主键 | 否 |
| **userID** | 用户名 | Varchar(20) |  | 否 |
| **password** | 密码 | Varchar(20) |  | 否 |
| **name** | 姓名 | Varchar(50) |  | 否 |
| **faculty** | 院系 | Varchar(200) |  | 否 |
| **email** | 邮箱 | Varchar(50) |  | 否 |
| **title** | 职称 | Varchar(100) |  | 是 |
| **phone** | 电话 | Varchar(20) |  | 是 |
| **power** | 权限 | Int（2） |  | 否 |
| **position** | 角色 | Varchar(50) |  | 否 |

表4-2课程信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段英文名称** | **字段中文名称** | **字段类型** | **字段约束** | **是否可为空** |
| **courseName** | 课程名称 | Varchar(200) | 主键 | 否 |
| **faculty** | 院系 | Varchar(200) |  | 否 |
| **shortName** | 简称 | Varchar(200) |  | 是 |
| **englishName** | 英文名称 | Varchar(200) |  | 是 |
| **classHour** | 课时 | Varchar(4) |  | 否 |
| **classScore** | 学分 | Varchar(4) |  | 否 |
| **attribute** | 课程属性 | Varchar(100) |  | 否 |
| **code** | 课程代码 | Varchar(50) |  | 是 |
| **description** | 课程简介 | Varchar(1000) |  | 否 |
| **teacher** | 任课教师 | Varchar(50) |  | 是 |

表4-3留言评论表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段英文名称** | **字段中文名称** | **字段类型** | **字段约束** | **是否可为空** |
| **leaveID** | 留言编号 | Int（4） | 主键 | 否 |
| **content** | 留言内容 | Varchar(500) |  | 否 |
| **recontent** | 回复留言 | Varchar(200) |  | 是 |
| **contenttime** | 留言时间 | Date |  | 否 |
| **recontenttime** | 回复时间 | Date |  | 否 |
| **leavename** | 留言人 | Varchar(50) |  | 否 |
| **releavename** | 回复人 | Varchar(50) |  | 否 |

表4-4投票表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段英文名称** | **字段中文名称** | **字段类型** | **字段约束** | **是否可为空** |
| **voteID** | 投票编号 | Int（20） | 主键 | 否 |
| **votecount** | 票数 | Int（20） |  | 否 |
| **type** | 类型 | Varchar(50) |  | 是 |
| **votename** | 投票人 | Varchar（50） |  | 否 |

# 5 系统实现

## 5.1 系统的软件结构

处理静态WEB是由Web浏览器向Web发送静态页面， Web服务器直接对发送的静态网页进行处理。

处理动态WEB数据库查询的过程是：当Web浏览器向Web服务器发送请求时，动态网页的处理流程是，屏蔽掉HTML语言，只输入动态网页文件，由Web服务器向数据库中传递信息，经过数据库的处理返回数据集，Web应用服务器再把含有程序代码的动态网页转换为静态网页返还给Web浏览器。这就是用动态网页对数据库进行查询并将数据返还给浏览器处理的全过程，具体见图5.1所示。



图5-1数据库查询示意图

## 5.2 登录/注册模块的实现

本模块主要是用户用过课程教学网站系统首页登录进入该系统，用户输入用户名和密码，若输入错误，会有提示信息，并重新输入，若用户名不存在，会提示用户名不存在，请先注册；若输入正确，会根据用户的身份进行相应的权限判断，学生和教师进入前台页面，管理员进入后台页面。具体页面如图所示：



图5-2登录

其实现代码如下：

Action层：

**public** String Login()

{

Users users = **new** Users();

UserService userService = **new** UserService();

users = userService.loginUser(username, password);

**if**(users.getUserCode() == **null**){

**return** *NONE*;

}**else** {

**return** *SUCCESS*;

}

}

Service层：

**public** Users loginUser(String username,String password){

Users users = **new** Users();

String sql = "select \* from users where userID=? and password=?";

**try** {

conn = Db.*createConn*();

ps=conn.prepareStatement(sql);

ps.setString(1, username);

ps.setString(2, password);

rs = ps.executeQuery();

rs.next(); `

}

**catch** (SQLException e) {

e.printStackTrace();

} **finally** {

db.*close*(rs);

db.*close*(conn);

}

**return** users;

}

## 5.3 课程管理模块的实现

课程管理模块主要包括添加课程，修改课程信息，查看课程信息，删除课程。其中，所有用户都可以查看课程信息，教师可以管理自己的课程信息，管理员可以管理所有的课程信息。

### 5.3.1 添加课程的实现

点击添加课程功能，填入课程的相应信息（不必填的可以不填），填写完整后，可以点击提交，具体页面如图所示：



图5-3添加课程

Action代码：

**public** String AddCourseInfo() **throws** Exception

{

Course course = **new** Course();

course.setCourseName(courseName);

course.setFaculty(faculty);

course.setShortName(shortName);

course.setEnglishName(englishName);

course.setClassHour(classHour);

course.setClassScore(classScore);

course.setAttribute(attribute);

course.setCode(code);

course.setDescription(description);

CourseInfoService courseService = **new** CourseInfoService();

courseService.addCourseInfo(course);

**return** *SUCCESS*;

}

Service代码：

**public** **void** addCourseInfo(Course course){

String sql="insert into courseinfo(courseName,faculty,shortName,englishName,classHour,classScore,attribute,code,description) values(?,?,?,?,?,?,?,?,?)";

**try** {

conn = db.*createConn*();

ps = conn.prepareStatement(sql);

ps.setString(1, course.courseName);

ps.setString(2, course.faculty);

ps.setString(3, course.shortName);

ps.setString(4, course.englishName);

ps.setString(5, course.classHour);

ps.setString(6, course.classScore);

ps.setString(7, course.attribute);

ps.setString(8, course.code);

ps.setString(9, course.description);

ps.executeUpdate();

} **catch** (SQLException e) {

e.printStackTrace();

}**finally** {

db.*close*(conn);

}

}

### 5.3.2 维护课程信息的实现

点击课程列表，可以看到所有课程信息的列表，在每条信息后都有编辑和删除的操作功能，列表信息如图所示：



图5-4课程信息列表

Action代码：

**public** String ListCourseInfo() **throws** Exception

{

CourseInfoService courseService = **new** CourseInfoService();

courses=courseService.listCourseInfo();

**return** "success1";

}

Service代码：

**public** List<Course> listCourseInfo(){

List<Course> list = **new** ArrayList<Course>();

**try** {

conn = Db.*createConn*();

String sql="select \* from courseinfo";

ps = conn.prepareStatement(sql);

rs = ps.executeQuery();

**while**(rs.next()){

Course course=**new** Course();

course.setCourseName(rs.getString("courseName"));

course.setFaculty(rs.getString("faculty"));

course.setClassHour(rs.getString("classHour"));

course.setClassScore(rs.getString("classScore"));

course.setAttribute(rs.getString("attribute"));

System.*out*.println(course.getCourseName());

list.add(course);

}

} **catch** (SQLException e) {

e.printStackTrace();

}

Db.*close*(ps);

Db.*close*(conn);

**return** list;

}

（1）删除课程功能的实现

Action代码：

**public** String DeleteCourseInfo() **throws** Exception

{

CourseInfoService courseService = **new** CourseInfoService();

courseService.DeleteCourseInfo(courseName);

**return** "delete";

}

Service代码：

**public** **void** DeleteCourseInfo(String courseName) {

Course course = **new** Course();

String sql ="delete from courseinfo where courseName=?";

**try** {

conn = Db.*createConn*();

ps = conn.prepareStatement(sql);

ps.setString(1, courseName);

ps.executeUpdate();

} **catch** (SQLException e) {

e.printStackTrace();

}

Db.*close*(ps);

Db.*close*(conn);

}

## 5.4 用户管理模块的实现

管理员点击用户管理功能，会显示教师用户和学生用户功能列表，通过点击教师用户或学生用户操作，会看到教师信息列表或学生信息列表，并在列表中有相应的编辑/删除操作功能。具体页面如图所示：



图5-5用户管理

### 5.4.1 维护教师用户信息的实现

具体页面如图所示：

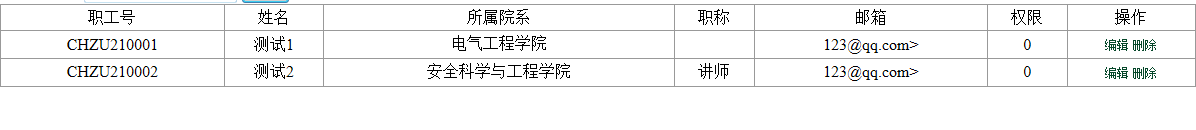


图5-6教师信息列表

Action代码：

**public** String TeacherInfo() {

UserService userService = **new** UserService();

users = userService.listTeacherInfo();

**return** "success1";

}

Service代码：

**public** List<Users> listTeacherInfo(){

List<Users> list = **new** ArrayList<Users>();

**try** {

conn = Db.*createConn*();

String sql="select \* from users where position='教师'";

ps = conn.prepareStatement(sql);

rs = ps.executeQuery();

**while**(rs.next()){

Users users = **new** Users();

users.setUserCode(rs.getString("userID"));

users.setTrueName(rs.getString("name"));

users.setFaculty(rs.getString("faculty"));

users.setTitle(rs.getString("title"));

users.setEmail(rs.getString("email"));

users.setPower(rs.getInt("power"));

list.add(users);

System.*out*.println(users.getTrueName());

}

} **catch** (SQLException e) {

e.printStackTrace();

}**finally** {

Db.close(conn);

}

**return** list;

}

## 5.5 评论管理模块的实现

学生可以留言评论课程，提出问题；教师可以回复留言，解答问题；管理员可以对评论进行查看、删除操作。

## 5.6 投票模块的实现

学生可以对课程进行投票，投票分为：非常满意、满意、还好、一般、看不懂。每个学生只能对课程投一次票。

## 5.7 上传下载模块的实现

教师可以上传他的课程对应的课件，学生可以下载这些课件。

# 6 系统测试

## 6.1 系统测试的方法与步骤

该系统在本地服务器上进行运行和调试，鉴于对系统的内部结构和处理算法的完全了解以及对系统功能的全面掌握对系统进行白盒测试和黑盒测试。

在开发软件系统的过程中，需要面对错综复杂的问题，因此，在软件生存周期的每个阶段都不可避免地会产生错误。测试目的在于：发现错误而执行一个程序的过程，测试的最重要目的是发现其中尚未发现的错误。

为了设计出有效的测试方案按照下面准则进行测试：所有测试都应追溯到用户需求；在完成了需求模型就要着手制定测试计划，在编码之前对所有测试工作进行计划和设计；御用Pareto原理着重对占出错误89%的容易出错的20%的模块进行测试，从小规模开始逐步进行大规模测试，通常先重点测试单个程序模块再转向集成的模块；精心设计测试方案，尽可能充分覆盖程序逻辑使之达到要求的可能性。

## 6.2 模块测试

### 6.2.1 登录/注册测试

（1）注册测试

进入注册页面，填写相应信息，点击注册按钮，如果职工号/学号不重复，且必填信息已填写，会注册成功。



图6-1教师注册

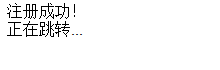


图6-2注册成功

（2）登录测试

进入登录界面，输入用户名、密码，点击登录按钮，如果用户名未填，提示用户名不能为空；如果密码未填，会提示密码不能为空；如果输入错误，会提示用户名或密码错误；如果用户名，密码都正确，则会成功登录。



图6-3教师登录

登录成功后



图6-4登陆成功

### 6.2.2 用户管理模块测试

管理员可以管理用户信息，主要功能操作有：添加用户，查看用户信息，修改用户信息，删除用户。

（1）添加用户

添加用户操作跟注册操作流程一样。

（2）查看用户信息

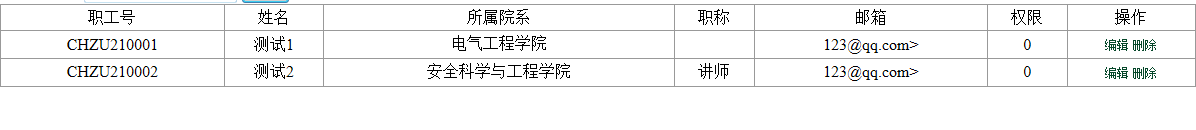


图6-5用户信息

（3）修改用户信息

未测试成功。

（4）删除用户

删除职工号为CHZU210001和CHZU210002的用户。



图6-6删除后的用户信息

### 6.2.3 课程管理模块测试

（1）新增课程测试

打开新增课程页面，输入相应信息，点击提交，会新增成功。



图6-7添加课程信息



图6-8查看添加的课程

（2）修改课程信息

点击编辑操作，可以看到课程的详细信息，可以修改相关信息，点击保存，会更新成功。

（3）删除课程

点击一条课程信息的删除操作，可以删除该条课程。

### 6.2.4 评论管理模块测试

学生对课程进行留言评论，提出问题，老师回复评论；管理员查看评论，也可以删除评论。

### 6.2.5 投票模块测试

学生对课程进行满意度投票，点击某个满意度类型后，此类型满意度的票数会加1.

### 6.2.6 上传下载模块测试

教师登录后，上传他的课程对应的课件，以供学生下载学习；学生登录后，下载课程所对应的课件。

# 结束语

不知不觉中，几个月过去了，毕业设计也已近尾声。这几个月，对我来说是短暂而充实。通过这几个月的学习实践，是我对软件工程规范式的开发系统网站有了更深的了解。从项目计划、需求分析到系统设计，再到系统实现和系统测试，一步一步的走下来，一点一点的解决问题，完善系统，到目前为止，虽然系统的一些功能还未完善，代码还在进行调试，但基本功能框架已实现。

本系统具有如下特点：

(1) 采用B/S三层体系结构，使系统具有很好的可维护性和可重用性。

(2) 在本系统的开发过程中采用的基于MVC的struts2框架开发模式，此模式将明显的把显示和逻辑分离，使代码容易管理，适合于大型项目的开发。

(3) 后台采用的MySQL数据库，MySQL数据库由于体积小、速度快、总体拥有成本低，可移植性好等特点，被大多中小型系统使用。

在本次毕业设计开始的时候，我在Java和Web应用程序设计方面不太熟悉，再加上毕业设计的时间有限，所以许多方面在下一步的工作中还需要进一步改进和完善。主要针对以下几方面：

(1)尽量采用满足MVC/模式要求的Struts2框架开发，使系统具有高的可重用性和适应性以及良好的可维护性。

(2)尽量从用户的角度来设计和完善系统，在一些验证提示信息上，做到人性化、易理解，界面做到简洁美观。

总之，经过本次毕业设计实践，让我学到了很多东西，也懂得了很多道理，这将为我今后更好的工作打下坚实的基础。

# 参考文献

[1]杨晓军.MVC Web开发学习实录[M].北京:清华大学出版社,2011.32-67

[2]高红岩.STRUTS实用开发指南[M].北京:科学出版社,2007.7.112-140

[3]廖义奎.Struts 编程基础与实例精讲[M].北京:中国电力出版社,2006.1.62-103

[4]李刚.轻量级JAVA EE企业应用实战[M].北京:电子工业出版社,2012:602-613

[5]Li, M. and H. Wang. A device management system based on JAVAEE Web[M]. 2009. Wuhan, China.

[6]Ou, J, et al. Design and research on teaching platform of stage task using JavaEE[M]. 2012. Chongqing, China.

[7]王国辉等.JSP数据库系统开发案例精选[M].北京:人民邮电出版社,2006.112-130

[8]史济民.软件工程-原理、方法与应用[M].北京:高等教育出版社,2012.345-380

[9]陆鑫,罗克露.[精品课程教学网站系统建设与教学信息化整体解决方案探索](http://www.cnki.com.cn/Article/CJFDTOTAL-DKJB2005S1018.htm%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20)[J].电子科技大学学报(社科版),2005.2-3

[10]郭霞.[基于UML的网上报名系统设计与实现](http://cdmd.cnki.com.cn/Article/CDMD-10056-2007043155.htm%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20)[D].天津大学,2006.155-170

[11]张胜华等.[基于MVC设计模式的Struts框架](http://www.cnki.com.cn/Article/CJFDTOTAL-XDJS200410008.htm)[J].现代计算机,2004.203-220

[12]孙卫琴.基于MVC的JAVA WEB设计与开发[M].北京:电子工业出版社,2005.87-122

[13]陈红.数据库系统原理教程[M].北京:清华大学出版社,2000.312-330

[14]佟伟光.软件测试[M].北京:人民邮电出版社,2008.42-77