202 －202 学年第 学期

编号：



### 大 作 业 报 告

课 程 名 软件架构与应用开发

课 程 号 B01077

学 生 姓 名 杨晨露

学 生 姓 号 32001298

专 业 班 级 计算机2001

所 在 学 院 计算学院

指 导 老 师 罗荣良

实验报告日期： 年 月 日

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 课程名 | 软件架构与应用开发 | | 课程号 | B01077 |
| 实验项目名称 | 期末大作业 | | | |
| 实验时间 |  | | | |
| 小组合作：■是 □否 | | 小组成员： | | |

## 实验目的和要求（或设计要求及指标）

【开发说明】

构建一体化的教育网络环境，集教育管理、教学资源、网络资源及运行保障为一体的教育云平台，云平台支持集群化部署、弹性伸缩、图形化展示等特性，提供一整套高可靠性、高扩展性、易管理的教育云解决方案。

教学管理系统以支持信息技术专业教学、实验为主，其他专业为辅，支持的专业越多越好，灵活性强，可以应对专业的调整和课程的变化。

【技术要求与指标】

（1）云计算平台部署：

①技术不限，可使用开源软件；

②支持集群化部署；

③支持虚拟化技术，构建虚拟资源组成的资源池，包括计算资源、存储资源等；

④支持图形化的展示，提供可视化的资源统一的监控界面。

（2）教学管理系统部署：

①支持B/S 架构，无需安装客户端软件；

②涵盖课程管理、用户管理、排班管理、课程作业管理、实验管理等管理功能，通过各个功能模块的相互协作形成平台中的流程闭环，让高校师生以及管理人员更加便捷的进行教学实验科研；

③支持专业课程资源的集中管理（如：文档课件、视频、实验资源等），支持用户与云平台的对接；

④支持教师通过平台进行理论教学和实验实训教学，并支持提供指导手册、实验步骤、实验软件包、实验报告和课后作业等功能；

⑤支持学生在线实验实训，并通过底层云计算平台提供课程实验实训所需IT环境。

（3）其他创新性功能：

①团队自由发挥设计。

【任务清单】

（1）整体方案设计；

（2）完成基础环境搭建；

（3）完成部署教学科研应用系统；

（4）测试验证主要功能和创新成果。

【开发工具与数据接口】

开发工具及开发平台不限，可借助开源工具。系统可正常部署与运行，系统运行日志不报严重错误，一二级主要功能都正常。

## 实验内容和原理（或设计方案及原理）

**第十二届中国大学生服务外包创新创业大赛**

**A19基于云计算平台构建教学管理系统【华云数据】**

**A**

1. 项目背景

1.1提出原因

党的二十大报告将科技、教育、人才统筹部署，集中表达了科技对教育发展的支撑作用，以及教育优先发展对国家拔尖创新人才培养的重大意义。围绕新一代互联网革命浪潮，教育系统在科技创新和人才培育上将迎来巨大发展机遇。因此我组选择题目A19——基于云计算平台构建教学管理系统来作为赛题，顺应“互联网+教育”的时代导向。

1.2环境背景

近年来我国大力引进以智能算法为方法论的教育评价理念和手段。然而，现有教育评价方式尚未确立语义算法地位，评价方式和标准依然以统一规范为基础，大多局限在自动判卷维度，对于评价方式模型缺少理论分析。

中共中央国务院印发《深化新时代教育评价改革总体方案》要求，改变相对固化的试题形式，完善学生综合素质档案，逐步转变简单的以考试成绩为唯一标准的招生模式，这与国家确立人才的主体地位和创造潜能高度。这要求我们合理开发新的平台，建立以人才成长为中心的分类评价模式。

1.3操作可行性

“互联网+”技术和教育行业的融合，促进高校形成更加科学的教学评价。传统的教学评价，主要是通过考试成绩进行，是结果性评价，缺乏对学生的过程性评价，这是因为缺乏有效的过程考核技术，而随着互联网技术在教育教学中的应用，教师可以借助互联网技术对学生的学习过程进行考核，从而实现对学生全面科学的评价。例如，教师可以借助互联网平台来对学生的学习过程进行全面地跟踪，了解学生的作业完成情况、单元小测情况等。

（二）项目分析与设计

2.1项目设计

①支持B/S 架构，无需安装客户端软件；

②涵盖课程管理、用户管理、排班管理、课程作业管理、实验管理等管理功能，通过各个功能模块的相互协作形成平台中的流程闭环，让高校师生以及管理人员更加便捷的进行教学实验科研；

③支持专业课程资源的集中管理（如：文档课件、视频、实验资源等），支

持用户与云平台的对接；

④支持教师通过平台进行理论教学和实验实训教学，并支持提供指导手册、实验步骤、实验软件包、实验报告和课后作业等功能；

⑤支持学生在线实验实训，并通过底层云计算平台提供课程实验实训所需IT环境。

2.2 项目分析

项目需要涵盖课程管理、用户管理、排班管理、课程作业管理、实验管理等管理功能，通过各个功能模块的相互协作形成平台中的流程闭环，让高校师生以及管理人员更加便捷的进行教学实验科研。

为涵盖教学教育环节内的整体过程，我们设计将用户分成四大类，分别是系统管理员、教务人员、教师、学生。系统管理员主要负责整体系统的维护，可实现所有功能的数据修改，管理员为其他群体账户负责，可进行信息修改与重置密码，同时可进行系统日志等信息的维护。教务人员主要负责排课排班，发放工资，统筹学生成绩与获奖情况等，而教师与学生系统可查看个人课表以及进行课程资料和学习情况的交互与反馈。

（三）技术路线及实现方案

3.1前端设计

3.1 .1 vue

Vue是一套用于构建用户界面的渐进式的JS框架，只关注视图层，采用自底向上增量开发的设计，Vue框架采用MVVM模式，实现数据的双向绑定，能将后端传递的数据转化成所看到的页面，也能将所看到的页面转化成后端的数据。Vue使用简单，运行速度较快，易于与第三方库整合。

3.1.2 采用组件化模式，提高代码的复用率，让代码易于维护。

将页面拆分成不同的部分，每个部分为一个组件，每个组件为一个单独的vue文件，每个vue文件的标签、样式、交互进行分开管理。提高代码的复用率：当其他的页面也需要使用相同的组件时，直接引入写好的对应的组件即可。

代码易于维护：每个vue文件不同组件的标签、样式、交互进行单独管理，不会互相影响。

3.2后端设计

3.2.1 SSM框架

SSM框架是spring、spring MVC 、和mybatis框架的整合，是标准的MVC模式。标准的SSM框架有四层，分别是dao层（mapper），service层，controller层和View层。使用spring实现业务对象管理，使用spring MVC负责请求的转发和视图管理，mybatis作为数据对象的持久化引擎。

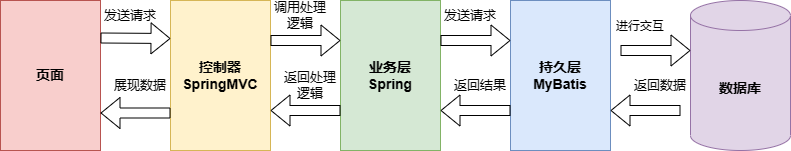


图1 SSM框架逻辑图

3.2.2 Tomcat服务器

Tomcat简单的说就是一个运行JAVA的网络服务器，底层是Socket的一个程序，它也是JSP和Serlvet的一个容器。Tomcat包括服务器Server，服务Service，连接器Connector、容器Container。连接器Connector和容器Container是Tomcat的核心。

一个Container容器和一个或多个Connector组合在一起，加上其他一些支持的组件共同组成一个Service服务，有了Service服务便可以对外提供能力了，但是Service服务的生存需要一个环境，这个环境便是Server，Server组件为Service服务的正常使用提供了生存环境，Server组件可以同时管理一个或多个Service服务。

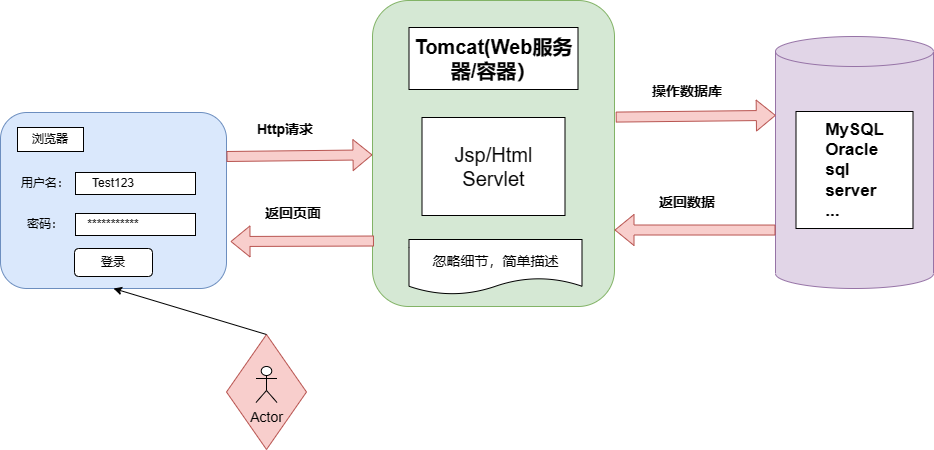


图2 TomCat作用示意图

3.2.3数据存储

MySQL数据库是一个它是一个轻量级的关系型数据库管理系统，使用了 ORACLE的部分功能，但又有自己独特的设计。SQL数据库是一个开放源代码的、用于关系型数据库管理系统（RDMS）的应用服务器软件，它被广泛地应用于个人计算机系统中。MySQL提供了一个简单、灵活和开放的源代码版本，其包含了许多有用的特性，如事务管理、数据加密、安全存储等。项目采用MySQL数据库用来存储基础数据和系统更新数据。

（四）项目可行性分析

4.1市场分析

4.1.1云计算

摘要：云计算是一种新兴的计算模式，它以强大的网络计算能力和丰富的存储资源，为用户提供高效、安全、灵活的服务。本文介绍了云计算的相关理论，分析了其优点和特点，并结合教学管理系统现状，提出了基于云计算平台构建教学管理系统的解决方案，最后指出云计算在教学管理系统中的应用前景。

云计算是指通过网络以按需、易扩展的方式获取所需资源的计算模式，云计算最早是由 Google在2010年提出的，并在2011年提出了“云计算”的概念。它是一种基于互联网的计算方式，其实质是通过网络以按需、易扩展的方式获取所需资源。其最大的特点就是通过网络将服务器等硬件资源共享，通过虚拟化技术将这些资源整合成一个可供用户使用的资源池，并提供相应的服务。云计算以其强大的网络计算能力和丰富的存储资源，为用户提供高效、灵活、安全、可靠、可控等服务，它改变了传统 PC和服务器应用软件，从而使人们可以随时随地访问所需应用程序和数据。

4.1.2教育管理系统现状

随着时代的发展，我国科学技术日新月异，每次亮相都让人眼前一亮。科学技术推动着高校信息化建设，以教学为主体的教学管理信息化建设成为重中之重的任务。高校教学管理信息建设依赖于大数据、云计算、人工智能等信息化前沿技术。

教育行业的信息化程度普遍不高，主要原因是教育行业的特殊性。首先是行业性质，教育行业涉及到众多的人群，各区域内的学校和培训机构分布较为分散，且各区域内的学校和培训机构之间在管理、运营等方面存在较大差异，难以实现信息的共享与整合。其次是管理模式。目前大多数教育行业的管理模式还处于粗放式管理阶段，而随着市场竞争的不断加剧，以及新技术和新产品的不断涌现，传统管理方式已经难以适应现代化管理的需要。

互联网技术、移动设备的发展、终端设备的普及和5G技术应用等，为教育信息化建设提供了良好的基础设施条件和技术保障。

目前，教学管理系统大多采用基于 Windows平台的B/S结构，这种方式部署方便，扩展性好。由于B/S模式是基于 Internet的，这就使得所有的用户都可以通过浏览器访问和使用网络，由于浏览器只负责显示界面，因此它不需要负责网络安全问题。而C/S模式是在客户端运行的，所以它的安全性就要差得多。另外，B/S模式可以让用户使用任何一台计算机上网访问信息系统，而C/S模式则不可以。同时B/S结构是基于 Internet的，所以它能实现真正意义上的完全分布式。在C/S结构下，由于服务器只能与一个用户交互，而当服务器处理用户提交的请求时，一般都要把数据发送到多个数据库中去处理。

4.2可行性

4.2.1技术可行性

技术可行性是指一个产品或者服务从设计到制造的过程中所需要的技术，其目的是为了满足该产品或者服务的功能需求和非功能需求。在项目开始之前，我们需要对所需技术进行评估，以确定一个项目是否具备技术可行性。

4.2.1.1项目采用ssm框架，即spring、spring MVC 、和mybatis框架的整合，是标准的MVC模式。标准的SSM框架有四层，分别是dao层（mapper），service层，controller层和View层。使用spring实现业务对象管理，使用spring MVC负责请求的转发和视图管理，mybatis作为数据对象的持久化引擎。在这个快速发展的互联经济的时代，SSM框架提高了开发人员的工作效率，及时的开发出具有良好的时效性的优秀的软件，这是开发人员此时更加需要的。

4.2.1.2MySQL数据库存放项目的各种基础数据和测试数据。使用 MySQL作为后台数据库是非常方便的，因为它支持多种数据格式（比如： Txt、 HTML、 XML等）和多种语言（比如： PHP、 ASP、。 NET等），并且它可以对数据库中的数据进行操作。

4.2.1.3项目采用tomcat作为服务器，运行JAVA的网络服务器，底层是Socket的一个程序，它也是JSP和Serlvet的一个容器。tomcat可以提供能够让别人访问自己写的页面的一个程序。tomcat技术先进、性能稳定，而且免费。也是目前比较流行的Web 应用服务器。

4.2.2使用可行性

①页面美观

主题明确，有创意，且排版协调统一，符合规范要求，符合用户的浏览习惯。其次色彩搭配要和谐，以用户体验为主。网页的文字字体合适，图片清晰美观，不影响用户体验。网页内容精简，没有过多无用的内容。

②操作简单

符合系统使用需求，打开网页，网页一览无余。网页内容言简意赅，容易操作，快速上手。

③功能齐全

实现系统管理员、教务管理者、老师、学生这四种角色使用教务系统平台，不同角色对应的功能都一一实现。

4.3项目评估

国内教育市场发展迅速，尤其是在线教育方面，目前已有20多万的用户规模，然而我国在线教育还处于发展初期阶段，市场中存在许多问题，如：用户缺乏系统的学习方法指导、教学资源不够丰富、教学内容不够权威等等。尤其是在信息技术快速发展的今天，如何通过信息化手段促进教育公平发展的问题越来越受到关注。在这样的背景下，教育平台项目需求量也倍增。

4.3.1项目价值

（一）提升学校管理水平，实现学校现代化管理

学校是教育教学和教育科研的基地，是为社会培养人才的摇篮，而教育教学和科研是一所学校的生命。如何利用信息化手段提高教育教学管理水平，成为近年来教育界讨论的焦点。从目前来看，高校信息化建设发展已取得了显著成效，但是教育信息化建设仍存在很多问题，如数据孤岛、重建设轻应用等。同时，高校信息化建设还存在诸多不足。以教育科研为例，当前高校科研经费主要来源于国家财政拨款、社会捐赠和科研课题的研究等，但其中相当部分经费不能直接用于科研活动，科研工作没有得到应有的重视。究其原因，主要是科研管理不规范、信息孤岛现象严重、缺乏有效的信息共享机制等。因此，要提高高校管理水平，就必须进行教学管理平台的建设。

（二）规范教学秩序，提高教育教学质量

在当今信息时代和知识经济时代下，高等教育要实现跨越式发展就必须要走信息化道路。通过建设一个功能强大的数字化平台，可以使高校对教学过程进行全方位、全过程、全天候的管理和监控；可以为学生提供一个自主学习的空间和环境；可以充分调动教师的积极性和创造性；可以及时掌握学生信息反馈；可以与社会各种资源进行充分整合和共享；可以实现对教育教学活动进行科学有效的管理与监控；可以大大提高高校管理水平和效率。从根本上说，数字化平台对于构建适应知识经济时代需求的现代教育模式具有十分重要的意义。

4.3.2项目创新点

4.3.2.1尝试选择云数据库存储信息，从硬件（计算机、网络交换器等设备）、软件、搭建、部署、运维等多个方面节省人力物力时间成本。

4.3.2.2如果数据库出现故障，那么整个系统就会崩溃。所以在传统的架构中，数据备份是一个难题，经常会因为数据丢失导致系统崩溃。尝试使用分布式架构，即可以让多个计算机系统设备共同组成一个数据库，提供完整的数据库服务，例如Oracle、MongoDB、TDSQL等，增加计算机系统的数量，就能提高数据库性能，提升数据库自身的读写性能。

## 主要仪器设备及工具或开发设计软件及工具名称

笔记本或台式机

Eclipse+Spring开发套件 、IntelliJ IDEA、Visual Studio Code等一种开发工具

## 实验步骤

小组成员：

学号： 32001076 姓名： 袁银君 班级： 计算机2001

学号： 32001027 姓名： 常睿嘉 班级： 计算机2001

学号： 32001017 姓名： 何佳静 班级： 计算机2001

学号： 32001298 姓名： 杨晨露 班级： 计算机2001

小组分工：

|  |  |  |
| --- | --- | --- |
| 成员 | 负责内容 | 贡献度 |
| 袁银君 | 系统管理员端和教务人员端的ui设计及前端实现；系统管理员端和教务人员端的后端实现。  文档部分：项目操作手册、项目详细方案以及PPT | 0.25 |
| 常睿嘉 | 登录页面ui设计及实现；教师端和学生端的后端实现。  文档部分：项目详细方案以及PPT | 0.25 |
| 何佳静 | 教师端和学生端的ui设计及前端实现；教师端和学生端的后端实现；主体程序的修改和维护。  文档部分：项目视频、项目详细方案以及所有文件整合 | 0.25 |
| 杨晨露 | 数据库处理；系统管理员端和教务人员端的后端实现。  文档部分：项目详细方案以及项目概要介绍 | 0.25 |

个人实验步骤：

1. 数据库设计以及实现

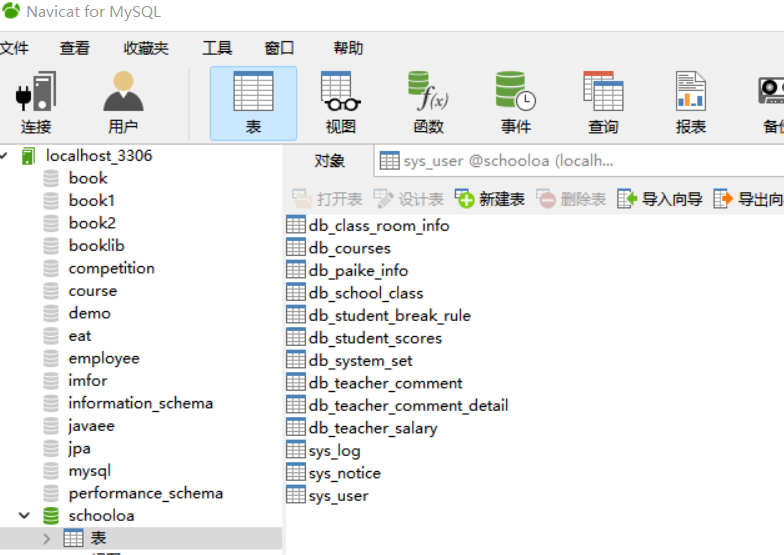
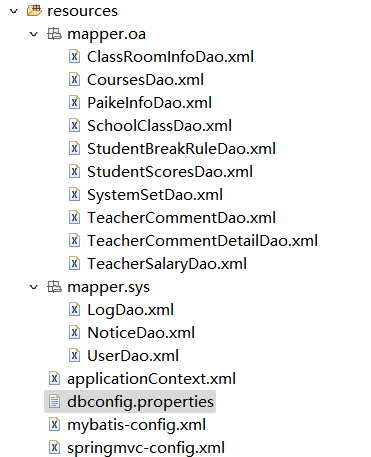


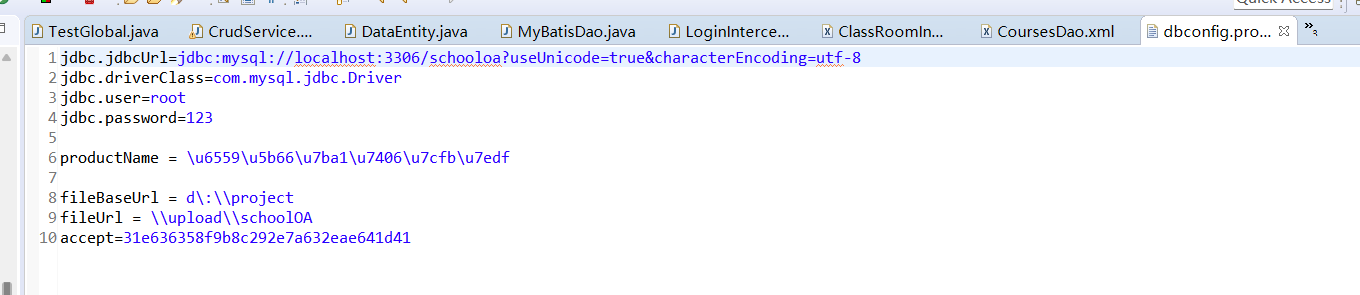
图-数据库

数据库总共有13个表。其中sys开头三个表完成基础信息和功能使用，剩下十个表用来系统的具体实现功能实现。

数据库连接：

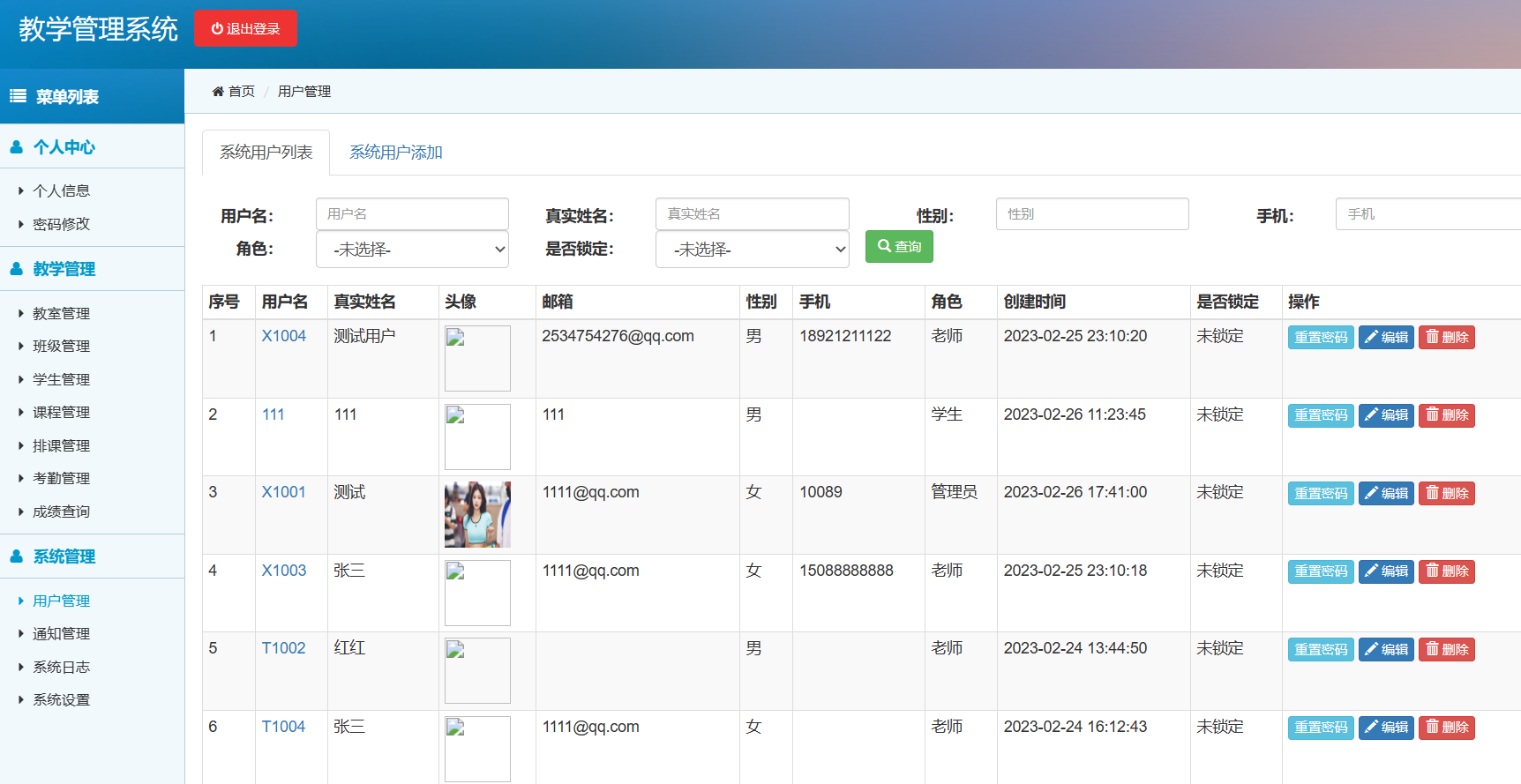


以上是对应表的数据库MySQL操作。

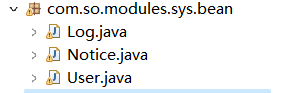


1. 系统管理员端

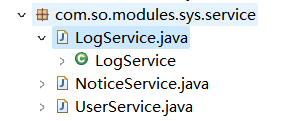
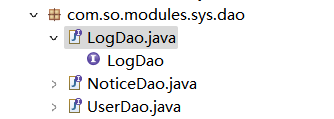




2.1个人中心和系统管理：

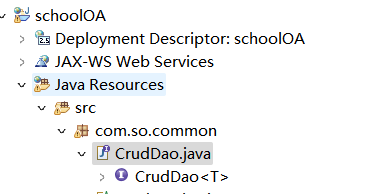


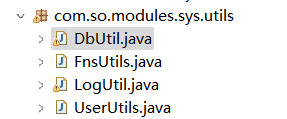
对系统的基础信息进行属性填写，分别是log.java系统日志、Notice.java系统通知、User.java用户信息。



设置相关的Dao接口和service层，用于controller调用。

Dao继承于：



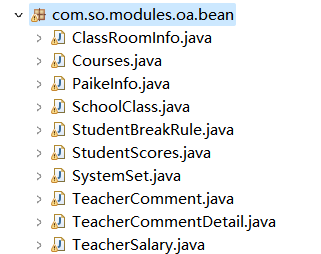


数据库相关连接。

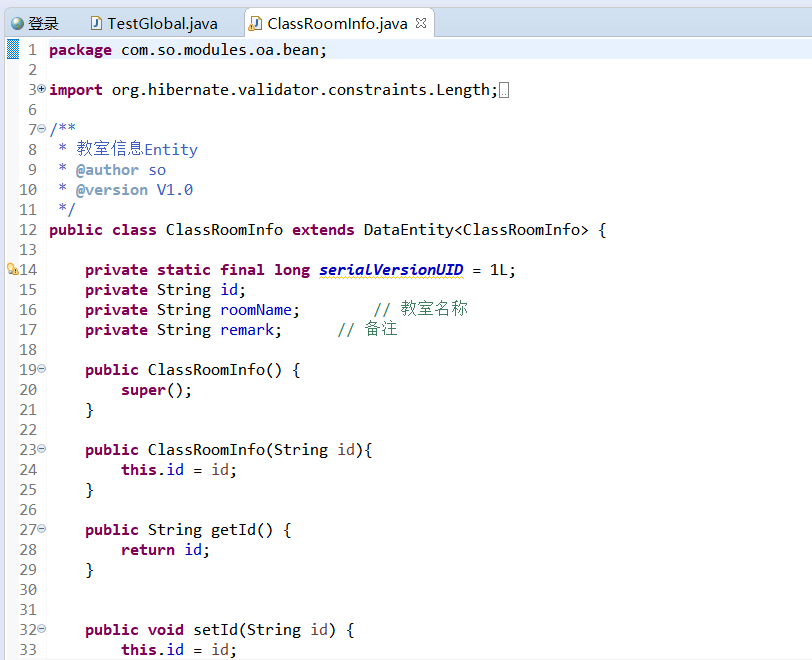


图-教学管理页面

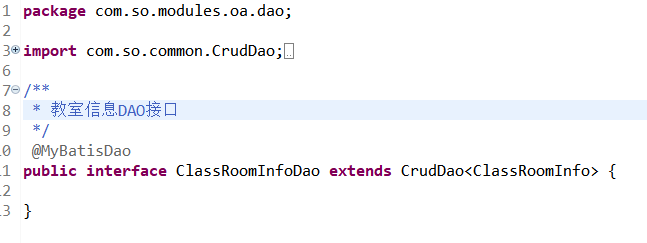
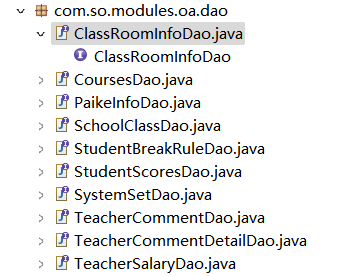
2.2教学管理



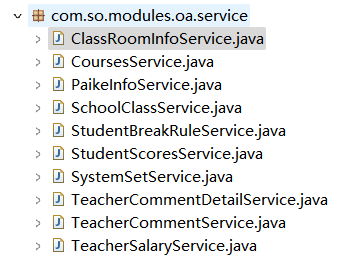
数据库剩下10个表的基础属性输入填写，例如教室信息：



相关Dao接口：

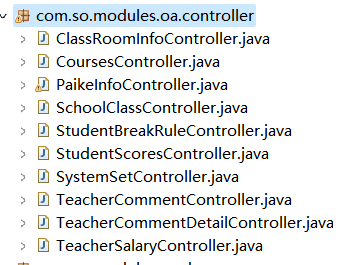


Service层：



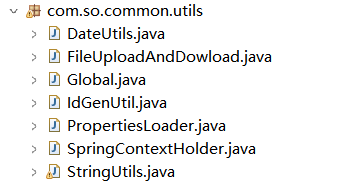
具体实现功能在相关service包里的对应Java实现。

Controller层：



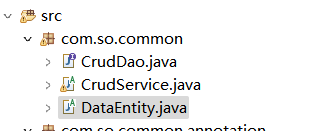
Controller实现调用。

DButil:

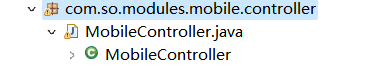


进行相关数据库连接。

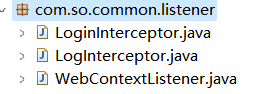
2.3相关包完善：



DataEntity.java存放日期实体类。



标识MyBatis的DAO，方便扫描以及排课系统的设置。



listener包主要包含了事件监听器接口和一些常用的事件监听器实现类。事件监听器用于监控应用程序所发生的事件，并在事件发生时执行一些响应操作。

测试类。

## 实验数据记录和处理（或s设计效果）

1.数据库设计以及实现

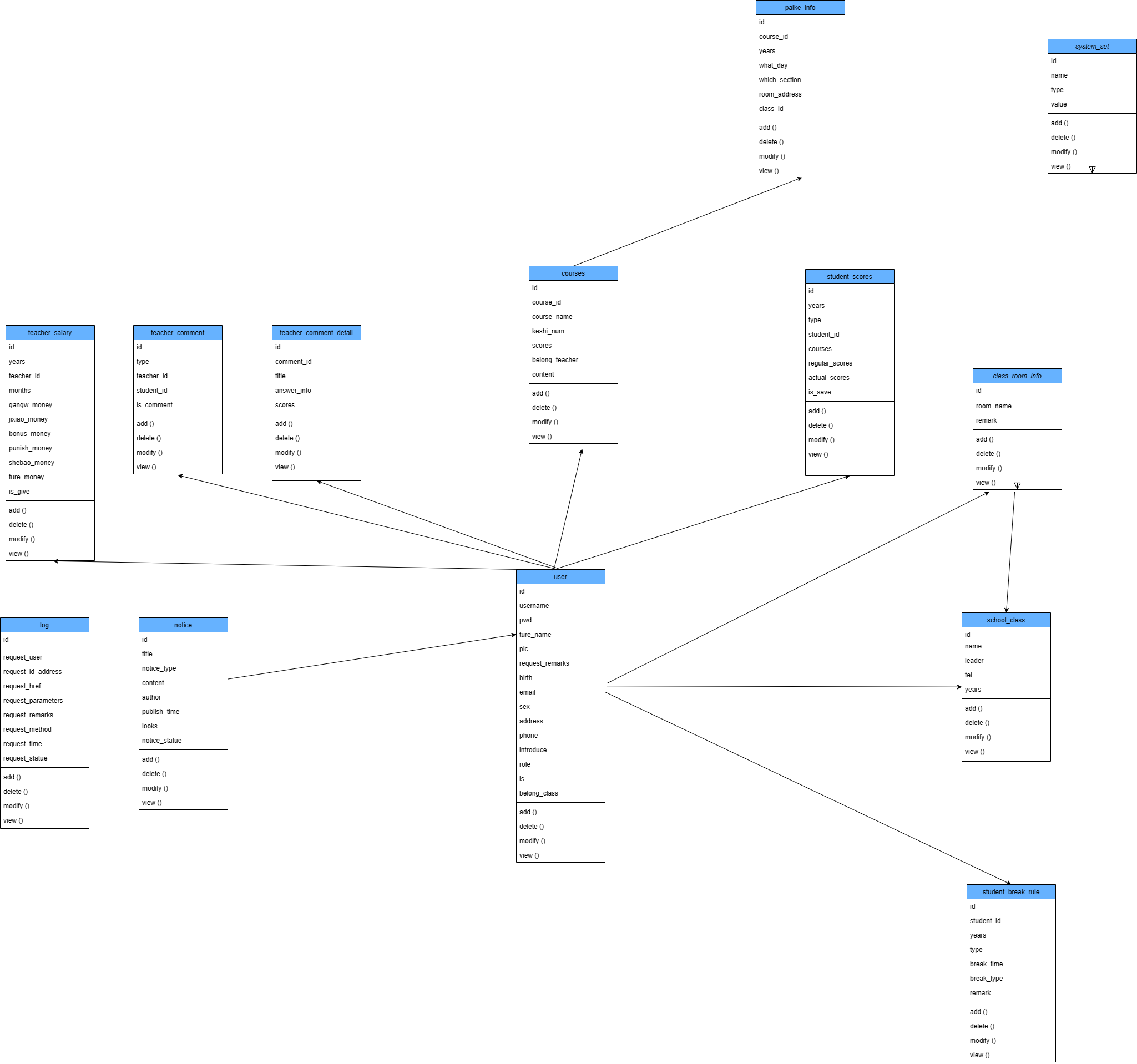


图-数据库类图设计

根据类图创建数据库：



图-sql文件

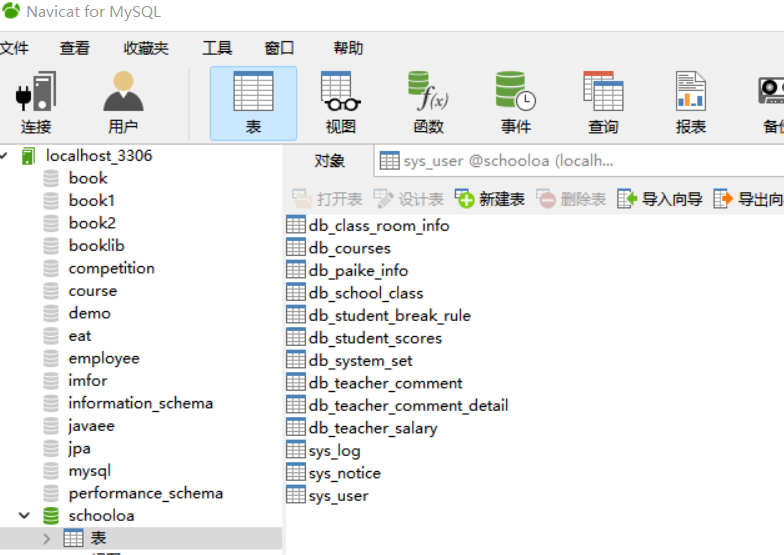
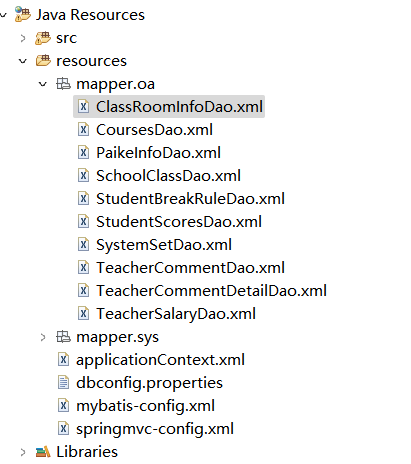
生成的数据库：

图-数据库

MySQL连接以及具体操作：



2.系统管理员端（主要负责除了排课管理以外的所有页面和功能）



因教室管理、班级管理、学生管理、课程管理、考勤管理和成绩查询页面的相关功能都类似，这里主要用教室管理来举例说明。

2.1教室管理：

教室信息实体类：

package com.so.modules.oa.bean;

import org.hibernate.validator.constraints.Length;

import com.so.common.DataEntity;

/\*\*

\* 教室信息Entity

\*/

public class ClassRoomInfo extends DataEntity<ClassRoomInfo> {

private static final long serialVersionUID = 1L;

private String id;

private String roomName; // 教室名称

private String remark; // 备注

public ClassRoomInfo() {

super();

}

public ClassRoomInfo(String id){

this.id = id;

}

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

@Length(min=1, max=40, message="教室名称长度必须介于 1 和 40 之间")

public String getRoomName() {

return roomName;

}

public void setRoomName(String roomName) {

this.roomName = roomName;

}

public String getRemark() {

return remark;

}

public void setRemark(String remark) {

this.remark = remark;

}

}

教室信息service层：

信息查询：

**public** PageInfo<ClassRoomInfo> findPage(ClassRoomInfo classRoomInfo) {

PageHelper.*startPage*(classRoomInfo.getPageNo(), classRoomInfo.getPageSize());

List<ClassRoomInfo> list = **super**.findList(classRoomInfo);

PageInfo<ClassRoomInfo> pageInfo = **new** PageInfo<ClassRoomInfo>(list, classRoomInfo.getPageSize());

**return** pageInfo;

}

信息修改和保存：

@Transactional(readOnly = **false**)

**public** **void** save(ClassRoomInfo classRoomInfo) {

**if** (StringUtils.*isEmpty*(classRoomInfo.getId())) {

classRoomInfo.setId(IdGenUtil.*getUUID*());

dao.insert(classRoomInfo);

}**else**{

dao.update(classRoomInfo);

}

}

信息删除：

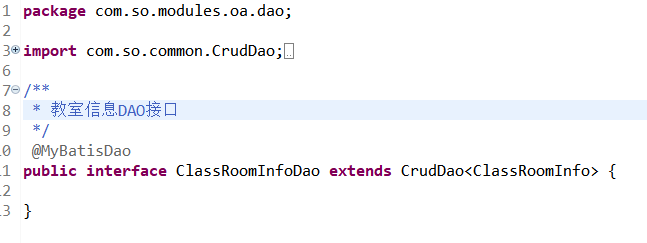
@Transactional(readOnly = **false**)

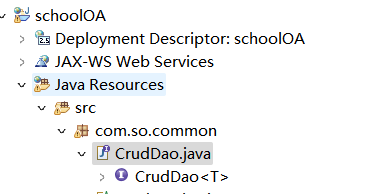
**public** **void** delete(ClassRoomInfo classRoomInfo) {

**super**.delete(classRoomInfo);

}

教室管理Dao类：





GrudDao.java具体源代码：

package com.so.common;

import java.util.List;

/\*\*

\* DAO支持类实现

\*/

public interface CrudDao<T> {

/\*\*

\* 获取单条数据

\* @param id

\* @return

\*/

public T get(String id);

/\*\*

\* 获取单条数据

\* @param entity

\* @return

\*/

public T get(T entity);

/\*\*

\* 查询数据列表，如果需要分页，请设置分页对象，如：entity.setPage(new Page<T>());

\* @param entity

\* @return

\*/

public List<T> findList(T entity);

/\*\*

\* 查询所有数据列表

\* @param entity

\* @return

\*/

public List<T> findAllList(T entity);

/\*\*

\* 查询所有数据列表

\* @see public List<T> findAllList(T entity)

\* @return

\*/

@Deprecated

public List<T> findAllList();

/\*\*

\* 插入数据

\* @param entity

\* @return

\*/

public int insert(T entity);

/\*\*

\* 更新数据

\* @param entity

\* @return

\*/

public int update(T entity);

/\*\*

\* 删除数据（一般为逻辑删除，更新del\_flag字段为1）

\* @param id

\* @see public int delete(T entity)

\* @return

\*/

@Deprecated

public int delete(String id);

/\*\*

\* 删除数据（一般为逻辑删除，更新del\_flag字段为1）

\* @param entity

\* @return

\*/

public int delete(T entity);

}

教室管理controller类：

信息展示：

@RequestMapping(value = {"list", ""})

**public** String list(ClassRoomInfo classRoomInfo, HttpServletRequest request, HttpServletResponse response, Model model) {

PageInfo<ClassRoomInfo> page = classRoomInfoService.findPage(classRoomInfo);

model.addAttribute("pageInfo", page);

**return** "modules/oa/classRoomInfoList";

}

@RequestMapping(value = "form")

**public** String form(ClassRoomInfo classRoomInfo, Model model) {

model.addAttribute("classRoomInfo", classRoomInfo);

**return** "modules/oa/classRoomInfoForm";

}

信息保存：

@RequestMapping(value = "save")

**public** String save(ClassRoomInfo classRoomInfo, Model model, RedirectAttributes redirectAttributes) {

classRoomInfoService.save(classRoomInfo);

redirectAttributes.addFlashAttribute("msg", "保存记录成功！");

**return** "redirect:"+"/oa/classRoomInfo";

}

信息删除：

@RequestMapping(value = "delete")

**public** String delete(ClassRoomInfo classRoomInfo, RedirectAttributes redirectAttributes) {

classRoomInfoService.delete(classRoomInfo);

redirectAttributes.addFlashAttribute("msg", "删除记录成功！");

**return** "redirect:"+"/oa/classRoomInfo";

}

教室管理数据库操作：

信息展示：

<sql id=*"classRoomInfoColumns"*>

a.id AS "id",

a.room\_name AS "roomName",

a.remark AS "remark"

</sql>

<sql id=*"classRoomInfoJoins"*>

</sql>

信息查询：

<select id=*"get"* resultType=*"ClassRoomInfo"*>

SELECT

<include refid=*"classRoomInfoColumns"*/>

FROM db\_class\_room\_info a

<include refid=*"classRoomInfoJoins"*/>

WHERE a.id = #{id}

</select>

<select id=*"findList"* resultType=*"ClassRoomInfo"*>

SELECT

<include refid=*"classRoomInfoColumns"*/>

FROM db\_class\_room\_info a

<include refid=*"classRoomInfoJoins"*/>

<where>

<if test=*"roomName != null and roomName != ''"*>

AND a.room\_name = #{roomName}

</if>

</where>

<choose>

<otherwise>

</otherwise>

</choose>

</select>

<select id=*"findAllList"* resultType=*"ClassRoomInfo"*>

SELECT

<include refid=*"classRoomInfoColumns"*/>

FROM db\_class\_room\_info a

<include refid=*"classRoomInfoJoins"*/>

<where>

</where>

<choose>

<otherwise>

</otherwise>

</choose>

</select>

信息添加：

<insert id=*"insert"*>

INSERT INTO db\_class\_room\_info(

id,

room\_name,

remark

) VALUES (

#{id},

#{roomName},

#{remark}

)

</insert>

信息更新：

<update id=*"update"*>

UPDATE db\_class\_room\_info SET

room\_name = #{roomName},

remark = #{remark}

WHERE id = #{id}

</update>

信息删除：

<update id=*"delete"*>

DELETE FROM db\_class\_room\_info

WHERE id = #{id}

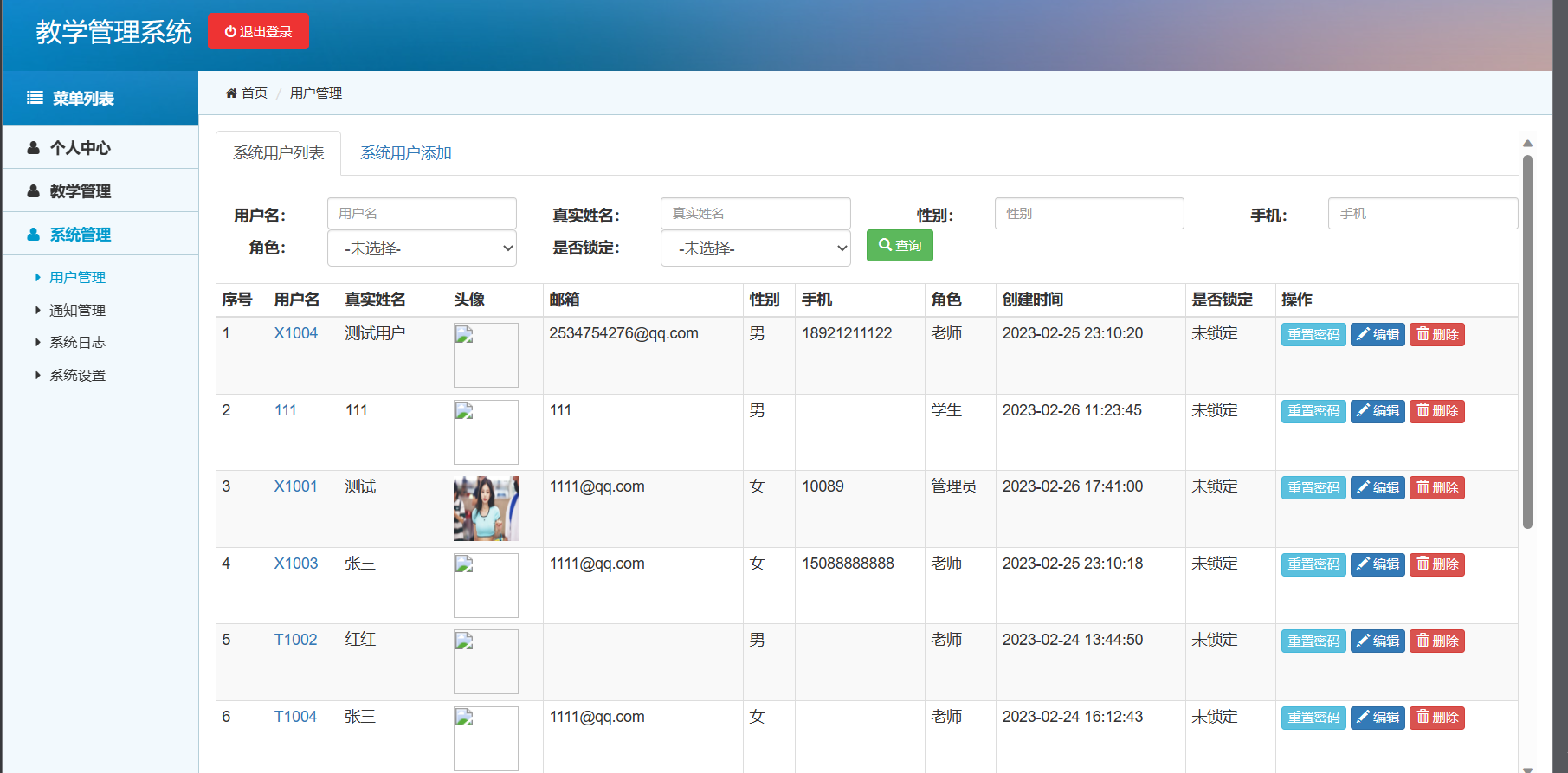
</update>

2.2系统管理（除了系统日志以外的其他页面）



用户管理、通知管理和系统设置都与上一级的教学管理操作类似。

用户管理：



只是用户管理这里加了一个锁定的选择。锁定之后，该账号不能进行任何操作。

1. 教务系统端



此系统功能相对于系统管理员端的功能有相对减少，基本功能与系统管理员端一致。

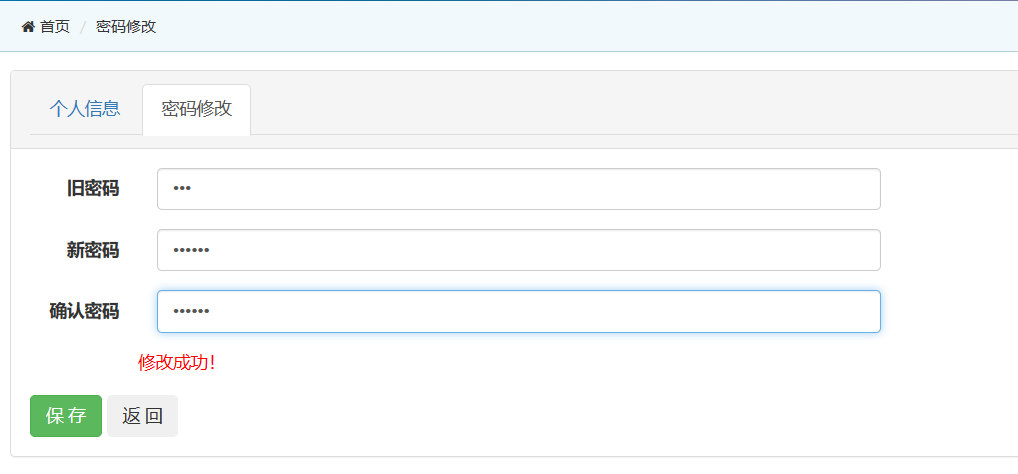
## 实验结果与分析（或设计成效分析）

教务系统端的功能基本实现：

1. 个人信息与密码修改



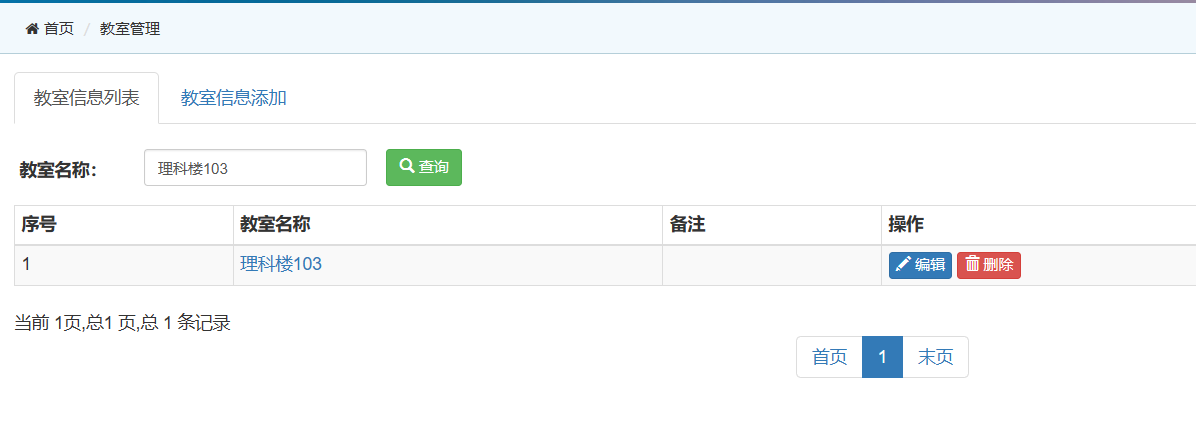




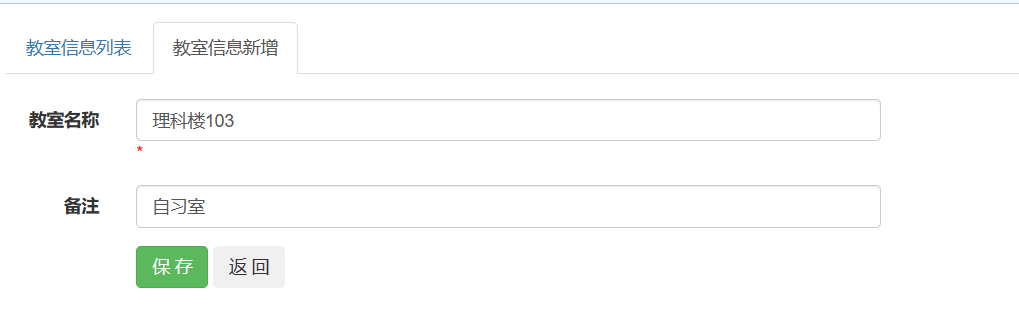
1. 教室管理



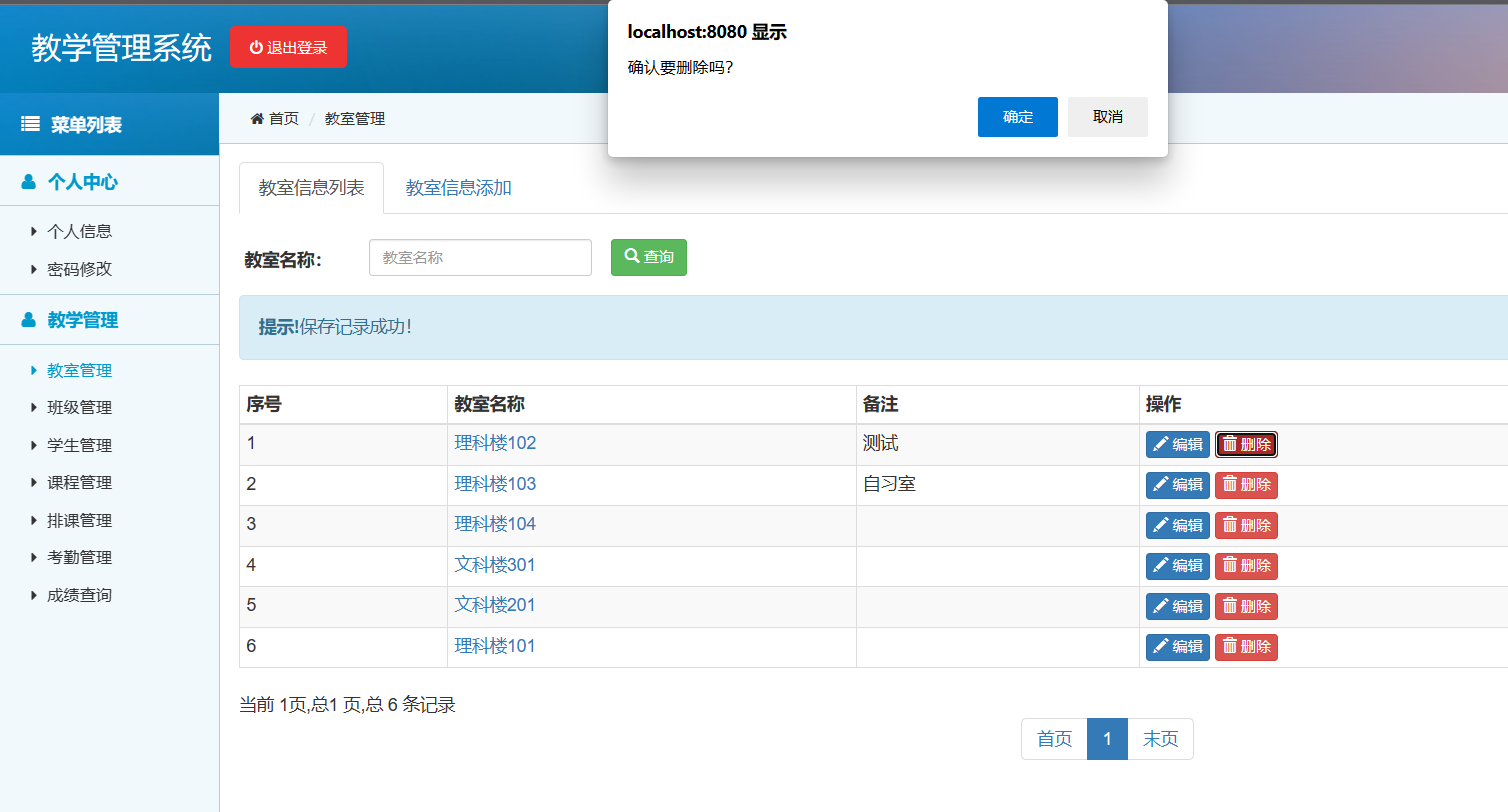
查询：



修改：



删除：



添加：



其他管理页面功能与之相似。

## 讨论、心得

运用代码实现对教学的信息化管理，方便学校各个对象对教学进行管理。这是一种有效率有层次的教学方法，给我们的生活带来了很大的便利。以此不难看出科技是为人类服务，以人为本。

|  |
| --- |
| 指导教师评语  实验报告评分（百分制）： 分  指导教师签名：  日 期： 年 月 日 |