logo_school

**毕业论文（设计）**

**基于SSH框架的学生信息管理系统**

**学 院 信息工程学院**

**专 业 计算机科学与技术**

**班 级 计算机科学与技术1班**

**学 号 16612500025**

**姓 名 孙文阁**

**指 导 教 师 向金明**

**成 绩**

云南工商学院

20 年 月

**云南工商学院**

**毕业论文（设计）诚信声明**

本人声明，所呈交的论文（设计）是本人在导师的指导下独立完成的研究成果。除了文中特别加以标注引用的内容外，本论文（设计）不包含法律意义上已属于他人的任何形式的研究成果,也不包含本人已用于其他学位申请的论文（或成果。对本文的研究作出重要贡献的个人和集体，均已在文中以明确方式表明。本人完全意识到本声明的法律后果由本人承担。本毕业论文(设计)成果归云南工商学院所有。

作者签名：

日 期：

**基于SSH框架的学生信息管理系统**

摘 要

该系统对于学校学生信息的特点以及管理中实际需要而设计，能够有效地实现学生信息管理的信息化，减轻管理人员的工作负担，高效率、规范化地管理大量的学生信息，促进学生对于学生的管理。

该系统采用了基于J2EE的SSH框架技术与三层架构软件设计思想。页面发送请求给控制器，控制器调用业务层处理逻辑，逻辑层向持久层发送请求，持久层与数据库交互，后将结果返回给业务层，业务层将处理逻辑发送给控制器，控制器再调用视图展现数据。

关键词：J2EE,SSH,学生信息管理

**ABSTRACT**

The system is designed for the characteristics of school students' information and the actual needs of management. It can effectively realize the informatization of student information management, reduce the workload of management personnel, manage a large number of student information efficiently and normatively, and promote the management of students.

The system adopts SSH Framework Technology Based on J2EE and three-tier architecture software design idea. The page sends the request to the controller, the controller calls the business layer processing logic, the logic layer sends the request to the persistence layer, the persistence layer interacts with the database, and then returns the result to the business layer, the business layer sends the processing logic to the controller, and the controller calls the view to display the data.

Key Words:J2EE,SSH, Student information management

前言

学生信息档案的管理对于学校的管理者来说至关重要。特别是近几年来，国家政策的调整，我国高等院校大规模的扩招，给高等院校的教学管理、学生管理、后勤管理等方面都带来不少的冲击。

随着科学技术的不断提高，计算机科学与技术日渐成熟，计算机应用的普及已进入人类社会生活的各个领域，并发挥着越来越重要的作用。这种传统的手工管理模式必然被以计算机为物质基础的信息管理方法所取代。

该系统对于学校学生信息的特点以及管理中实际需要而设计，能够有效地实现学生信息管理的信息化，减轻管理人员的工作负担，高效率、规范化地管理大量的学生信息，促进学生对于学生的管理。

该系统采用了基于J2EE的SSH框架技术与三层架构软件设计思想。页面发送请求给控制器，控制器调用业务层处理逻辑，逻辑层向持久层发送请求，持久层与数据库交互，后将结果返回给业务层，业务层将处理逻辑发送给控制器，控制器再调用视图展现数据。

管理员能够实现对学校、学院、专业、班级的增删改，以及对菜单的增删改，对用户权限的开启，禁用，对任课信息的管理和公告信息的管理。

学生用户登录后能发表自己的信息，展示在前端平台上，能够根据条件查询自己各个科目的成绩，可以根据学期统一查询成绩。

教师用户登录后对学生信息进行审核，修改。对自己授课的学生进行打分与评价。目 录（三号黑体，行距为单倍）

[摘 要 3](#_Toc32940879)

[**ABSTRACT** 3](#_Toc32940880)

[前言 4](#_Toc32940881)

[第一章、绪 论 7](#_Toc32940882)

[1. 开发背景 7](#_Toc32940883)

[2. 开发意义 7](#_Toc32940884)

[3. 可行性分析 7](#_Toc32940885)

[1) 经济可行性分析 7](#_Toc32940886)

[2) 技术可行性分析 7](#_Toc32940887)

[3) 应用价值的可行性分析 8](#_Toc32940888)

[4. 研究内容介绍 8](#_Toc32940889)

[5. 章节介绍 8](#_Toc32940890)

[第二章、相关技术概论 9](#_Toc32940891)

[1. B/S架构 9](#_Toc32940892)

[2. JavaWeb网站技术 9](#_Toc32940893)

[3. Spring框架 9](#_Toc32940894)

[4. SpringMVC思想 9](#_Toc32940895)

[5. Hibernate 9](#_Toc32940896)

[6. Layui前端框架 9](#_Toc32940897)

[7. Git组合开发 9](#_Toc32940898)

[8. 开发工具MyEclipse 9](#_Toc32940899)

[9. 服务器MY SQL 9](#_Toc32940900)

[10. Tomcat8.0 9](#_Toc32940901)

[第三章、需求分析 10](#_Toc32940902)

[1. 编写目的 10](#_Toc32940903)

[2. 需求描述 10](#_Toc32940904)

[1) 功能需求描述 10](#_Toc32940905)

[3. 运行环境需求分析 10](#_Toc32940906)

[第四章、概要设计 11](#_Toc32940907)

[1. 系统逻辑架构设计 11](#_Toc32940908)

[2. 系统功能模块设计 11](#_Toc32940909)

[3. 角色权限设计 11](#_Toc32940910)

[4. 系统数据库概要设计 11](#_Toc32940911)

[5. 数据库的实体设计 11](#_Toc32940912)

[1) 数据库背景 11](#_Toc32940913)

[2) 术语定义 11](#_Toc32940914)

[3) 表详细设计 11](#_Toc32940915)

[4) 视图详细设计 11](#_Toc32940916)

[第五章、详细设计 12](#_Toc32940917)

[1. 系统层次设计 12](#_Toc32940918)

[2. 系统项目规范 12](#_Toc32940919)

[1) 数据库的命名规范 12](#_Toc32940920)

[2) 项目文件命名规范 12](#_Toc32940921)

[3. 系统功能类图设计 12](#_Toc32940922)

[4. 系统业务接口设计 12](#_Toc32940923)

[5. 系统交互动作接口设计 12](#_Toc32940924)

[6. 系统UI设计 12](#_Toc32940925)

[第六章、系统的实现与测试 13](#_Toc32940926)

[1. 环境搭建 13](#_Toc32940927)

[2. 数据层的实现 13](#_Toc32940928)

[3. 数据连接层的实现 13](#_Toc32940929)

[4. 基于Hibernate的数据库连接创建与调用 13](#_Toc32940930)

[5. 数据操作层的实现 13](#_Toc32940931)

[6. 数据操作层的调用实现 13](#_Toc32940932)

[7. 实体交换层的实现 13](#_Toc32940933)

[8. 系统业务层的实现 13](#_Toc32940934)

[9. 信息业务层的调用实现 13](#_Toc32940935)

[10. 控制层的实现 13](#_Toc32940936)

[11. 控制层的调用实现 13](#_Toc32940937)

[12. 表示层的实现 13](#_Toc32940938)

[1) 代码 13](#_Toc32940939)

[2) 实用结果 13](#_Toc32940940)

[13. 软件测试用例说明 13](#_Toc32940941)

[1) 测试目的 13](#_Toc32940942)

[2) 网站端登录功能测试用例 13](#_Toc32940943)

[14. 测试报告 14](#_Toc32940944)

[第七章、总结 15](#_Toc32940945)

[1. 总结 15](#_Toc32940946)

[2. 展望 15](#_Toc32940947)

[结 论 （与正文空一行，三号黑体居中） 17](#_Toc32940948)

[致 谢（与正文空一行，三号黑体居中） 17](#_Toc32940949)

[参考文献（黑体三号，居中） 17](#_Toc32940950)

[附 录 （黑体三号，居中） 17](#_Toc32940951)

### 第一章、绪 论

#### 开发背景

如今是飞速发展的信息化时代，在各个行业中都离不开信息处理，这正是计算机被广泛应用于信息管理系统的环境。使用计算机进行信息管理，不仅可以提高工作效率，而且能提高其安全性。尤其是对于复杂的信息管理，计算机能充分发货它的优势。

信息管理系统是进行信息的采集、存储、加工、维护和使用的系统。它是科学管理和科学技术的发展而形成的。学生信息管理系统是各个高校不可缺少的部分，它的功能对于学校的决策者和管理者都至关重要，所以学生信息管理系统应该能为高校管理者提供更方便的信息处理与查询手段。传统人工的管理方式存在着效率低、保密性差等很多缺点，在存储和查询使都会出现不少问题。

作为计算机应用的一部分，使用计算机对学生信息进行管理，具有着手工管理所无法比拟的有点，如精准查询迅速、可靠性高、存储方便、安全性高、寿命长、成本低等。这些优点能极大提高管理者的工作效率，也是学校科学，正规化管理的必要条件。因此开发这样一套管理软件是很有必要的。

因此，本文将于SSH设计与开发一个学生信息管理系统，介绍了学生信息管理系统的实现过程：可行性分析、需求分析、数据库设计、功能设计、系统实现、系统测试以及系统功能简介。

#### 开发意义

开发学生信息管理系统，使用计算机对学生信息进行管理，可提高学校职工办公效率和规范管理，实现学生信息维护工作的系统化、规范化和自动化。系统可在学校的管理中发挥巨大作用。实现工作过程的计算机化，提高工作效率和工作质量。

学生信息管理系统提供了学生信息管理功能，方便管理者对学生信息的创建、查询、修改、删除、汇总、统计、保持等操作，可以让学校职工减轻工作压力，系统的对教务，教学的各个信息进行管理。对于高校规范管理有巨大意义。

#### 可行性分析

##### 经济可行性分析

按本系统开发方案，对于本系统软件投资方向，体现与软件开发、软件生产及软件测试。系统完全按开发计划进行估算，办公用品的消耗，办公设备的消耗，开发人员的生活与维护（包括水、电、房、工作餐）等。项目总体预估耗费10000元。本系统非收益性质开发，重要的是为解决学校进行学生信息管理的需要。

##### 技术可行性分析

本系统的开发环境为Windoows10系统，主要使用了MyEclipse、Tomcat8.0和navicat等工具做开发。该系统主要使用的技术是JavaWeb技术、SpringMVC、Hibernate、JSP和Layui前端框架。所用到的技术都已掌握，该系统只是把所学技术合理应用即可开发，所以技术上是完全可行的。

##### 应用价值的可行性分析

开发学生信息管理系统，使用计算机对学生信息进行管理，可提高学校职工办公效率和规范管理，实现学生信息维护工作的系统化、规范化和自动化。系统可在学校的管理中发挥巨大作用。实现工作过程的计算机化，提高工作效率和工作质量。

学生信息管理系统提供了学生信息管理功能，方便管理者对学生信息的创建、查询、修改、删除、汇总、统计、保持等操作，可以让学校职工减轻工作压力，系统的对教务，教学的各个信息进行管理。所以开发本系统对于高校管理拥有巨大的价值。

#### 研究内容介绍

#### 章节介绍

本文的章节介绍如下：

第一章前言，本章主要是介绍系统开发的背景、意义、系统的可行性和本文研究的内容。

第二章相关技术，本章主要是介绍开发的相关技术、工具和框架。

第三章需求分析，本章主要从系统的功能需求、设计、数据描述和用户角色等三方面分析系统需求。

第四章概要设计，本章主要是从系统总体设计和系统数据库概要设计来介绍。

第五章详细设计，本章主要从系统的数据库详细设计、系统界面设计、接口设计等来介绍。

第六章系统实现和测试，本章对系统运行结果进行展示，还从测试目的、系统测试、测试报告几个方面来描述测试过程和测试结果。

第七章总结与展望，本章介绍对本系统的总结和对未来的展望。

### 第二章、相关技术概论

#### B/S架构

B/S架构，就是浏览器/服务器，是一种网络结构模式。工作模式是浏览器请求服务器响应，这种模式在于统一了客户端，把实现系统功能的主要代码集中在服务器上。这样既简化了开发，还便于后期的维护和使用

#### JavaWeb网站技术

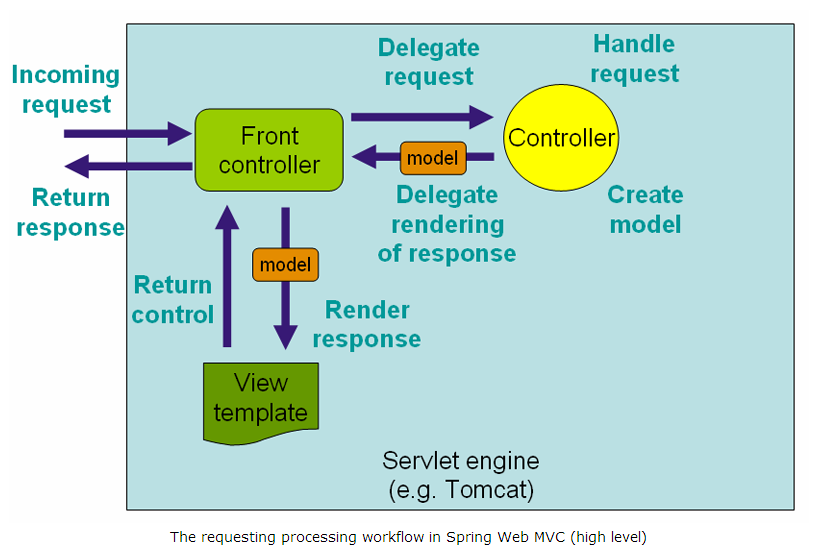
JavaWeb是一种技术总和，顾名思义就是后台用java代码编写web项目，标准的web服务器只具有与客户端浏览器通讯的功能，需要编写程序来复制处理客户端的请求。JavaWeb应用中可以包含Servlet、JSP、实用类、静态文档等。

#### Spring框架

Spring框架是由于软件开发的复杂性而创建的。Spring使用的是基本的JavaBean来完成以前只可能由EJB完成的事情。然而，Spring的用途不仅仅限于服务器端的开发。从简单性、可测试性和松耦合性角度而言，绝大部分Java应用都可以从Spring中受益。

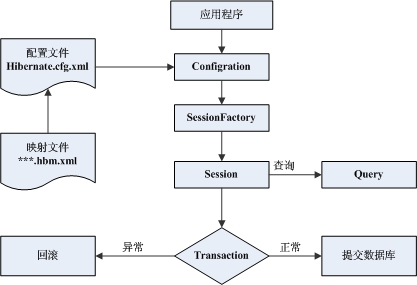
#### SpringMVC思想

Spring MVC 是一个模型 - 视图 - 控制器（MVC）的Web框架建立在中央前端控制器servlet（DispatcherServlet），它负责发送每个请求到合适的处理程序，使用视图来最终返回响应结果的概念。Spring MVC 是 Spring 产品组合的一部分，它享有 Spring IoC容器紧密结合Spring松耦合等特点，因此它有Spring的所有优点。



#### Hibernate

Hibernate是一种ORM框架，是轻量级JavaEE应用的持久层解决方案。它充当了面向对象程序设计语言和关系数据库之间的桥梁，使得JavaEE应用的OOA(面向对象分析)、OOD(面向对象设计)、OOP（面试相对象编程）三个过程一脉相承，成为一个整体。

Hibernate的运行过程如图所示：

#### Layui前端框架

layui是一款开源的模块化前端UI框架，它提供了丰富的内置模块，每个模块都可以通过模块化的方式按照调用者的需要加载。layui由布局、字体图标、按钮、表单、导航条、选项卡、表格等元素组成。

#### Git组合开发

Git本质上是一套内容寻址（content-addressable）文件系统，它主要具备的功能有协同修改、数据备份、版本管理、权限控制、历史记录、分支管理，它允许开发团队在工作过程中，使用多条生产线同时推进任务，更好的提高工作效率。

#### 开发工具MyEclipse

MyEclipse是 Genuitec公司开发出来的一款企业级开发环境，它是以eclipse为基础，然后再加上一些属于它自己的插件，再进一步的进行开发，才形成了这个功能强大的开发环境。它主要适用于企业级开发、Web开发和云开发。

#### 服务器MY SQL

MySQL 是一个关系型数据库管理系统，由瑞典 MySQL AB 公司开发，目前属于 Oracle 公司。MySQL 是一种关联数据库管理系统，关联数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。

#### Tomcat8.0

Tomcat是一个Apache公司研发的Web 应用服务器，简单来说是运行Java的网络服务器，它实现了Servlet和JSP规范，还提供HTTP服务，是一个很受欢迎的Java Web容器。它的主要优点是技术先进并且性能稳定。

### 第三章、需求分析

#### 编写目的

该项目可发布与高校网内，用于规范，合理的管理学生信息，减轻管理人员的工作负担，增加信息交互的效率，可节约纸张资源。对于促进高校学生管理有着重大意义。该项目使我增加大量开发经验，熟悉项目开发的完整流程，增强自己的开发能力，对于复杂的义务能够更好的理解。

#### 需求描述

##### 学生用户需求

学生用户登录可查询，修改，提交自己的基本信息，并可以根据条件查看自己各个学期，各个科目的成绩及教师评价。

##### 教师用户需求

教师登录后能够按条件查询学习信息，显示学生信息列表，并能对学生学期内自己授课的科目进行评价与打分。

##### 教师用户需求

管理员能够实现对学院、专业、班级的增删改，对课程信息，任课信息的添加，修改，删除，以及对教师身份真实性的校验,对公告信息的管理。

#### 运行环境需求分析

该系统的数据库采用navicat带的Mysql，服务器未来申请在阿里云中。

使用该系统的计算机所需最低配置：

CPU： P4 3.0GHz

内存： 4G以及4G以上

硬盘： 160G以上

光驱： CD

该系统一般兼容现市场上的主流浏览器，较推荐使用Google浏览器。

使用该系统的Adroid手机所需最低配置

系统：Android 8.0

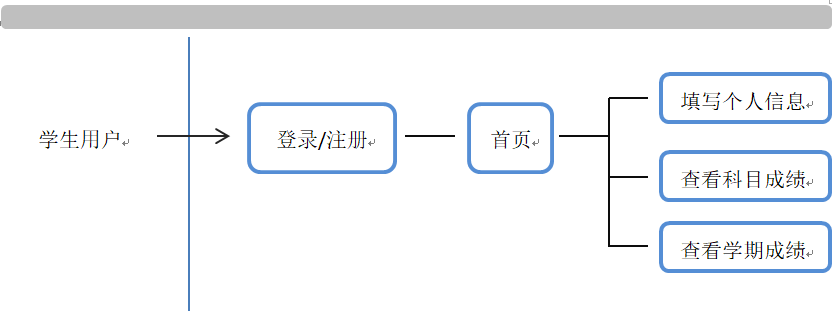
CPU：1000MHz ARMv5，

RAM：128MB

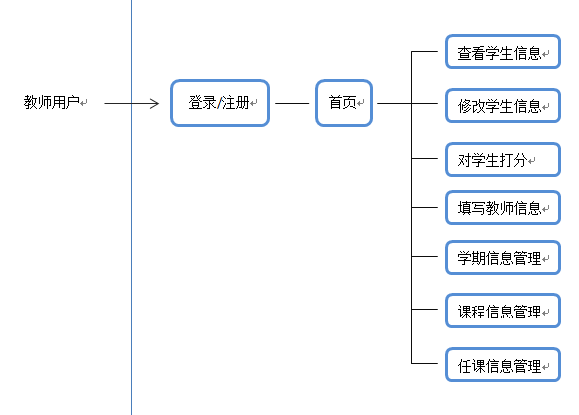
该系统网站端主要采用B/S模式。运行原理大致是，将系统部署在web服务器，用户在客户端【浏览器页面】提交表单操作，向服务器发送请求，等待服务器响应；服务器端接收并处理请求，应用服务器端通常使用服务器端技术，如JSP等，对请求进行数据处理，并产生响应；服务器端把用户请求的数据（网页文件、图片、声音等等）返回给浏览器，最后以一定的形式向客户展示数据。

### 第四章、概要设计

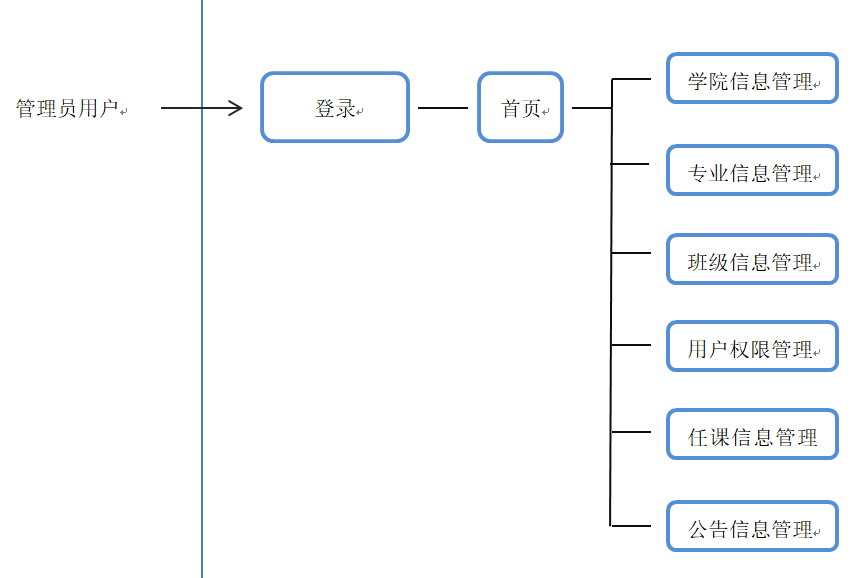
#### 系统逻辑架构



学生用户登录/注册，判断成功后进入首页，学生可填写，修改自己的个人信息，查看自己每个科目成绩，查看每个学期的成绩

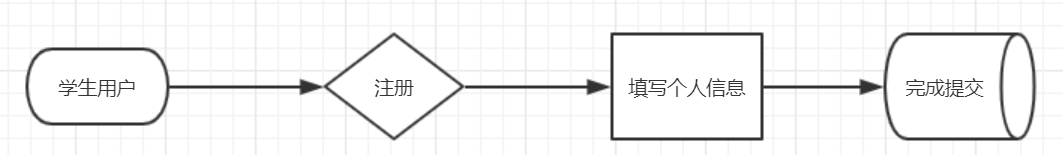


教师用户登录/注册，判断成功后进入首页，教师用户可填写自己基本信息，根据条件查询学生信息，列出学生列表，选中学生，可修改学生信息。可根据学院，班级查询，可根据自己任课查询。查询后对学生成绩进行评分，对学生进行评价。教师登录后进行权限判断，可管理学期信息，课程信息，任课信息。

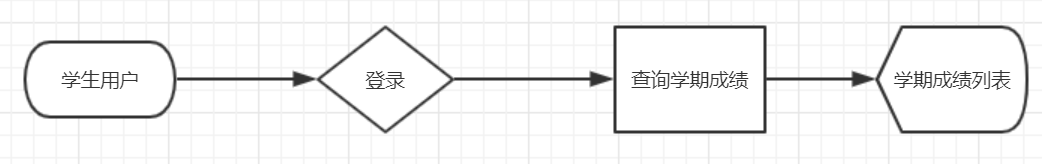


管理员用户登录，判断成功后进入首页，可对学院信息进行增删，查改以及对学院院长的绑定，对专业信息的增删，查改，对班级信息的增删查改，对班级辅导的绑定，对各个用户菜单权限的开启，禁用，对教师用户任课信息的增删，查改，对学校公告信息的发布，删除，修改。

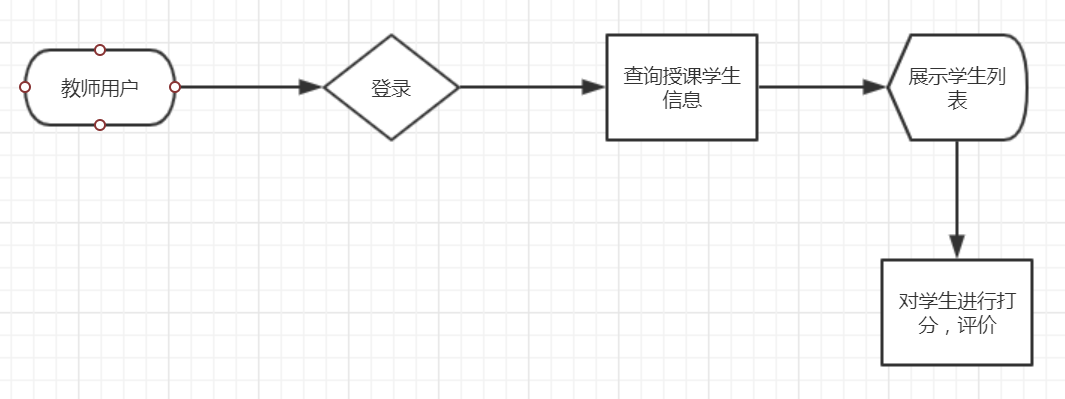
#### 系统功能模块设计



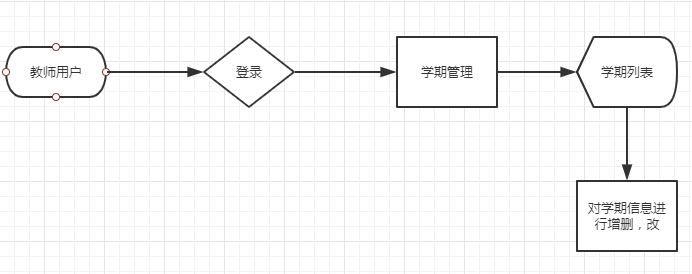
学生用户初次进行注册，注册成功后填写完善个人信息，完成提交。



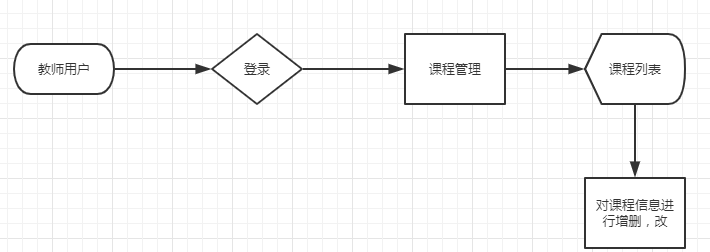
学生用户进行登录，登录成功后可按条件查询每个学期成绩，成功显示查询到的学期成绩列表。



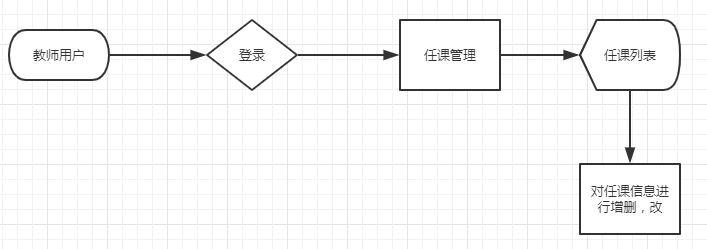
教师用户登录，判断成功后可按条件查询自己授课学生，查询后展示学生列表，可对学生成绩进行打分，对学生进行评价。



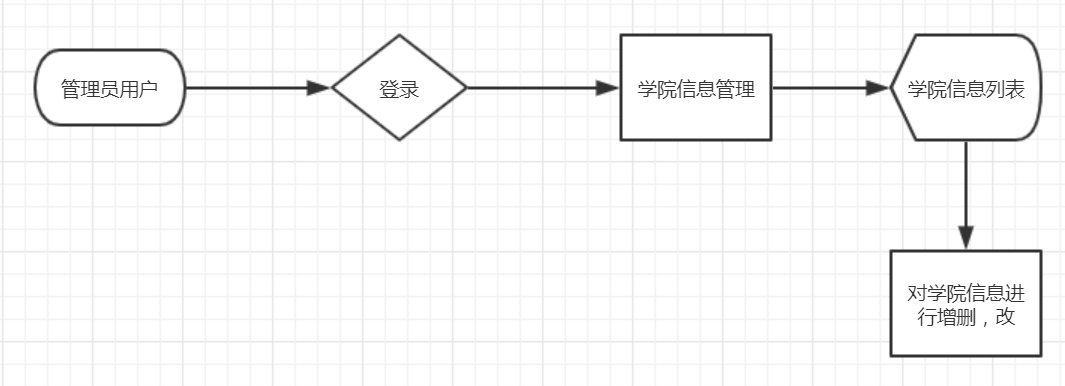
教师用户登录，判断成功后可按条件查询学期信息，查询后展示学期列表，可对学期信息进行添加，修改，删除。



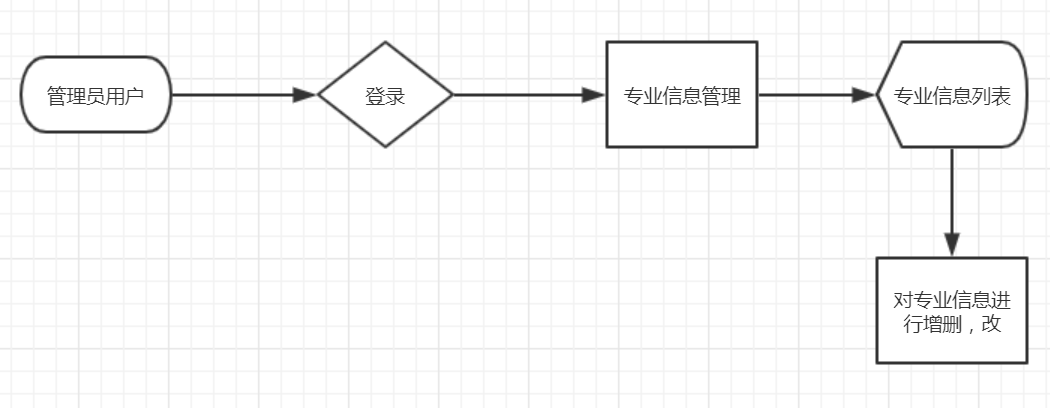
教师用户登录，判断成功后可按条件查询课程信息，查询后展示课程信息列表，可对课程信息进行添加，修改，删除。



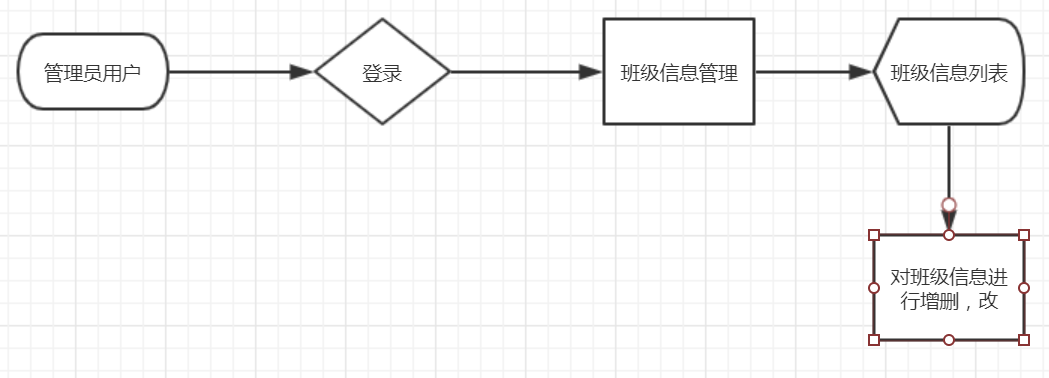
教师用户登录，判断成功后可按条件查询任课信息，查询后展示任课信息列表，可对任课信息进行添加，修改，删除。



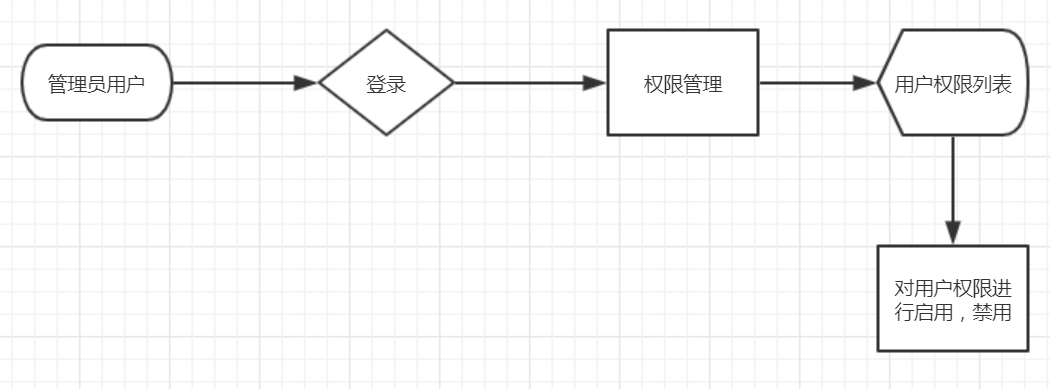
管理员用后登录，判断成功后可进入学院信息管理，获得学院信息列表，可对学院信息进行增删，改，对学院院长进行绑定，修改。



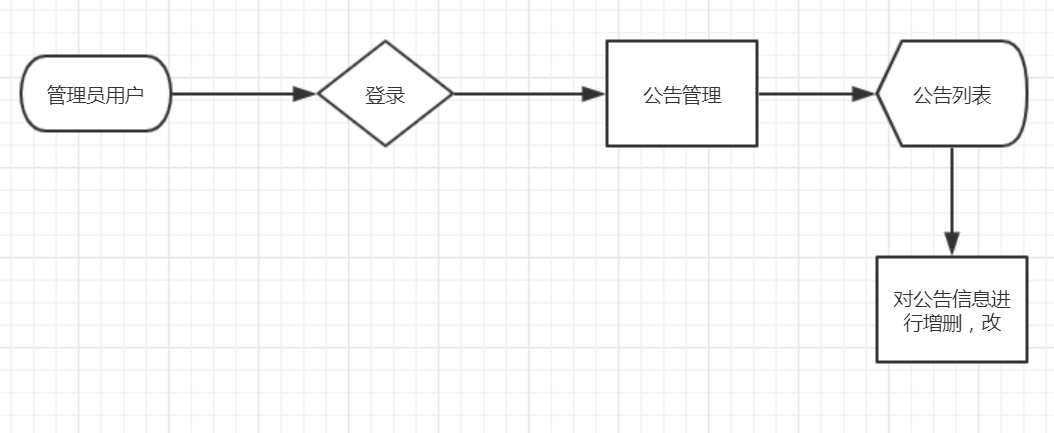
管理员用后登录，判断成功后可进入专业信息管理，获得专业信息列表，可对专业信息进行增删，改。对专业科长进行绑定，修改。



管理员用后登录，判断成功后可进入班级信息管理，获得班级信息列表，可对班级信息进行增删，改，对班级进行辅导员的绑定，修改。



管理员用后登录，判断成功后可进入权限管理，获得用户权限列表，可按条件查询用户权限，对用户权限进行启用，禁用。

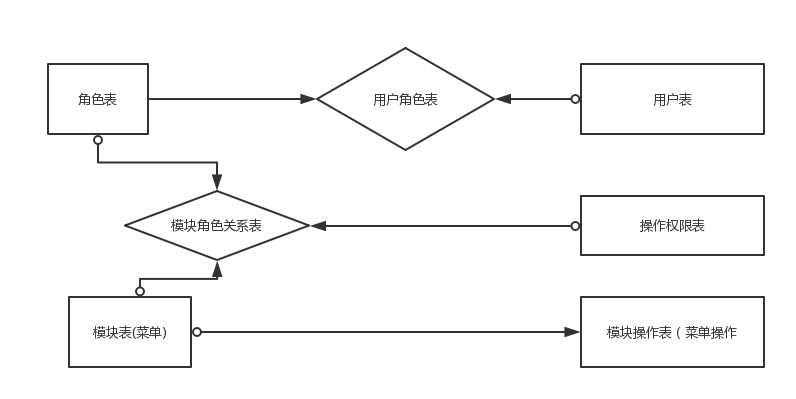


管理员用后登录，判断成功后可进入公告管理，获得公告列表，可按条件查询公告信息，对公告信息进行增删，改。

#### 角色权限设计

权限分配系统是每个系统中必不可少的最基本系统。每个角色都能拥有操作系统不同功能模块的权限，更方便于系统的管理员授权，如高级管理员可以管理和操作整个系统的所有功能。

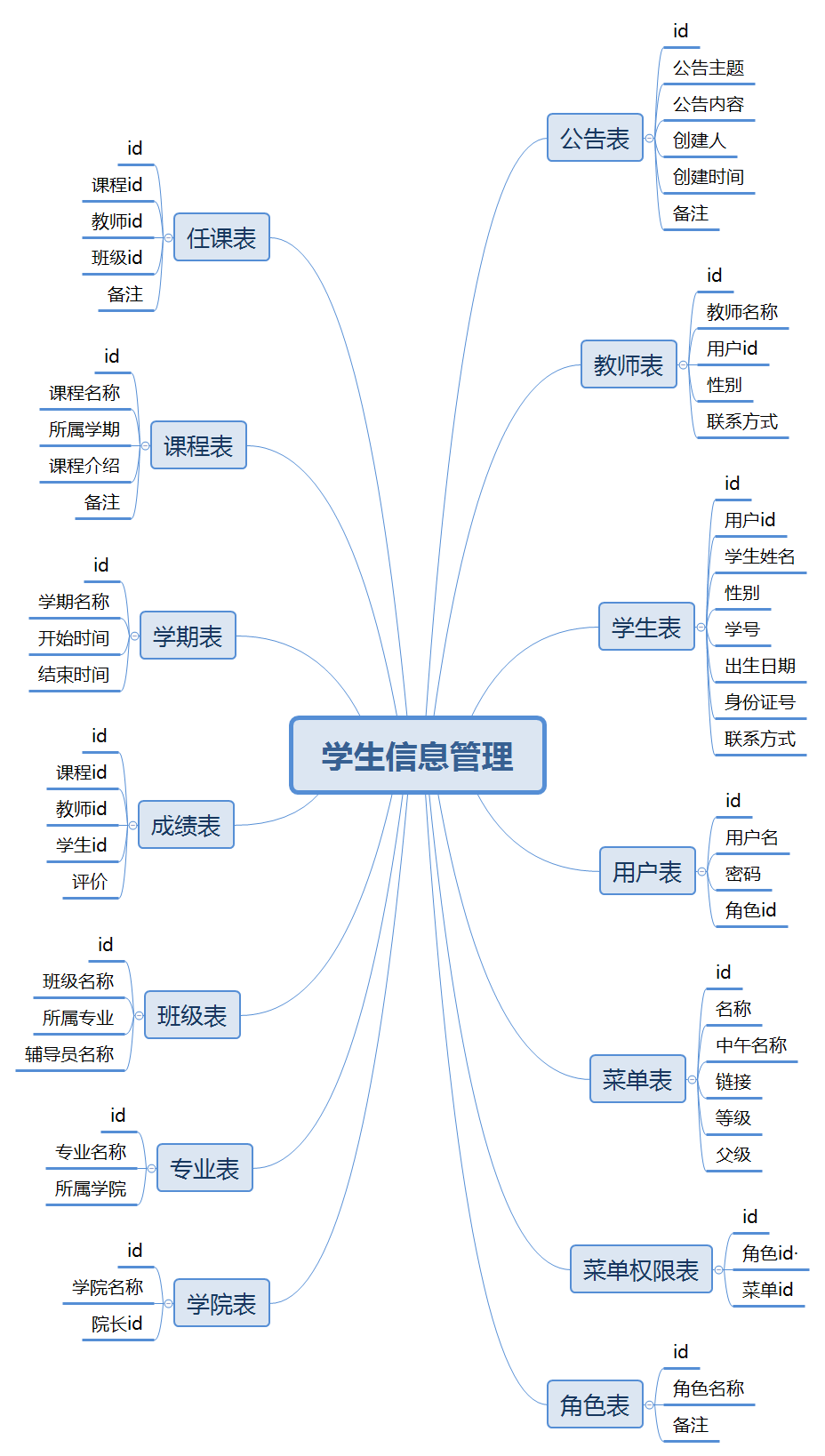
在权限设计中，有三大核心表即用户表、角色表、权限表，另外再加上用户角色和角色权限两个映射表。根据用户名查询是否拥有某个操作权限。在用户注册的时候根据其角色授予系统各个模块的权限，在用户登录后就可以给出用户所对应的权限。权限结构设计如图4-2所示：



该系统的角色可管理员用户自行添加，角色权限由管理员进行分配管理。

#### 系统数据库概要设计

系统数据库设计ER图如图:



#### 数据库的实体设计

##### 术语定义

表格“备注”列表说明：“\*”表示唯一主键，要做索引，“1++”表示值为系统自增，步长为1。

蓝色：表示该属性为外键。

表名前缀，表示table\_

V\_ : 视图前缀，表示view\_

up\_ : 存储过程前缀，表示userprocedure\_

utri\_ : 触发器前缀，表示usertrigger

pk\_ : 表列的主键约束前缀

fk\_ : 表列的外键约束前缀

ck\_ : 表列的check约束前缀

def\_ : 表列的默认约束前缀

##### 表详细设计

用户信息表T\_User:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| 用户信息ID | userID | int |  | \* |
| 用户名 | account | nvarchar | 80 |  |
| 密码 | pwd | nvarchar | 40 |  |
| 用户状态 | usertype | bit | 1 |  |
| 角色id | roleIID | int |  |  |
| 是否删除 | isdel | int |  |  |

角色信息表T\_RoleB:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| 角色ID | roleID | int |  | \* 1++ |
| 角色名称 | rolename | nvarchar | 80 |  |
| 备注 | remark | ntext |  |  |
| 是否删除 | isdel | int |  |  |

角色权限表T\_RoleSystemModel:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | id | int |  | \* 1++ |
| 角色ID | roleID | int |  |  |
| 菜单id | sysid | int |  |  |
| 是否删除 | isdel | int |  |  |

菜单表T\_SystemModel:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | id | int |  | \* 1++ |
| 名称 | name | nvarchar | 80 |  |
| 中文名称 | chinesename | nvarchar | 80 |  |
| 链接 | navurl | nvarchar | 80 |  |
| 等级 | deepth | int |  |  |
| 父级 | parentid | int |  |  |
| 序号 | displayorder | int |  |  |
| 图标链接 | imageurl | nvarchar | 80 |  |
| 是否删除 | isdelete | int |  |  |

日志信息表T\_SystemLog:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | id | int |  | \* 1++ |
| 类型 | opertype | nvarchar | 80 |  |
| 操作 | description | nvarchar | 80 |  |
| 操作地址 | opermethod | nvarchar | 80 |  |
| 请求IP | requestip | nvarchar | 80 |  |
| 异常代码 | exceptioncole | nvarchar | 80 |  |
| 异常详细信息 | exceptiondetail | nvarchar | 80 |  |
| 操作用户 | params | nvarchar | 80 |  |
| 创建时间 | createdate | datetime |  |  |
| 创建人 | createby | nvarchar | 80 |  |

学生信息表T\_Student:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | studentid | int |  | \* 1++ |
| 用户id | userID | nvarchar |  | \* |
| 学生姓名 | stuName | nvarchar | 80 |  |
| 学生性别 | stusex | nvarchar | 10 |  |
| 学生学号 | stunumber | nvarchar | 80 |  |
| 出生日期 | stubirthday | datetime | 80 |  |
| 身份证号 | stucard | nvarchar | 80 |  |
| 邮箱 | email | nvarchar | 80 |  |
| 联系方式 | phone | nvarchar | 80 |  |
| 班级 | classid |  |  |  |
| 是否删除 | isdelete | int |  |  |

教师信息表T\_ Teacher:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | teacherid | int |  | \* 1++ |
| 用户id | userID | nvarchar |  |  |
| 教师姓名 | teaName | nvarchar | 80 |  |
| 教师性别 | teasex | nvarchar | 10 |  |
| 联系方式 | teaphone | nvarchar | 80 |  |
| 确认状态 | auditstatus | bit | 1 |  |
| 是否删除 | isdelete | int |  |  |

公告表T\_ Notice:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | noticeid | int |  | \* 1++ |
| 公告主题 | noticeName | nvarchar | 80 |  |
| 公告内容 | noticecontent | text |  |  |
| 备注 | remarks | text |  |  |
| 创建人 | userID | nvarchar |  |  |
| 创建时间 | createdate | datetime |  |  |
| 是否删除 | isdelete | int |  |  |

学院表T\_ College:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | collegeid | int |  | \* 1++ |
| 学院名称 | collegeName | nvarchar | 80 |  |
| 院长 | userID | nvarchar |  |  |
| 是否删除 | isdelete | int |  |  |

专业表T\_ major:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | majorid | int |  | \* 1++ |
| 班级名称 | majorName | nvarchar | 80 |  |
| 所属学院 | collegeid | int |  |  |
| 科长 | userID | nvarchar |  |  |
| 是否删除 | isdelete | int |  |  |

班级表T\_ Class:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | classid | int |  | \* 1++ |
| 班级名称 | className | nvarchar | 80 |  |
| 所属专业 | majorid | int |  |  |
| 辅导员 | userID | nvarchar |  |  |
| 是否删除 | isdelete | int |  |  |

学期表T\_ Semester:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | semesterid | int |  | \* 1++ |
| 学期名称 | semesterName | nvarchar | 80 |  |
| 开始时间 | startdate | datetime |  |  |
| 结束时间 | enddate | datetime |  |  |
| 是否删除 | isdelete | int |  |  |

课程表T\_ Curriculum:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | curriculumid | int |  | \* |
| 课程名称 | curriculumName | nvarchar | 80 |  |
| 所属学期 | semesterid | int |  |  |
| 课程介绍 | introduce | Text |  |  |
| 备注 | remarks | text |  |  |
| 是否删除 | isdelete | int |  |  |

任课表T\_ Course:

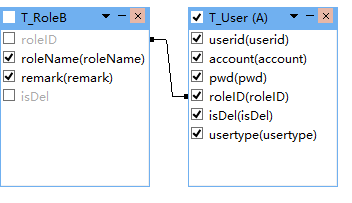
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | courseid | int |  | \* |
| 课程id | curriculumid | int |  |  |
| 教师id | teacherid | int |  |  |
| 班级id | classid | int |  |  |
| 备注 | remakes | text |  |  |
| 是否删除 | isdelete | int |  |  |

成绩表T\_ Sheet:

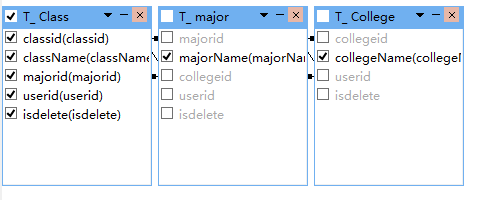
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段标识 | 类型 | 长度 | 备注 |
| ID | sheetid | int |  | \* |
| 任课id | courseid | int |  |  |
| 学生id | studentid | int |  |  |
| 分数 | fraction | doeble |  |  |
| 评价 | curriculumName | nvarchar | 80 |  |
| 是否删除 | isdelete | int |  |  |

##### 视图详细设计

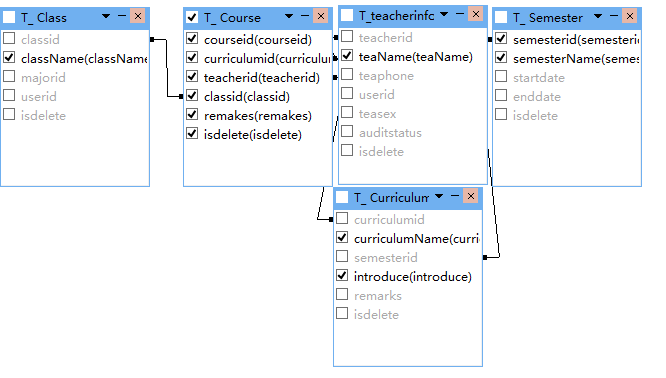
用户视图V\_ AdminUser:



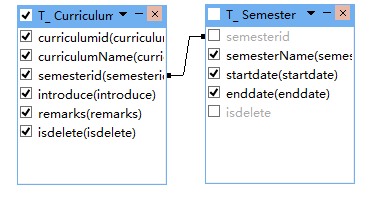
班级视图V\_ class:



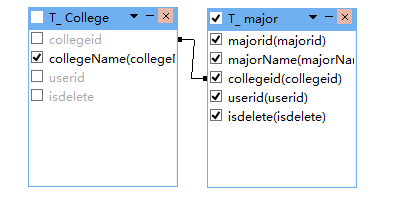
任课视图V\_Course:



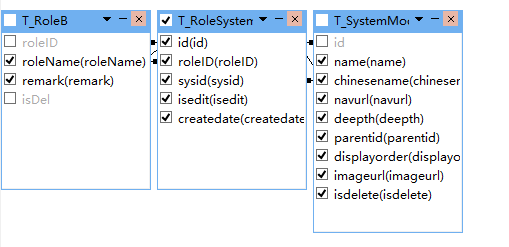
课程视图V\_Curriculum:



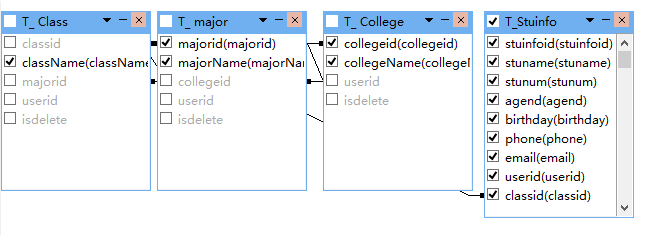
专业视图V\_major:



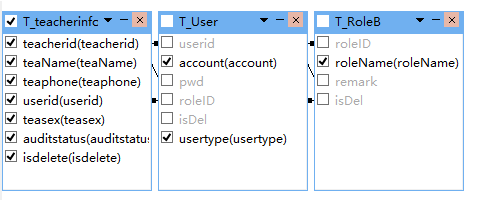
菜单视图V\_RoleSystemModel:



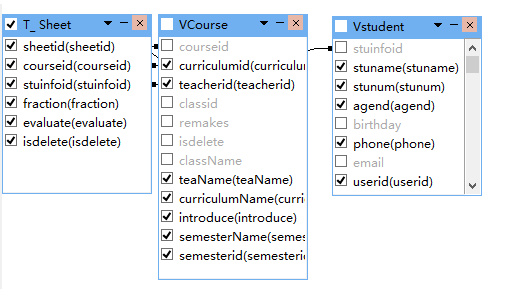
学生视图V\_student:



教师视图V\_teacher:



学生成绩视图 V\_sheet:



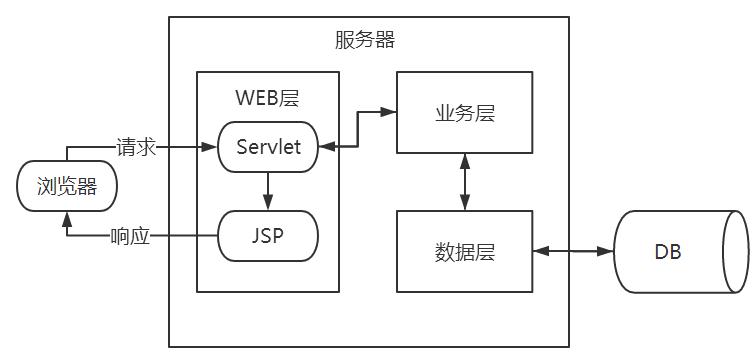
### 第五章、详细设计

#### 系统层次设计

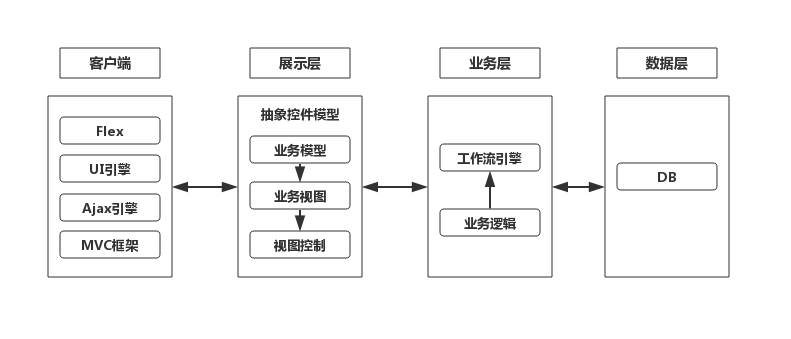
系统分层：是对模型中的包进行分组的一种特定方式。其优点在于，限制子系统间依赖关系，让开发人员能够更好的去维护系统。

系统分层的三种常用方法：

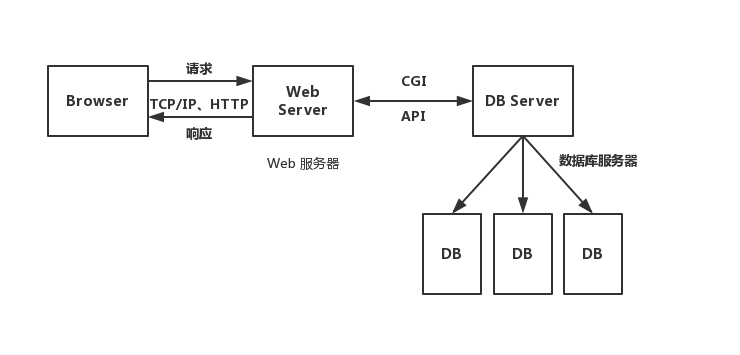
1）三层服务：用户层、业务层、数据层；



2） 多层结构的技术组成模型：展示层、业务层、数据层；



3） B/S模式结构：表现层、业务逻辑层、服务接口层、数据持久层；



分层，即将每个功能模块的代码分组。这样能更好的管理代码，使得开发更方便、快捷，实现了高内聚低耦合，不仅能降低后期的维护成本，而且使其更容易扩展。

#### 系统项目规范

##### 项目文件命名规范

项目文件命名规范如下表5-2所示：

|  |  |
| --- | --- |
| 标识符 | 说明 |
| T实体类名 | 表实体类，如T\_AdminUser表的实体类为TAdminUserModel |
| V实体类名 | 视图实体类，如V\_AdminUser视图的实体类为VAdminUserModel |
| 业务逻辑类名DAO | 业务逻辑接口类,如AdminUserDao |
| 业务逻辑类名Impl | 业务逻辑实现类，如AdminUserDaoImpl |
| 控制类类名controller | 业务控制类，如后台用户管理的控制类AdminUserController |

#### 系统功能类图设计

#### 系统业务接口设计

系统业务接口设计如下所示，表中需要实现的方法例举了出来，在接口说明中需要实现的方法会详细说明。详细说明如下：

AdminUserDAO（后台用户信息管理业务类接口）:

|  |  |  |  |
| --- | --- | --- | --- |
| 接口名 | 参数 | 返回值 | 描述 |
| getaAdminUserList | wherecondition、page、pageSize（查询条件、当前页、每页条数） | 成功返回List<VAdminUser>对象，失败返回null | 用户信息列表(带分页) |
| getaAdminUserList | roleID（角色id） | 成功返回List<TroleB>对象，失败返回null | 根据角色查询用户列表 |
| getaAdminUserList |  | 返回List<VAdminUser> | 返回所有用户列表 |
| getAdaminUserAmount | wherecondition（查询条件） | 成功返回int，失败返回0 | 返回查询到的信息数量 |
| addAdminUser | Tuser（用户列表实体类） | 成功返回true对象，失败返回false | 添加用户 |
| login | VAdminUser（查询条件） | 成功返回VAdminUser对象，失败则null | 用户登录 |
| delAdminUser | Tuser（用户表实体类） | 成功返回true对象，失败返回false | 删除用户信息 |
| updatePwd | userid,pwd（用户名，密码） | 成功返回true对象，失败返回false | 修改用户密码 |
| changeState | userid(用户名) | 成功返回true对象，失败返回false | 修改用户状态 |
| getuser | userid(用户名) | 成功返回Tuser对象，失败返回null | 根据用户名查询信息 |
| update | Tuser（用户表实体类） | 成功返回true对象，失败返回false | 修改用户信息 |

AdminRoleDAO（后台角色信息管理业务类接口）:

|  |  |  |  |
| --- | --- | --- | --- |
| 接口名 | 参数 | 返回值 | 描述 |
| getRolelist | Opreation（查询条件） | 成功返回List<TroleB>，失败返回null | 查询用户角色信息 |
| delAdminRole | TroleB（角色实体类） | 成功返回true，失败返回false | 删除角色信息 |
| addAdminRole | TroleB（角色实体类） | 成功返回true，失败返回false | 添加角色信息 |
| edlAdminRole | TroleB（角色实体类） | 成功返回true，失败返回false | 修改角色信息 |

CollegeDAO（后台学院信息管理业务类接口）:

|  |  |  |  |
| --- | --- | --- | --- |
| 接口名 | 参数 | 返回值 | 描述 |
| getCollegeList | wherecondition、page、pageSize（查询条件、当前页、每页条数） | 成功返回List<TCollege>对象，失败返回null | 学院信息列表(带分页) |
| getCollegeList | wherecondition（查询条件） | 成功返回int，失败返回0 | 返回查询到的信息数量 |
| getCollegeList |  | 成功返回List<TCollege>对象，失败返回null | 返回所有学院信息列表 |
| addCollege | TCollege（学院实体类） | 成功返回true，失败返回null | 添加学院信息 |
| delCollege | TCollege（学院实体类） | 成功返回true，失败返回null | 删除学院信息 |
| update | TCollege（学院实体类） | 成功返回true，失败返回null | 修改学院信息 |

NoticeDAO（后台公告信息管理业务类接口）:

|  |  |  |  |
| --- | --- | --- | --- |
| 接口名 | 参数 | 返回值 | 描述 |
| getNoticeList | wherecondition、page、pageSize（查询条件、当前页、每页条数） | 成功返回List< TNotice>对象，失败返回null | 公告信息列表(带分页) |
| getNoticeList | wherecondition（查询条件） | 成功返回int，失败返回0 | 返回查询到的信息数量 |
| addNotice | TNotice（公告实体类） | 成功返回true，失败返回null | 添加公告信息 |
| delNotice | TNotice（公告实体类） | 成功返回true，失败返回null | 删除公告信息 |
| update | TNotice（公告实体类） | 成功返回true，失败返回null | 修改公告信息 |

CurriculumDAO（后台课程信息管理业务类接口）:

|  |  |  |  |
| --- | --- | --- | --- |
| 接口名 | 参数 | 返回值 | 描述 |
| getVcurrList | wherecondition、page、pageSize（查询条件、当前页、每页条数） | 成功返回List< VCurriculum >对象，失败返回null | 课程信息列表(带分页) |
| getVcurrList | wherecondition（查询条件） | 成功返回int，失败返回0 | 返回查询到的信息数量 |
| getVcurrList |  | 成功返回List< VCurriculum >对象，失败返回null | 返回所有课程信息列表 |
| addTCurr | TCurriculum（课程实体类） | 成功返回true，失败返回null | 添加课程信息 |
| delTCurr | TCurriculum（课程实体类） | 成功返回true，失败返回null | 删除课程信息 |
| update | TCurriculum（课程实体类） | 成功返回true，失败返回null | 修改课程信息 |

CourseDAO（后台任课信息管理业务类接口）:

|  |  |  |  |
| --- | --- | --- | --- |
| 接口名 | 参数 | 返回值 | 描述 |
| getCourseList | wherecondition、page、pageSize（查询条件、当前页、每页条数） | 成功返回List< VCourse >对象，失败返回null | 任课信息列表(带分页) |
| getCourseList | wherecondition（查询条件） | 成功返回int，失败返回0 | 返回查询到的信息数量 |
| getCourseList |  | 成功返回List< VCourse >对象，失败返回null | 返回所有任课信息列表 |
| getCourseList | teacherid,classid(教师id,班级id) | 成功返回List< VCourse >对象，失败返回null | 根据条件返回任课信息列表 |
| getVsheetList | stuinfoid,ourseid(教师id,班级id) | 成功返回List< Vsheet >对象，失败返回null | 根据条件返回成绩信息列表 |
| addTCourse | TCourse（任课实体类） | 成功返回true，失败返回null | 添加任课信息 |
| delTCourse | TCourse（任课实体类） | 成功返回true，失败返回null | 删除任课信息 |
| update | TCourse（任课实体类） | 成功返回true，失败返回null | 修改任课信息 |

StudentDAO（后台学生信息管理业务类接口）:

|  |  |  |  |
| --- | --- | --- | --- |
| 接口名 | 参数 | 返回值 | 描述 |
| getVstuList | wherecondition、page、pageSize（查询条件、当前页、每页条数） | 成功返回List< Vstudent >对象，失败返回null | 学生信息列表(带分页) |
| getVstuList | wherecondition（查询条件） | 成功返回int，失败返回0 | 返回查询到的信息数量 |
| getVstuList | wherecondition（查询条件） | 成功返回List< Vstudent >，失败返回null | 返回查询到的信息数量 |
| getsheetList | wherecondition、page、pageSize（查询条件、当前页、每页条数） | 成功返回List< Vsheet >对象，失败返回null | 学生成绩信息列表(带分页) |
| getsheetList | wherecondition（查询条件） | 成功返回int，失败返回0 | 返回查询到的信息数量 |
| getVstudentList | classid(班级id) | 成功返回List< Vstudent >对象，失败返回null | 根据班级查询学生信息 |
| addsheet | TSheet（学生成绩实体类） | 成功返回true，失败返回null | 添加学生成绩信息 |
| upsheet | TSheet（学生成绩实体类） | 成功返回true，失败返回null | 修改学生成绩信息 |
| getSheet | id（学生成绩id） | 成功返回TSheet，失败返回null | 获取学生成绩信息 |
| getstu | id（学生id） | 成功返回TStuinfo，失败返回null | 获取学生成绩信息 |
| addStu | TStuinfo（学生实体类） | 成功返回true，失败返回null | 添加学生信息 |
| delStu | TStuinfo（学生实体类） | 成功返回true，失败返回null | 删除学生信息 |
| update | TStuinfo（学生实体类） | 成功返回true，失败返回null | 修改学生信息 |

StudentDAO（后台学生信息管理业务类接口）:

|  |  |  |  |
| --- | --- | --- | --- |
| 接口名 | 参数 | 返回值 | 描述 |
| getteacherList | wherecondition、page、pageSize（查询条件、当前页、每页条数） | 成功返回List< VTeacher >对象，失败返回null | 教师信息列表(带分页) |
| getteacherList | wherecondition（查询条件） | 成功返回int，失败返回0 | 返回查询到的信息数量 |
| getteacherList |  | 成功返回List< VTeacher >对象，失败返回null | 返回所有教师信息列表 |
| getVteacherList | userid（用户名） | 成功返回List< VTeacher >对象，失败返回null | 根据用户名查询教师 |
| addTteacher | Tteacher（教师实体类） | 成功返回true，失败返回null | 添加教师信息 |
| delTteacher | Tteacher（教师实体类） | 成功返回true，失败返回null | 删除教师信息 |
| changeState | Teacherid（教师id） | 成功返回true，失败返回null | 修改教师状态 |
| update | TCurriculum（教师实体类） | 成功返回true，失败返回null | 修改教师信息 |

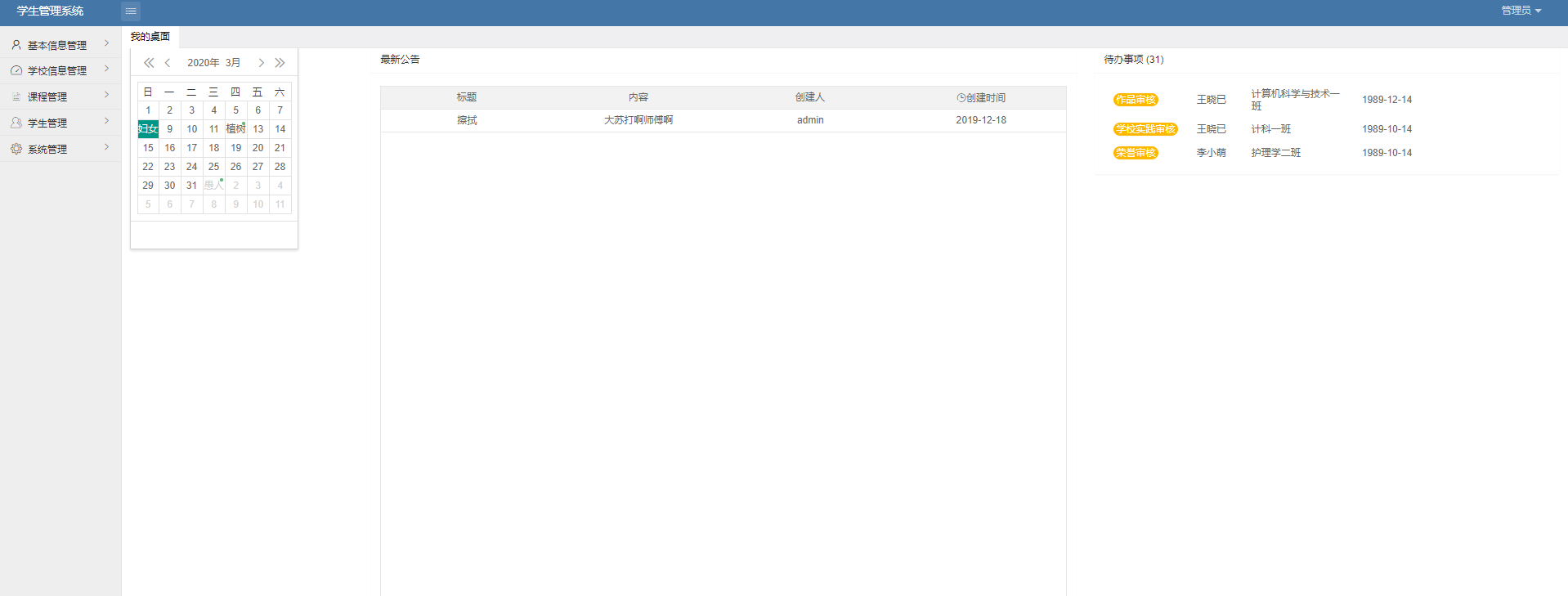
#### 系统交互动作接口设计

系统交互动作接口设计信息如下所示，在控制类说明、方法名和方法功能中都有详细的文字描述。

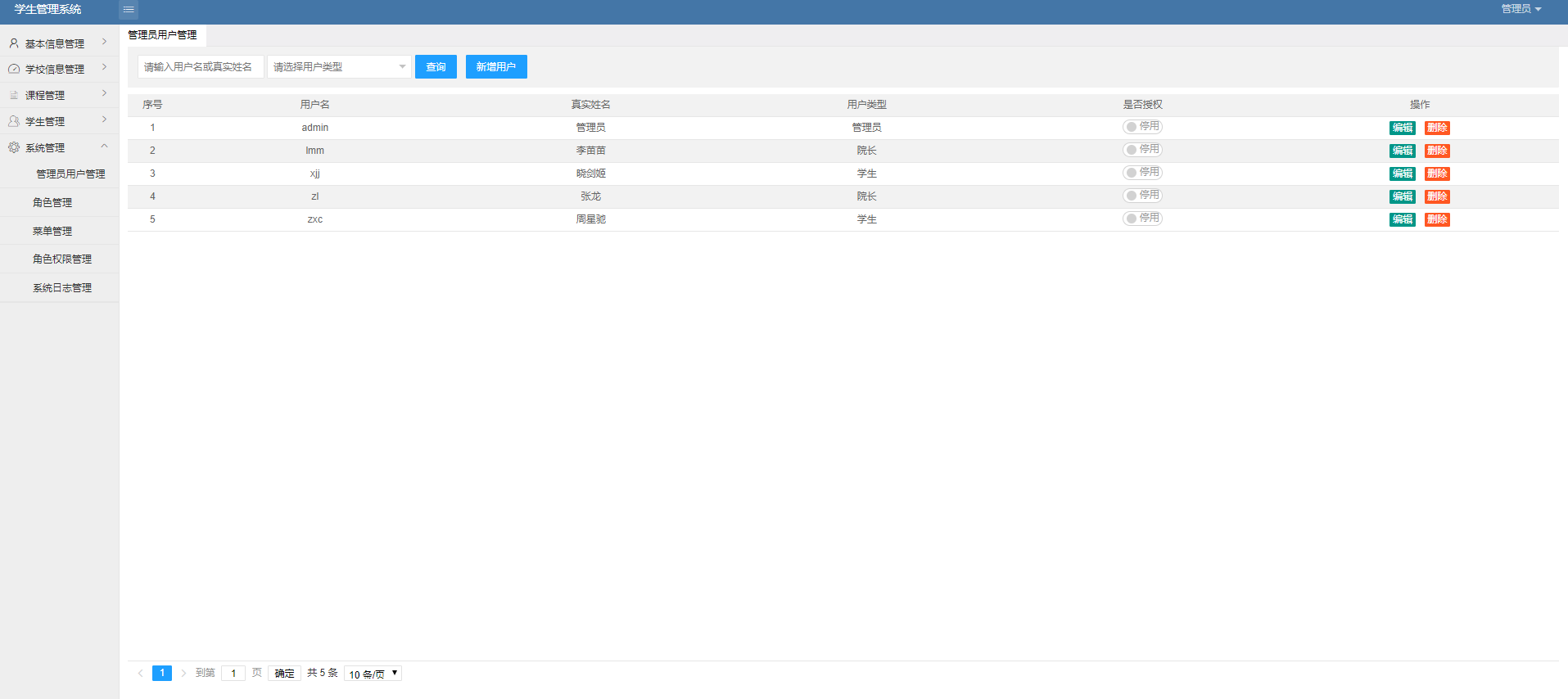
|  |  |  |  |
| --- | --- | --- | --- |
| 控制类名称 | 控制类说明 | 请求方法名 | 方法功能 |
| AdminControllerr  url=”/ admin2  ” | 管理员用户处理类 | getNoticeList (HttpServletRequest request, intpage,intlimit,tringnoticename,HttpServletResponseresponse, Model model)  url=”/getnotice” | 公告信息分页 |
| addNotice(String noticeName,String noticecontent,String remarks,HttpServletRequest request,HttpServletResponse response, Model model)  url=” deleteAdminRole” | 添加公告信息 |
| getTeacherList (HttpServletRequest request, intpage,intlimit,tringnoticename,HttpServletResponseresponse, Model model) url=”/getnotice” | 教师信息分页 |
| Addteacher(String teaName,String teaphone,String userid,String teasex, HttpServletRequest request,  HttpServletResponse response, Model model)  url=” addteacher” | 添加教师信息 |
| changeUser(String userid, HttpServletRequest request,HttpServletResponse response, Model model)  url=” changeuser” | 修改用户状态 |
| GetLoacdtear(HttpServletRequestrequest,HttpServletResponse response, Model model)  url=” loacdtea” | 加载教师下拉框 |
| CollegeController  url=”/ college  ” | 学院信息处理类 | getCollegeList(HttpServletRequest request, intpage,intlimit,tringnoticename,HttpServletResponseresponse, Model model)  url=”/ getcollege” | 学院信息分页 |
| addCollege(String collegeName,String userid,  HttpServletRequest request,  HttpServletResponse response, Model model)  url=”/ getcollege” | 添加学院信息 |
| upCollege(Integer collegeid,String collegeName,String userid,  HttpServletRequest request,  HttpServletResponse response, Model model)  url=”/ upcollege” | 修改学院信息 |
| GetLoacduser(HttpServletRequestrequest,HttpServletResponse response, Model model)  url=” loacduser” | 加载用户下拉框 |
| CurrContrller  url=”/ curr | 课程信息处理类 | getcurrList(HttpServletRequest request, intpage,intlimit,tringnoticename,HttpServletResponseresponse, Model model)  url=”/ getcurr” | 课程信息分页 |
| addcurr(String curriName,Integer semesterid,String introduce, String remarks, HttpServletRequest request,  HttpServletResponse response, Model model)  url=”/ addcurri” | 添加课程信息 |
| GetLoacdcurr(HttpServletRequestrequest,HttpServletResponse response, Model model)  url=”/ loacdcurr” | 加载课程下拉框 |
| GetLoacdcurrby(HttpServletRequest request,Integer classid,HttpServletResponse response, Model model)  url=”/ loacdcourseby” | 根据班级获取课程 |

#### 系统UI设计

本系统UI应用Layui框架技术，首页如下图所示：



功能模块页面：



### 第六章、系统的实现与测试

#### 环境搭建

1.下载JDK. 网址:<http://www.oracle.com/technetwork/java/javase/downloads/index.html>,

2 安装jdk,

3.安装好jdk后配置环境变量, .安装完成后，右击"我的电脑"，点击"属性"，选择"高级系统设置"；选择高级选项卡，点击环境变量在 "系统变量" 中设置 3 项属性，JAVA\_HOME、PATH、CLASSPATH(大小写无所谓),若已存在则点击"编辑"，不存在则点击"新建"。

如果使用 1.5 以上版本的 JDK，不用设置 CLASSPATH 环境变量，也可以正常编译和运行 Java 程序。

设置3个系统变量，参数如下：

变量名：JAVA\_HOME

变量值：D:\Program Files\Java\jdk-12.0.2 // 要根据自己的实际路径配置

变量名：CLASSPATH

变量值：.;%JAVA\_HOME%\lib\dt.jar;%JAVA\_HOME%\lib\tools.jar; //记得前面有个"."

变量名：Path

变量值：%JAVA\_HOME%\bin;%JAVA\_HOME%\jre\bin;

#### 数据层的实现

本课题数据库采用navicat 服务器上的Mysql，数据库名称：StudentDB，链接名：swg，IP：116.62.14.178；

#### 数据连接层的实现

该系统数据连接层访问数据库采用Hibernate技术，hibernate.cfg.xml为数据连接层的配置文件。



#### 基于Hibernate的数据库连接创建与调用

示例代码：



#### 数据操作层的实现

数据操作层，采用hibernate框架技术，封装了对数据的增删查改的方法，便于业务逻辑层的调用。

类名：HibBaseDAO

示例代码：



#### 数据操作层的调用实现

业务层根据业务模块先定义DAO业务接口，再实现DAO以创建IMPL业务实现类。

类名：AdminuserDaoImpl

示例代码：



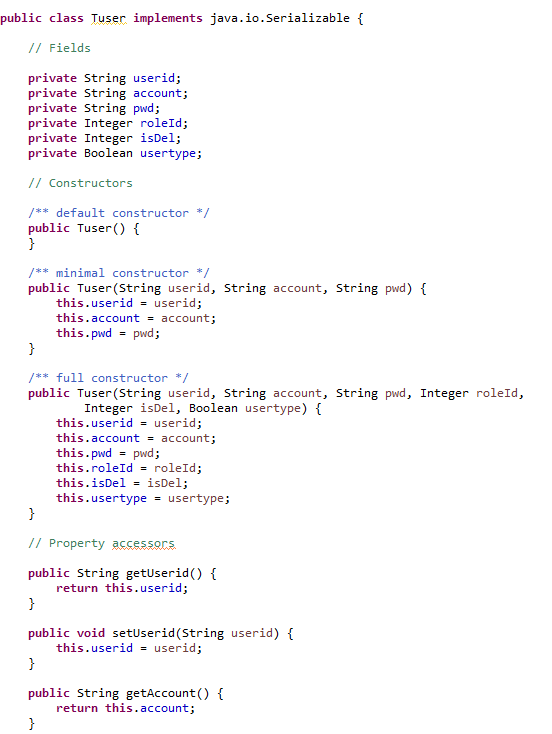
#### 实体交换层的实现

实体交换层，即模型层，用于存储数据对象，进行数据的存储和传输，使用Hibernate可以直接逆向工程将表生成实体对象，实体对象一般放在model层。

生成的实体层的.xml文件Tuser.hbm.xml实例代码：



实体层TTask类实例代码：



#### 系统业务层的实现

系统业务层接口调用model层。

类名：AdminUserDAO

示例代码：



#### 信息业务层的调用实现

类名：DAOFactory

业务层的的调用代码示例：



#### 控制层的实现

控制层采用springMVC框架技术，利用注解实现URL的定义。

类名：TaskController

示例代码：



#### 控制层的调用实现

添加信息类型控制层的调用采用ajax，代码示例如下：



#### 表示层的实现

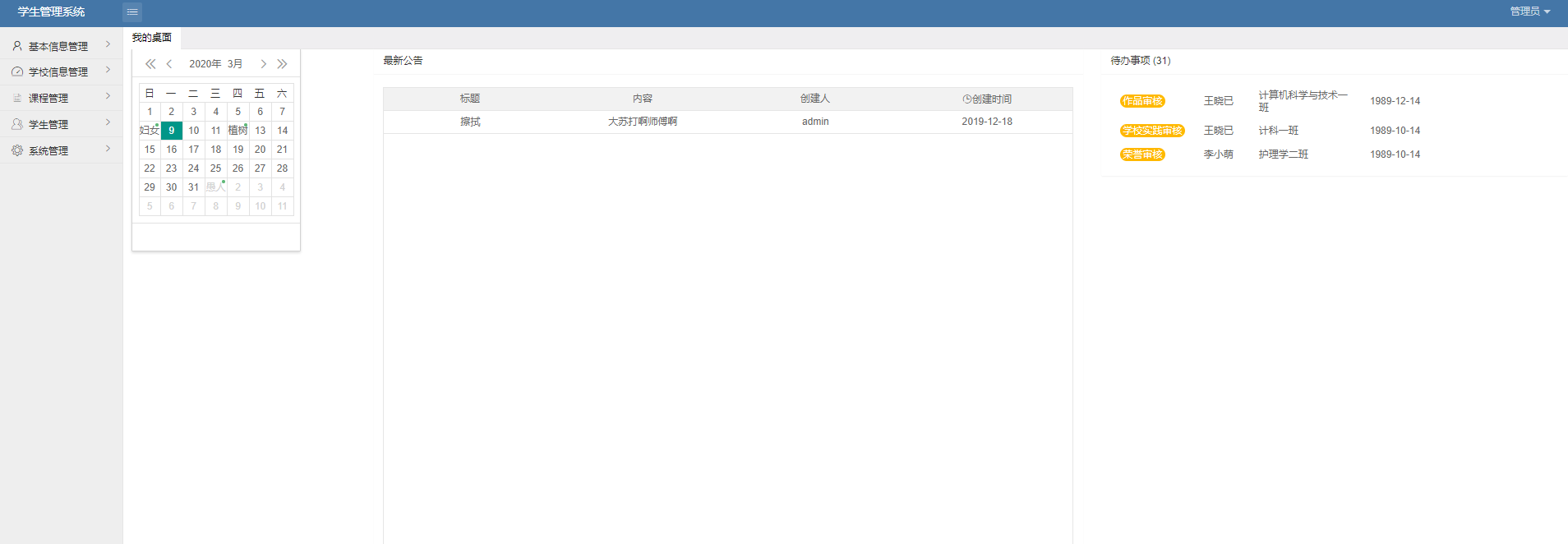
##### 代码

表示层由html页面组成，主要采用LayUI框架编写。

代码示例：



##### 实用结果



#### 软件测试用例说明

##### 测试目的

测试检查系统是否能按照预期的效果运行；

检测系统运行的时候是不是具有良好的可操作性和较美观的界面效果，是不是能最大程度的满足用户的使用要求，通过测试检查和发现系统的不足之处，然后改进完善。

##### 网站端功能测试用例

网站登录功能测试如表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 前提条件 | 输入动作 | 期望输出 | 实际显示 | 结果 |
| 输入正确的用户名、密码、验证码 | 输入“admin”，密码“123456” | 跳转网站主界面 | 跳转成功 | 成功 |
| 输入错误的密码 | 输入“admin”，密码“123” | 提示用户名或者密码输入错误 | 提示用户名或密码错误 | 成功 |
| 输入未赋予权限的教师用户 | 输入“lmm”，密码“123456” | 提示用户登录失败 | 提示用户登录失败 | 成功 |

网站权限功能测试如表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 前提条件 | 输入动作 | 期望输出 | 实际显示 | 结果 |
| 输入管理员用户 | 输入“admin”，密码“123456” | 跳转网站主界面，所有权限 | 跳转网站主界面，所有权限 | 成功 |
| 输入教师用户 | 输入“lmm”，密码“123456” | 跳转网站主界面，教师权限 | 跳转网站主界面，教师权限 | 成功 |
| 输入学生用户 | 输入“xjj”，密码“123456” | 跳转网站主界面，学生权限 | 跳转网站主界面，学生权限 | 成功 |

任课功能测试如表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 前提条件 | 输入动作 | 期望输出 | 实际显示 | 结果 |
| 输入正确的班级，教师，课程 | 班级“16级汉语言一班”，教师“lmm” ,课程“数学” | 添加成功 | 任课添加成功 | 成功 |
| 输入重复的班级，教师，课程 | 班级“16级汉语言一班”，教师“lmm” ,课程“数学” | 教师记录已存在 | 教师记录已存在 | 成功 |

修改教师状态功能测试如表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 前提条件 | 输入动作 | 期望输出 | 实际显示 | 结果 |
| 点击教师启用 | 点击教师启用 | 教师启用成功 | 教师启用成功 | 成功 |

学生成绩评定功能测试如表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 前提条件 | 输入动作 | 期望输出 | 实际显示 | 结果 |
| 教师用户登录 | 点击成绩表格输入“80”,评定“优秀” | 数据保存数据库，无输出 | 数据保存数据库，无输出 | 成功 |

角色功能测试如表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 前提条件 | 输入动作 | 期望输出 | 实际显示 | 结果 |
| 管理员登录进行添加 | 角色名称“测试”，描述“测试” | 角色添加成功 | 角色添加成功 | 成功 |

角色功能测试如表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 前提条件 | 输入动作 | 期望输出 | 实际显示 | 结果 |
| 管理员登录进行修改 | 查询院长角色，授予“系统管理权限” | 授权成功，请刷新页面 | 授权成功，请刷新页面 | 成功 |
| 管理员登录进行修改 | 查询院长角色，取消授予“系统管理权限” | 取消授权成功，请刷新页面 | 取消授权成功，请刷新页面 | 成功 |

#### 测试报告

网站功能已达到预期目标，所测试功能皆无问题。

### 第七章、总结

#### 总结

#### 展望

结 论 （与正文空一行，三号黑体居中）

XXXXXXXXXXXXXXX（小四宋体，与正文要求一致）

致 谢（与正文空一行，三号黑体居中）

XXXXXXXXXXXXXXX（小四宋体，与正文要求一致）

参考文献（黑体三号，居中）

**一、中文部**分

[1] 易百教程.

[2]吴海霞，沈剑平．电视论坛[C] . 北京：人民教育出版社，2003．（2）：

[3]…… （宋体5号，固定值20磅行距）

**二、英文**部分

[1]Chow, W. S., Shyu, J. C. and Wang, K. C.Developing A Forecast System for Hotel Occupancy Rate Using Integrated ARIMA Models[J]. *Journal of International Hospitality, Leisure & Tourism Management*, 1998.(2)：

[2]……. （Times New Roman11号，固定值20磅行距）

附 录 （黑体三号，居中）

若附录有好几项，则每项附录均另起一页。

注：文档结构规范和基本要求只是一个最基本的规定，对不同专业也可能有不太适合的方面。各分院可根据本专业特点对其中各项进行更为合理的要求和规定，但全分院必须统一格式且符合规范。