



**本科毕业论文**

**基于ssm实现学生选课系统**

**黄少杰**

**201430340506**

指导老师**张义青讲师**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 学院名称 |  | **数学与信息学院** | 专业名称 |  | **软件工程** |
| 论文提交日期 |  | 2018年4月25日 | 论文答辩日期 |  | 2018年5月7日 |

摘 要

信息化时代发展，各种信息化涌现，企业信息化建设、信息化管理、信息化教育等等。在各种信息化的涌现下，要跟得上时代，教育管理也应该进行信息化。教育管理信息化让管理学生事务变得更加便捷，减轻了教务管理员的压力，让老师能更专注于教育事业。在信息化的推动下，教育管理系统也变得更加标准和易使用，越来越多的高校也用上了一些教育管理系统，如最出名的是正方系统。在所有教育管理系统的事务中，选课事务是尤其复杂的，除了要满足大多的制约条件（如：上课地点、上课时间、选课时间等）之外，还要解决老师排课矛盾，学生退课流程。本文由此选择教务事务中的选课事务进行研究，本系统编程使用编程语言为Java，使用SSM(Spring,SpringMVC,Myabtis)以及一些主流的web技术来开发系统，用restful风格设计和开发一个学生选课系统。

本系统共分为三个模块、学生选课模块、老师排课模块，学生模块下有：选课、查看老师、查看课表、退课、个人信息管理功能。老师模块有：个人信息管理、查看课程、排课、查看个人课表、查看已选课老师、导出学生名单。管理员模块：学生信息管理、老师信息管理、增加课程、查看排课、发起公告、系统管理。

系统界面简洁，功能齐全，适合大多数量高校使用。用信息化手段使得学生选课简单、规范、标准。

关键词：学生选课系统 java SSM restful

**Student selective courses system base on SSM Technologies**

Huang Shaojie

(College of Software Engineering, South China Agricultural University, Guangzhou 510642, China)

**Abstract:** The development of information era, the emergence of all kinds of information, enterprise information construction, information management, information education and so on. Under all kinds of information springing up, in order to keep up with the times, education and management should also be informationalized. Informatization of education management makes management of student affairs more convenient, alleviates the pressure of educational administrators, and enables teachers to focus more on education. Under the impetus of information technology, the education management system has become more standard and easy to use. More and more colleges and universities have also used some educational management systems, such as the most famous is the square system. In all the affairs of the educational management system, the course selection is particularly complex. In addition to meeting most of the constraints (such as: class place, class time, course selection time, etc.), it is also necessary to solve the contradiction between the teachers and the course. This paper studies the course selection in educational affairs. This system uses programming language as Java, uses the combination framework of Spring, SpringMVC, Mybatis and some mainstream web technologies to develop the system, uses Git to manage the code, and designs and develops a student selection system with restful style.

This system is divided into three modules, the students' course selection module and the teacher's class module. Under the student module, we have the functions of selecting courses, checking the teachers, checking the timetable, retreating the class, and the personal information management. The teacher module includes personal information management, viewing courses, arranging courses, checking personal timetable, checking the list of selected teachers and exporting students. Administrators module: student information management, teacher information management, adding courses, checking timetable, launching announcements, and system management.

The system interface is simple and functional, which is suitable for most universities. With information technology, student selection is simple, standard and standard.

**Keywords:** Student selective courses system java SSM restful

**目 录**

1 前言 1

1.1 课题背景 1

1.2 课题研究内容与思路 1

1.3 课题研究方法 1

2 涉及技术介绍 3

2.1 Java 3

2.2 Spring框架 3

2.3 Mybatis持久化框架 3

2.4 Tomcat容器 4

2.5 MySQL数据库 4

2.6 Nginx 4

3 系统需求分析 5

3.1 登陆 5

3.2 修改密码 6

3.3 学生信息管理 8

3.4 教师信息管理 9

3.5 管理员课程管理 10

3.6 公告系统 11

3.7 教师课程管理 12

3.8 学生选课 12

4 系统设计 14

4.1 系统体系结构设计 14

4.2 系统特性 15

4.3 功能设计 15

4.3.1 登陆 15

4.3.2 修改密码 16

4.3.3 学生信息管理 19

4.3.4 教师信息管理 21

4.3.5 管理员课程管理 21

4.3.6 公告系统 23

4.3.7 查看个人信息 24

4.3.8 教师课程管理 26

4.3.8 学生选课 26

5.4 数据库设计 27

5.4.1 E-R图 27

5.4.2 物理数据库 28

5.4.3 表结构 28

5.4.4 存储过程 33

6.5 界面设计 34

6.5.1 登陆 34

6.5.2 修改密码 35

6.5.3 学生信息管理 37

6.5.4 教师信息管理 38

6.5.5 课程信息管理 39

6.5.6 公告系统 41

6.5.7 教师课程管理 43

6.5.8 学生选课 44

7 总结与展望 47

7.1 总结 47

7.2 展望 47

参考文献 48

致谢 49

# 1 前言

## 1.1 课题背景

## 21世纪人类已经迈入了信息化时代，人类经历过从远古时期的绳结记录信息、到后来的纸笔记录信息，到了今天使用计算机记录信息。信息化技术和计算机网络已经进入千家万户。信息化的水平已经逐渐成为一个国家和地区的综合实力的衡量标准。信息化技术已经渗入社会的方方面。如电子政务、电子商务、电子金融、科学计算、教育、通信技术、能源、交通控制等，在这个信号化时代的今天，一些重复、繁琐、计算量大的事务。逐渐地慢慢交由计算机代替处理。如erp系统、人事管理系统、审计系统、政务处理系统、电子上传等。随着信息化发展，各行各业信息化的需求也越来越大。像传统的教务管理，需要教务人员重复有大量的工作，需要教务职工重复地记录信息，又需要学生班干配合统计班级同学的信息。就如选课，要老师先联系教务人员开始开课，然后在联系纪委，纪委再通知个班班长或学习委员让学生准备好选课，其中一层又一层，能否有一个直接能连接学生和老师的平台代替那些重复繁琐的教务事务呢？学生选课系统就是一个用来连接授课老师和上课学生的系统。如今高校方希望有这么一个信息化的选课系统帮助老师安排课程和学生选课。

## 可知教务系统信息也十分必要，目前在市面也有不少教务系统，但大多数都使用比较传统的Java web技术开发如EJB，系统庞大、老旧而且复杂，调试十分困难。不适合技术快速迭代的今天。本文以选课这个教务事务来研究，用SSM框架开发一个方便调试，轻量级的教务系统。

## 1.2 课题研究内容与思路

**研究内容：**

本系统以高校的选课教务内容来开展，针对学生选课和老师排课进行设计和分析，重点是对课程安排和选课上，研究的内容主要可划分为两个方面：

（1）针对老师对选课系统的需求，保证老师能顺利安排课程，配合好老师教学工作

（2）满足学生的需求，尽量地简化选课步骤，省去学生的学习成本和使用成本，尽量保证同一时间段学生大量选课的请求。

**研究思路：**

通过走入老师教务工作和学生工作，熟悉理解学校教务需要，帮助高校构建选课系统的需求，找出目前高校选课上的难点，了解一些教务处理的流程并由此寻找设计思路，熟悉和理解处理流程后提出设计方案。完成第一期调研后，看看市面上目前存在相关教务系统，研究目前现有教务系统的优缺点，并想出改善方案整合优化到自己系统中，选择其好的地方来借鉴学习，经过深入了解和分析后确定出了系统一定逻辑功能。向学生透露设计方案和思路，让同学们指出优缺点，吸纳同学们的意见。以学生和老师的角度设计和研究出更好的选课系统。

## 1.3 课题研究方法

本系统用restful风格设计和开发，采用前后端分离技术。不再像以往传统的软件系统一样将前端页面依附与后端中。这样一来前后端代码改动不会互相影响，前后端交互使用json格式序列化的数据，通过http方式来传输（即后端提供接口交由前端调用）。接口描述和设计符合restful风格，对资源操作描述清晰。

用Nginx服务器来作为反向代理服务器，将前后端资源彻底分离。Nginx来分发前后端资源请求。前端静态资源请求即返回静态资源数据，若是后端动态资源请求到tomcat中，由tomcat中运行的Java程序来处理动态资源请求。

使用SSM框架技术开发Java后端，使用Spring轻量级的IOC框架。用SpringMVC开发controller层为前端提供接口。使用Mybatis轻量级持久层技术作为数据库访问层。以后即使更换数据库只需要更换jdbc的数据库驱动。

**2 涉及技术介绍**

## 2.1 Java

Java是sun公司在1995年推出的高级程序设计语言。可以运行与不同的平台之间，具有跨平台性。

Java是基于C++拓展的程序语言，比C++更加贴合面向对象，摒弃C++多继承和指针的混乱。是一门完全面向对象的编程语言。在Java之中一切皆为对象，具有简单性、可移植性、跨平台性。可以编写web程序、桌面应用程序，嵌入式应用程序和分布式系统。

## 2.2 Spring框架

Spring是一个轻量级的IOC和AOP的容器框架。为了解决企业级应用开发的复杂性而生。使用java bean 代替传统复杂的EJB。

Spring的目标是为J2EE应用提供全方位的整合框架，在Spring框架下实现了多个子框架组合，这些子框架彼此独立

## 2.3 Mybatis框架

 Mybatis是一款优秀的持久层框架，它支持定制化 SQL、存储过程以及高级映射。

Mybatis最大的优点就是简单上手，相较灵活。在性能方面因为不同于传统重量级的持久化框架要保持Pojo（Plain Old Java Object）的完整性要额外加载一些关联数据，所以性能和灵活性更加好。避免了几乎所有的 JDBC 代码和手动设置参数以及获取结果集。MyBatis 可以使用简单的 XML 或注解来配置和映射原生信息，将接口和 Java 的 POJO映射成数据库中的记录

## 2.4 Tomcat服务器

Tomcat是一款Apche开源项目下的一个核心项目，后来得到了sun公司的参与和支持。现在能支持最新的servlet和jsp规范，是servlet的运行容器。Tomcat是一个轻量级的javaweb服务器，也一个良好轻量servlet运行容器。

2.5 MySQL数据库

[MySQL](https://baike.baidu.com/item/MySQL/471251" \t "https://baike.baidu.com/item/MySQL%E6%95%B0%E6%8D%AE%E5%BA%93/_blank)是一种[开放源代码](https://baike.baidu.com/item/%E5%BC%80%E6%94%BE%E6%BA%90%E4%BB%A3%E7%A0%81" \t "https://baike.baidu.com/item/MySQL%E6%95%B0%E6%8D%AE%E5%BA%93/_blank)的关系型[数据库管理](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86)系统（RDBMS），使用最常用的数据库管理语言--[结构化查询语言](https://baike.baidu.com/item/%E7%BB%93%E6%9E%84%E5%8C%96%E6%9F%A5%E8%AF%A2%E8%AF%AD%E8%A8%80)（SQL）进行数据库管理。

MySQL是开放源代码的，因此任何人都可以在General Public License的许可下下载并根据个性化的[需要](https://baike.baidu.com/item/%E9%9C%80%E8%A6%81/9109492" \t "https://baike.baidu.com/item/MySQL%E6%95%B0%E6%8D%AE%E5%BA%93/_blank)对其进行修改。

MySQL因为其速度、可靠性和适应性而备受关注。大多数人都认为在不需要[事务](https://baike.baidu.com/item/%E4%BA%8B%E5%8A%A1" \t "https://baike.baidu.com/item/MySQL%E6%95%B0%E6%8D%AE%E5%BA%93/_blank)化处理的情况下，MySQL是管理内容最好的选择。

## 2.6 Nginx服务器

Nginx是一个http服务器，长期以来被用于http服务和反向代理。它可以用作于http服务器、反向代理服务器、邮件服务器、tcp/udp连接的代理服务器。由俄罗斯的工程师Igor Sysoev编写的，长期以来许多俄罗斯的网站如：Yandex, Mail.Ru, VK, 还有 Rambler服务器都使用nginx作为http服务器。

# 3 系统需求分析

3.1交互设计需求

界面交互是用户接触系统的第一步，对于用户而言，界面就是软件系统最直接的体现。界面交互最好是简介和直观，不需要用户花费过多时间学习使用，同时界面上功能点又必需齐全。但是过多的功能追求又会影响到界面的简洁性和美观。在产品设计中，一直在设计美观、功能、技术性上寻找平衡点。在这个基础上，紧跟时代潮流，贴合时代的主题风格，设计出精致的界面

以用户体现为最优。因此在简介设计风格的基础上，选择美观的基础元素模块，以绿色打底，贴合大自然元素，蓝色为主题元素象征着自然脉动。内嵌标签使得用户切换功能使用更快捷方便，左边菜单栏列出用户可以使用的所有一级目录，点击一级菜单展示详细功能。整个布局清晰明了，用户可以根据这二级的菜单栏熟悉使用，使用该系统能够快速上手。

3.2功能需求

通过参与学生工作，和走访学生老师而进行调查，熟悉了老师、教务人员、学生的需求，根据这三方面的需求整合出功能需求。

根据这三类型的人讲系统的功能分为三个模块，管理员模块代表这教务人员可以管理学生信息、教师信息以及课程信息是系统的管理者、教师模块则是代表着参与教学工作的老师，主要功能点则是课程安排，查看上课学生。最后就是学生模块了，学生模块主要代表这选课的学生，在这个模块中有这最核心的功能—选课。为这三大类型使用者划分好了不同的首页，有着不同的功能。管理员首页是关于管理的功能，如学生管理、课程管理、课程管理、以及系统公告。教师首页，个人信息管理、课程安排、查看公告、查看课表、导出上课学生名单。学生首页则有个人信息管理、选课、退选课程、个人课表、以及查看老师。不同的首页之间不能互相访问，不能跨越首页访问。保证了系统的安全性。以下为三个模块的用例图



图1 学生模块用例图



图2 教师模块用例图



图3 管理员模块用例图

3.3 登录

对于登录功能，老师、学生以及管理员都有登录操作。系统可以根据用户选择不同的登录来判断用户不同身份。系统根据不同的用户身份来进行用户名和密码的匹配，若匹配成功则进入相对应的用户页面，反之则出现提示信息。



图3登录用例图

3.4 修改密码

对于修改功能，老师、学生以及管理员都有修改密码操作。系统可以根据用户选择不同的登录状态来判断用户不同身份。系统根据不同的用户身份来区别不同用户的密码修改，修改密码先输入原密码，输入新密码和确认密码。



图4修改密码用例图

3.5学生信息管理

学生信息管理主要使用对象为管理员，管理员可以查看学生信息、增加学生信息、修改学生信息、删除学生信息。学生基本信息包括学生编号、学生姓名、学生性别、学生年龄、性别、院系、专业，以及入学日期。



图5学生信息管理管理用例图

（1）查询学生信息（通过学生id或名称或入学日期、院系查询学生信息）：管理员登录之后进入了管理员首页，点击左侧菜单栏的“学生管理”菜单中的“查询学生信息”，即列出所有的学生信息（前5个）。在学生查询页面的上方有查询条件可以选择，可以选择精确搜索输入学生学号查询学生，或者输入学生名字、院系、入学年限进行模糊匹配查询。

（2）修改学生信息：管理员登录之后进入了管理员首页，点击左侧菜单栏的“学生管理”菜单中的“修改学生信息”，根即列出所有的学生信息（前5个）。查看先显示出所有的学生信息，根据系统输入的条件进行查询出要删除的学生信息。选择某个要修改的学生，点击修改按钮，会弹出学生信息简略图，修改完毕，点击保存会提示学生信息修改成功，会提示是否提交这次修改，确定则提交这次修改到系统，反之则撤回本次操作。

（3）添加学生信息（将学生信息导入到系统中）：管理员登录之后进入了管理员首页，点击左侧菜单栏的“学生管理”菜单中的“添加学生信息”，会弹出增加学生页面，填写学生信息，如学号（密码会自动生成默认跟学号一样），姓名，年龄，入学日期、院系。填写完毕后点击保存，会提示信息保存成功。

（4）删除学生信息：管理员登录之后进入了管理员首页，点击左侧菜单栏的“学生管理”菜单中的“删除学生信息”，查看先显示出所有的学生信息，根据系统输入的条件进行查询出要删除的学生信息。选择要删除的学生信息然后点击删除按钮，点击后会弹出提示框询问是否删除，若点确定则会提示删除成功，反之则撤回当前操作。

3.6 教师信息管理

教师信息管理模块为管理员提供管理教师信息的功能，管理员可以查看教师信息、增加教师信息、修改教师信息、删除教师信息。教师基本信息包括教师工号、教师姓名、教师院系、入职日期、教师简介。

（1）查询教师信息（通过教师工号、教师姓名或入学日期、院系查询教师信息信息）：管理员登录之后进入了管理员首页，点击左侧菜单栏的“教师管理”菜单中的“查询教师信息”，即列出所有的教师信息（前5个）。在教师查询页面的上方有查询条件可以选择，可以选择精确搜索输入教师工号查询教师，或者输入教师姓名、院系、入职日期进行模糊匹配查询。

（2）修改教师信息：管理员登录之后进入了管理员首页，点击左侧菜单栏的“课程管理”菜单中的“修改学生信息”，根即列出所有的教师信息（前5个）。查看先显示出所有的教师信息，根据系统输入的条件进行查询出要修改的教师信息。选择某个要修改的教师，点击修改按钮，会弹出教师信息简略图，修改完毕，点击保存会提示学生信息修改成功，会提示是否提交这次修改，确定则提交这次修改到系统，反之则撤回本次操作。

（3）添加教师信息（将教师信息导入到系统中）：管理员登录之后进入了管理员首页，点击左侧菜单栏的“教师管理”菜单中的“添加教师信息”，会弹出增加教师页面，填写教师信息，如工号（密码会自动生成默认跟学号一样），姓名，入职日期、院系。填写完毕后点击保存，会提示信息保存成功。

（4）删除教师信息：管理员登录之后进入了管理员首页，点击左侧菜单栏的“教师管理”菜单中的“删除教师信息”，查看先显示出所有的教师信息，根据系统输入的条件进行查询出要删除的教师信息。选择要删除的教师信息然后点击删除按钮，点击后会弹出提示框询问是否删除，若点确定则会提示删除成功，反之则撤回当前操作。

## 3.7管理员课程管理

管理员课程管理主要使用对象是管理员，用于管理员新增开设课程（让老师有课程可以安排），修改课程信息或下架过时课程，其中课程信息包括课程号、课程名称、学分、课程性质、课程简介。

（1）查询课程信息（通过课程号码、课程名称或课程性质程信息）：管理员登录之后进入了管理员首页，点击左侧菜单栏的“课程管理”菜单中的“查询教师信息”，即列出所有的课程信息（前5个）。在教师查询页面的上方有查询条件可以选择，可以选择精确搜索输入教课程号查询课程，或者输入课程名、课程性质进行模糊匹配查询。

（2）修改课程信息：管理员登录之后进入了管理员首页，点击左侧菜单栏的“课程管理”菜单中的“修改课程信息”，根即列出所有的课程信息（前5个）。查看先显示出所有的课程信息，根据系统输入的条件进行查询出要修改的课程信息。选择某个要修改的课程信息，点击修改按钮，会弹出课程信息简略图，修改完毕，点击保存会提示学生信息修改成功，会提示是否提交这次修改，确定则提交这次修改到系统，反之则撤回本次操作。

（3）添加课程信息（将课程信息导入到系统中）：管理员登录之后进入了管理员首页，点击左侧菜单栏的“课程管理”菜单中的“添加课程信息”，会弹出增加教师页面，填写课程信息，如课程号，课程名称，学分、课程性质。填写完毕后点击保存，会提示信息保存成功。

## （4）删除课程信息：管理员登录之后进入了管理员首页，点击左侧菜单栏的“课程管理”菜单中的“删除课程信息”，查看先显示出所有的课程信息，根据系统输入的条件进行查询出要删除的课程信息。选择要删除的课程信息然后点击删除按钮，点击后会弹出提示框询问是否删除，若点确定则会提示删除成功，反之则撤回当前操作。

## 3.8 公告系统

公告系统用于告知所有的系统使用者的信息（如开始选课），一般公告有5到10条，由管理员发布公告，下架公告，编辑公告。



图6管理员公告系统用例图

（1）发布公告，管理员登录后进入到管理员首页，点击左侧菜单栏的公告系统，会展示出二级菜单栏点击公告管理，会进入公告管理页面。点击左上方的发布广告，会进入公告编辑页面，填写公告标题和发布内容，点击保存则会成功发布公告让所有系统使用者看到。

（2）编辑公告，管理员登录后进入到管理员首页，点击左侧菜单栏的公告系统，会展示出二级菜单栏点击公告管理，会进入公告管理页面。可以看到所有的公告列表，点击要编辑的公告，选择编辑则会跳转到公告编辑页面，将已经在线上的公告内容修改后点击保存则可以把这次修改提交到系统。

（3）查看公告，管理员登录后进入到管理员首页，点击左侧菜单栏的公告系统，会展示出二级菜单栏点击公告管理，会进入公告管理页面。可以看到所有的公告列表和浮动的所有公告栏目。

（3）删除公告，管理员登录后进入到管理员首页，点击左侧菜单栏的公告系统，会展示出二级菜单栏点击公告管理，会进入公告管理页面。可以看到所有的公告列表，点击要删除的公告，会弹出提示框询问是否删除，点击是则会删除这条公告以后不再显示，点击否则撤回本次操作。

## 3.9 个人信息管理

教师和学生都可以进行个人信息管理，个人信息管理是关于个人信息的管理，分别在教师和学生页面里面，在教师或学生在加载其首页的时候就会弹出个人信息的页面，学生或教师可以在此时进行个人信息管理。



图7个人信息管理用例图

（1）查看个人信息：学生或教师在成功登录后会跳转学生首页或教师首页，这个时候会今日个人信息页面，会看到关于个人的信息，如教师可以看到自己教师信息（教师工号、教师姓名、入学日期、院系），而学生则可以看到自己的学生信息（学生编号、学生姓名、学生性别、学生年龄、性别、院系、专业，以及入学日期）

（2）修改个人信息：学生或教师在成功登录后会跳转学生首页或教师首页，这个时候会今日个人信息页面，会看到关于个人的信息，这个时候用户（学生或老师）可以进行关于个人信息的修改，修改后点击保存则可以把本次修改提交到系统。

## 3.10 教师课程管理

教师课程管理的主要使用对象为教师。为教师提供查看已在线课程，安排课程，查看教师个人课表和查看个人课程选课学生信息。其中安排课程信息包括上课时间、时刻地点、最大人数、开课时间。



图8教师课程管理用例图

（1）查看课程：教师登录系统之后跳转进入教师首页，点击左侧菜单栏的“课程管理”栏目中的“安排课程”，查看先显示出所有的课程信息（前5个可翻页）。课程信息如上面管理员的课程管理一样，教师可以根据课程号精准查询课程或者课程名或课程性质查询课程。

（2）安排课程：教师登录系统之后跳转进入教师首页，点击左侧菜单栏的“课程管理”栏目中的“安排课程”，查看先显示出所有的课程信息（前5个可翻页）。然后点击要准备开课的课程，页面会弹出对话框，选择开课的时间，开课地点，开课的人数，系选择 “确认”按钮，系统会提示是否开课，再次点击“确定”按钮，会保存本次提交到系统中。

（3）查看个人课表：教师登录系统之后跳转进入教师首页，点击左侧菜单栏的“课程管理”栏目中的“个人课表”，会显示教师的个人课表，课表信息包括（课程号，课程名称，课程性质，学分，上课人数，开课地点，开课时间）。

（4）查看选课学生（查询老师的开课选择该课的学生）：教师登录系统之后跳转进入教师首页，点击左侧菜单栏的“课程管理”栏目中的“个人课表”，会显示教师的个人课表，点击某个课程要看的学生，点击查看学生则会列出所有选择该课的学生。

（5）导出学生名单（老师可以导出学生信息的excel表）：教师登录系统之后跳转进入教师首页，点击左侧菜单栏的“课程管理”栏目中的“个人课表”，会显示教师的个人课表，点击某个课程要看的学生，点击查看学生则会列出所有选择该课的学生然后点击下方导出学生名单会把选择该课的所有学生信息导出excel表格，学生信息包括（学生学号，学生姓名，院系，专业，年龄，性别，入学日期）

3.11 学生选课

学生选课是用于学生来选课的功能，是本系统的核心用例，本系统的大多数使用者是学生。学生可以在学生选课功能上进行开始选课、查看个人课表、退选课程、查看上课老师的操作



图9学生选课用例图

（1）查看开课课程：学生登录选课后台后，跳转到学生首页，点击左侧菜单栏选择”学生选课”，点击”开始选课”，则会转入开始选课页面列出所有开课的课程（前5个可翻页）。开课的信息包括（课程代码、课程名、课程性质、学分、上课老师、上课人数、剩余人数、课时间、上课地点、开课时间）。上面还有条件栏，可以进行按条件搜索课程，可以按课程号精致搜索，或按教师名搜索该教师的课程，上课地点搜索课程，上课时间搜索课程，上课性质搜索课程，还有通过课程名模糊搜索课程。

（2）选课：学生登录选课后台后，跳转到学生首页，点击左侧菜单栏选择”学生选课”，点击”开始选课”，则会转入开始选课页面列出所有开课的课程（前5个可翻页）。这时学生点击想要选择的课程，点击选课按钮，则会弹出提示框询问是否选课，点确定则会选择该科在个人课表会增加一条记录，弹出提示框成功选择该课。

（3）查看上课老师（学生可以查看那些上课老师的信息）：学生登录选课后台后，跳转到学生首页，点击左侧菜单栏选择”学生选课”，点击”查看上课老师”，则进入老师信息页面，列出所有老师信息（前5个可），在教师查询页面的上方有查询条件可以选择，可以选择精确搜索输入教师工号查询教师，或者输入教师姓名、院系、入职日期进行模糊匹配查询。

（4）查看个人课表（学生可以查看自己的个人课程表）：学生登录选课后台后，跳转到学生首页，点击左侧菜单栏选择”学生选课”，点击”个人课表”，则会列出所有的个人课表，个人课表信息包括（课程代码、课程名称、课程性质、学分、上课老师、上课地点、上课时间）

（5）退选课程（学生可以退选自己已选课程）：学生登录选课后台后，跳转到学生首页，点击左侧菜单栏选择”学生选课”，点击”退选课程”，则会列出所有的个人课表，个人课表信息包括（课程代码、课程名称、课程性质、学分、上课老师、上课地点、上课时间），选择自己要退选的课程，点击退选按钮，弹出对话框询问是否退选，点击确定则会把本次修改提交系统，反之则撤回当前操作。

# 

# 4系统设计

系统的设计部分是根据对学生选课和教师开课这两个核心功能点来分析，并基于这两个分析结果来进行功能设计。采用模块化划分，讲系统的功能点划分为三个大模块。并在这三个大模块中划分。并在这三个模块的基础下继续划分，抽离出公共部分。并继续划分，每个大的功能点再继续细化。采用这样的设计方式，使得模块和模块之间的关系以及模块内部直接的联系更加清晰，能够着眼于整个系统。抽离出公共部分能够减少开发编码的工作量，使得设计更加低耦合高内聚。系统划分成一个一个小的功能点。能够更好地估算工作量，清楚了解目前的开发进度。

## 4.1 系统体系结构设计

4.1.1 restful风格架构

本系统的大致体系呈现出restful的风格，即是前后端完全独立分离。前后端交互的整个过程采用http协议通信，交互数据采用标准格式(json)。基于统一接口和通信载体，前端后端直接的改动互不影响，不同于以往传统的软件系统将前端代码耦合到后端代码中，将前后端代码解耦。



图10 restful风格架构图

4.1.2 Nginx分发请求

采用Nginx服务器来进行反向代理和请求分化，Nginx服务器能够接受更大的请求数和连接数相比以往传统软件系统能够支持更大的并发量。在学生选课高峰的时候可以多开实例通过Nginx负载均衡，解决学生选课高峰选课难问题。Nginx分发URL资源请求和API接口请求，API接口请求交由tomcat处理，使得前后端彻底分离。

图11 Nginx请求分发图

4.1.3 SSM架构体系

运行在tomcat中的Java web程序采用SSM框架技术开发整个架构按照MVC架构。Spring作为一个轻量级的IOC和AOP容器用作于项目中，掌管着web应用中所有bean的生命周期，维护整个应用中的对象的创建、初始化、销毁以及对象直接的关系。SpringMVC作为控制层，承担着接受处理来自浏览器的API接口请求，并作文restful中接口提供者。Myabtis作为持久层，即数据库访问层。完成对数据库的操作。

应用架构分为三层：Controller控制层接受和处理API接口请求、Service业务层处理项目中的业务逻辑和事务控制、Dao层数据库访问层完成对数据库的操作。



图12 SSM架构图

4.1.3系统整体架构设计

结合上面需求分析，本系统分三个模块。且三个模块之间不能跨越访问，抽出其中公公共部分，在业务层中调用。使用拦截器，用于拦截跨越请求和验证登录状态。结合以上需求和技术设计，总结系统架构设计图如下。



图13系统架构图

## 4.2 系统特性

采用面向对象的设计方式和restful的风格设计使得系统有如下特性：

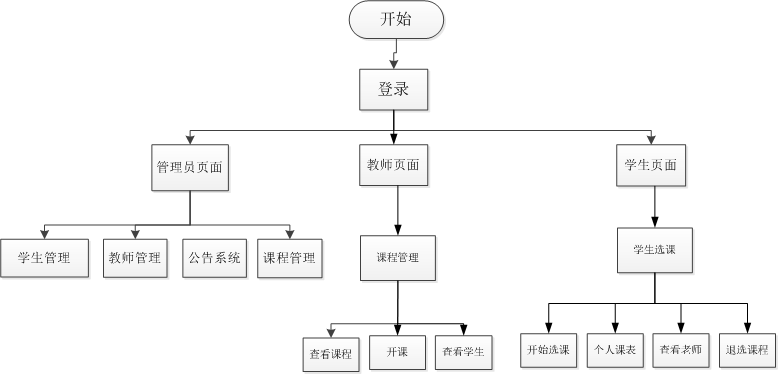
类分层结构：系统按照按照不同功能的类来将分层，使得各个类包有不同的功能各司其职根据代码职能划分为：Controller（控制层）、Service（业务逻辑层）、Dao（数据访问层）、Util（工具包）、Entity（实体类层）。划分后使得系统代码结构清晰，能够更加清晰了解不太模块直接关系，方便开发与维护。

Restful风格：前后端的彻底分离，前后端的代码变动互相不收影响。强调http应当以资源为中心。具备以下特点1.资源通过URL来指定和操作。2.对资源的操作包括获取、创建、修改和删除，正好对应HTTP协议的GET,POST,PUT和DELETE方法。3.连接是无状态性的。4.能够利用Cache机制来提高性能。

面向切面编程：将系统功能的代码和业务功能的代码彻底分开在运行时候通过动态代理织入。

## 4.3 功能设计

系统分由三个大模块，划分出不同的小功能。

 图14系统总功能介绍图

### 4.3.1 登陆

登陆功能用于对用户身份进行身份校验，在用户登录系统首页的时候第一个遇到的功能。用户输入自己的帐号和密码能够成功进入系统。系统根据用户不同的登录方式，来判断用户的不同身份，然后根据不同的用户身份来进行用户名和密码匹配，若匹配成功则能进入系统。反之，则匹配失败弹出提示框提示用户密码或用户名错误。



图15登陆模块流程图

4.3.2 修改密码

修改密码：用户登录后进入根据不同身份进入不同的主页，但不同主页在左侧菜单栏都有密码修改菜单，点击“密码修改”后转入修改密码页面，先输入原密码，再输入新密码和确认密码（跟新密码一致）。点击保存，则提交到系统判断，反之则取消当前操作。提交到系统判断则会先判断原密码是否一致，若不是则提示原密码错误，然后判断新密码和确认密码是否一样，否则提示用户两次密码不一致。



图16修改密码流程图

4.3.3学生管理管理

学生管理包括查询“添加学生信息”、“修改学生信息”、“查看农产品信息”、“删除学生信息”功能。

（1）“查询学生信息”：管理员可以查看所有的学生信息还提供搜索方法可以通过输入学号（精准查询）或学生名字模糊查询或院系或入学日期来查询，。

（2）“修改学生信息”：管理员查看所有的学生信息，系统列出管理员查询出的学生选修。选择要修改的学生信息，页面会弹出学生信息简略图，管理员可以更改学生信息，输入完所有学生信息后，点击保存按钮会询问是否保存当前修改，反之回到修改页面，若点确定则将本次修改提交到系统。系统会刷新一次，管理员会看到本次修改结果。

（3）“添加学生信息”：管理员可以添加导入学生信息，进入学生管理页面，点击“添加学生信息”按钮，在输入框中输入学生学号、学生姓名、学生性别、年龄、院系、专业、入学日期，选择“保存”按钮，系统提示是否要添加该学生信息，点击确定，则把本次数据提交到系统中，反之撤回当前操作。

（4）“删除学生信息”：管理员查看所有的学生信息，系统列出管理员查询出的学生选修。选择要删除的学生信息，页面会弹出对话框询问是否删除若点确定则将此学生信息从系统中删除，反之则不做任何操作。系统会刷新一次，管理员会看到本次修改结果。。

## 4.3.4 教师信息管理

教师信息管理模块为管理员提供管理教师信息的功能，管理员可以查看教师信息、增加教师信息、修改教师信息、删除教师信息。教师基本信息包括教师工号、教师姓名、教师院系、入职日期、教师简介

（1）查询教师信息（通过教师工号、教师姓名或入学日期、院系查询教师信息信息）：管理员登录之后进入了教师管理页面，即列出所有的教师信息（前5个）。在教师查询页面的上方有查询条件可以选择，可以选择精确搜索输入教师工号查询教师，或者输入教师姓名、院系、入职日期进行模糊匹配查询。

（2）修改教师信息：管理员登录之后进入了管理员首页，点击左侧菜单栏的“课程管理”菜单中的“修改学生信息”，根即列出所有的学生信息（前5个）。查看先显示出所有的学生信息，根据系统输入的条件进行查询出要删除的学生信息。选择某个要修改的学生，点击修改按钮，会弹出学生信息简略图，修改完毕，点击保存会提示学生信息修改成功，会提示是否提交这次修改，确定则提交到系统中判断，反之则撤回本次操作。提交到系统的时候，系统会对输入的信息进行格式判断和完整性判断，若通过判断则提交本次修改提示修改成功，反之则提示格式错误或信息不完整。

（3）添加教师信息（将教师信息导入到系统中）：管理员登录之后进入了管理员首页，点击左侧菜单栏的“教师管理”菜单中的“添加教师信息”，会弹出增加教师页面，填写教师信息，如工号（密码会自动生成默认跟学号一样），姓名，入职日期、院系。填写完毕后点击保存，会进入信息验证，信息验证会验证信息完整性和格式，若通过判断则提示添加成功，反之会提示信息不完整或格式不对。

（4）删除教师信息：管理员登录之后进入了管理员首页，点击左侧菜单栏的“教师管理”菜单中的“删除教师信息”，查看先显示出所有的教师信息，根据系统输入的条件进行查询出要删除的教师信息。选择要删除的教师信息然后点击删除按钮，点击后会弹出提示框询问是否删除，若点确定则会提示删除成功，反之则撤回当前操作。

## 4.3.5管理员课程管理

课程管理是针对管理员对课程的编排和修改等一些列基本操作。课程信息包括课程号、课程名称、学分、课程性质、课程简介，主要是关于课程信息的增删查改操作。

（1）查询课程信息（通过课程号码、课程名称或课程性质程信息）：管理员登录之后进入了管理员首页，点击左侧菜单栏的“课程管理”菜单中的“查询教师信息”，即列出所有的课程信息（前5个）。在教师查询页面的上方有查询条件可以选择，可以选择精确搜索输入教课程号查询课程，或者输入课程名、课程性质进行模糊匹配查询。

（2）修改课程信息：管理员登录之后进入了管理员首页，点击左侧菜单栏的“课程管理”菜单中的“修改课程信息”，根即列出所有的学生信息（前5个）。查看先显示出所有的课程信息，根据系统输入的条件进行查询出要修改的课程信息。选择某个要修改的课程，点击修改按钮，会弹出课程信息简略图，修改完毕，点击保存会提示课程信息修改成功，会提示是否提交这次修改，确定则提交这次修改到系统进行系统验证，反之则撤回本次操作。系统验证会判断用户输入信息的完整性和格式是否正确，若通过判断，则将这次修改的信息提交到系统中，并提示用户修改成功，刷新一次页面保证用户看到最新修改的数据，反之则提示用户信息不完整或格式问题并撤回本次操作

（3）添加课程信息（将课程信息导入到系统中）：管理员登录之后进入了管理员首页，点击左侧菜单栏的“课程管理”菜单中的“添加课程信息”，会弹出增加教师页面，填写课程信息，如课程号，课程名称，学分、课程性质。填写完毕后点击保存，会将信息提交到系统中进行系统信息验证，判断信息完整性和格式的正确性，若通过判断则把信息录入到系统中并提示用户添加课程信息成功，反之则提示信息不完整或格式不正确，并撤回本次的添加课程信息操作

（4）删除课程信息：管理员登录之后进入了管理员首页，点击左侧菜单栏的“课程管理”菜单中的“删除课程信息”，查看先显示出所有的课程信息，根据系统输入的条件进行查询出要删除的课程信息。选择要删除的课程信息然后点击删除按钮，点击后会弹出提示框询问是否删除，若点确定则会提示删除成功，反之则撤回当前操作。

## 4.3.5 公告系统

公告系统用于告知所有的系统使用者的信息（如开始选课），一般公告有5到10条，由管理员发布公告，下架公告，编辑公告。普通用户如学生和教师可以查看公告。

（1）查看公告：用户进入用户自己的主页后，点击公告系统，则会弹出最近的公告。

（2）发布公告：发布公告的主要使用者是管理员，管理员进入管理员主页后，点击公告系统，转入公告系统主页点击发布公告，转入发布公告页面，编写公告标题和内容，点保存则提交到系统做判断，判断格式是否正确和字数限制，若通过判断则提示管理员发布成功，反则提示字数或格式不正确。若点击返回按钮则不做任何操作

（3）编辑公告：管理员可以编辑已经发布了的公告，进入公告系统查看已经在线公告选择其中一个点击编辑，修改公告内容后提交到系统进行字数和格式验证，通过后加入系统提示修改成功。

## （4）下架公告：管理员进入公告系统页面，查看在线的公告点击下架，询问是否下架确定则下架该公告。

## 4.3.7 个人信息管理

学生和教师都可以进行个人信息管理，学生或老师进入其主页的时候。可以浏览到个人的信息，点击修改按钮则解除个人信息的项目只读的限制，可以编辑个人信息，但学号和工号不能修改，修改完毕后点击保存则可以提交本次修改，系统判断这次提交信息格式是否正确和信息是否完整，若验证通过则提交到系统，反之撤回本次操作。



图16修改个人信息流程图

## 4.3.8 教师课程管理

教师课程管理包括查看课程，开设课程、查看个人课表、导出学生名单、查看选课学生。

（1）查看课程：教师进入教师页面后点击课程管理后点击查看课程可以列出所有的课程信息、并可以根据课程号和课程名查询

（2）查看个人课表：教师进入教师主页，点击课程管理一级菜单后再点击查看个人课表。可以看到自己开设课程的课程名、课程代码、上课时间、上课地点。

（3）查看上课学生：教师查看了个人课表后可以点击查看上课学生则可以查看自己开设的课程上课学生并可以导出excel表格。

（4）安排课程：这个功能点为教师的核心功能点，教师找出想要开课的课程，点击排课按钮，会弹出对话框，填开课信息，上课时间、上课地点、上课人数。提交系统判断，系统判断空间和时间是否冲突后，通过会提示开课成功，反之提示失败信息。



图17学生选课流程图

## 4.3.9 学生选课

学生选课功能包括开始选课、查看个人课表、查看教师、退选课程

（1）开始选课：学生进入主页后点击左侧开始选课进入选课页面，列出所有已经开设的课程，可以通过课程号、上课老师、上课时间、上课地点选择想要上的课程，点击选课后询问是否选课，点确定则提交到系统。系统判断这课程上课人数是否达到上限，若是则提示该课程已经满人。继续判断学生的上课时间是否冲突在同一时间内是否有别的课程，若是提示时间冲突。然后改课程的剩余减1，添加上学生选课的记录。提示选课成功。



图18学生选课流程图

（2）查看上课老师（通过教师工号、教师姓名或入学日期、院系查询教师信息信息）：学生登录后点击查看上课老师，即列出所有的教师信息（前5个）。在教师查询页面的上方有查询条件可以选择，可以选择精确搜索输入教师工号查询教师，或者输入教师姓名、院系、入职日期进行模糊匹配查询。

（3）查看个人课表：学生登录后点击个人课表，可以列出所有学生已经选择课程信息。课程信息包括（课程代码、课程名称、课程性质、学分、上课老师、上课地点、开课时间）

（4）退选课程：学生点击退选课程按钮，进入退选页面，先判断是否过了退选时间，若过了则提示已过了退选时间不能退选，列出所有已经选课的课程信息。选择一个要退选的课程点击退选。提交到系统，该课程的剩余人数+1，并删除选课记录。



图19退选流程图

## 4.4 数据库设计

### 4.4.1 E-R图

系统中三类使用者，管理员、学生、教师。管理员可以发布多个公告。在一个学院之中有多个专业，而一个专业只能属于一个学院。一个专业中有多个学生。教师可以安排多个课程，一个课程也可以由不同的老师来上。学生可以选择多个课程而一个课程又可以有多个学生上课，由此关系画出er图。

图20系统E-R图

4.4.2 物理数据库

根据er图得出每一个实体的对应关系，如学院与专业关系是一对多关系。专业对学生是一对多关系。老师与课程是多对多关系等。根据这些实体不同的关系，设计出数据库物理结构图如下



图21系统物理数据库图

### 4.4.3 表结构

（1）管理员表主要存储管理帐号信息

表1系统管理员表（admin）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| Username | Varchar(15) | 管理员帐号(主键) |
| password | Varchar(15) | 密码 |

（2）排课表记录老师编排课程的信息用于学生选课

表2排课表（arrange）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| Id | Int | 排课id（主键） |
| Cno | Varchar(15) | 课程编号 （外键） |
| Tno | Varchar(15) | 教师编号 （外键） |
| Anum | int | 剩余人数 |
| Maxnum | int | 最大人数 |
| Atime | Int | 上课时间 |
| Address | Int | 上课地点 |
| Sd | Date | 开课时间 |

（3）课程表记录课程信息

表3课程信息表（course）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| Cno | Varchar(15) | 课程编号(主键) |
| Cname | String | 课程名称 |
| credit | int | 学分 |
| Type | Tinint | 售价 |
| State | Tinint | 课程状态 |
| Comment | Varchar(255) | 课程简介 |

（4）院系表记录院系信息

表4院系表（department）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| did | int | 院系编号(主键) |
| dname | Varchar(15) | 学院名称 |
| comment | Varchar(255) | 简介 |

（5）专业表：记录专业信息

表5专业表（major）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| Mid | int | 专业id(主键) |
| did | int | 学院id(外键) |
| Mname | Varchar(15) | 专业名称 |
| Comment | Varchar(15) | 简介 |

（6）广告表：存放公告轮播消息

表6公告信息表（notice）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| id | int | 公告编号(主键) |
| Creator | Varchar(15) | 创建者帐号(外键) |
| Content | Varchar(255) | 公告内容 |
| Dt | date | 更新或创建日期 |

（7）学生选课表：存放学生记录

表7学生选课表（sc）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| Aid | int | 排课id (外键) |
| Sno | Varchar(15) | 学生学号(外键) |

（8）学生表：存放学生帐号以及学生个人信息

表8学生表（student）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| Sno | Varchar(15) | 学生学号(主键) |
| Sname | Varchar(15) | 学生姓名 |
| Password | Varchar(15) | 密码 |
| Sex | Tinint | 性别 |
| Age | Int | 年龄 |
| Mid | Int | 专业id(外键) |
| Dt | Date | 入学日期 |

（9）教师表：记录教师帐号信息和教师个人信息

表9教师表（teacher）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| Tno | Varchar(15) | 教师工号(主键) |
| Tname | Varchar(15) | 教师姓名 |
| Password | Varchar(15) | 密码 |
| Did | Int | 学院id(外键) |
| Dt | Date | 入职日期 |
| Comment | Varchar(255) | 教师简介 |

# 5 系统实现与界面展示

## 5.1 登录主页

在浏览器地址栏输入 <http://xuanke.com>

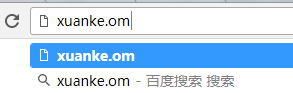


图22地址栏输入示意图

用户进入登录主页页面，有三个单选框可以选择不同登录方式



图23登录主页图

域名配置方式如下

server {

listen 80;

server\_name xuanke.com;

location ~ ^/(sscm/api){

proxy\_pass 127.0.0.1:8080 //Java web 后端地址

}

……

}

## 5.2 管理员主页

管理员页面是一个单页应用，左侧是菜单栏，显示管理员可以进行的操作。管理可以点开一级菜单选择不同的功能。页面中间是学生信息在管理员进入主页的时候自动加载前学生信息。



图24管理员主页图

管理员点击二级菜单的时候会通过载入子页面的方式载入子页面，并生成tag标签在上方方便切换



图24管理员主页tag图

实现如下

（1）登录主页自动加载

function actionFn(node, url) {

return tab.add({

'id':node.id,

'title':node.text,

closable:true, //通过html载入目标页

html:'<iframe scrolling="auto" frameborder="0" width="100%" height="100%" src="' + url + '"></iframe>'

});

}

（2）在左侧菜单栏添加菜单

var root3 = new Ext.tree.TreeNode({id:"root3",text:"根"});

root3.appendChild(QueryTeacher);

root3.appendChild(AddTeacher);

（3）生成tag标签

var tab = new Ext.TabPanel({

region:'center',

deferredRender:false,

activeTab:0,

resizeTabs:true, // turn on tab resizing

minTabWidth: 115,

tabWidth:135,

enableTabScroll:true,

html:'<iframe scrolling="auto" frameborder="0" width="100%" height="100%" src="queryStudent.html"></iframe>'

});

5.3添加学生页面

进入添加学生页面，输入框可以填写学生相关信息，下拉框中有获取到的学院信息可以选择，点击提交后会先验证信息完整性，若不完整则会出现相关提示。点击重置按钮可以重置所有输入过信息。



图25添加学生页面图

若没有填写完整信息点击提交



图26添加学生页面错误提示图

填写完整信息点提交

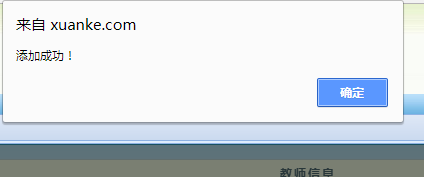


图27添加学生页面成功提示图

5.4教师个人课表页面

教师点击个人课表会列出该教师所有排课的信息

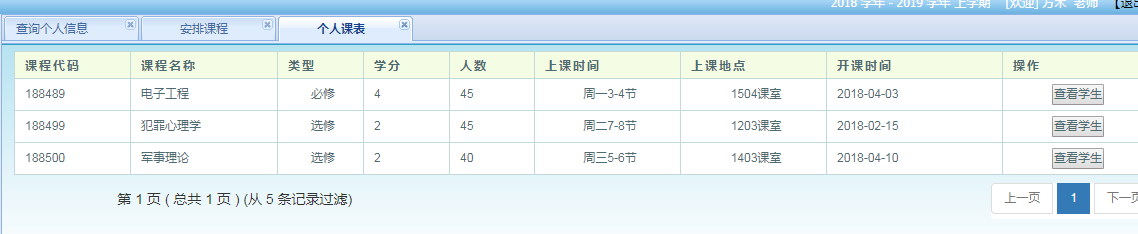


图28教师个人课表图

点击查看学生按钮可以显示选该课程的所有学生



图29查看学生图

点击导出学生名单

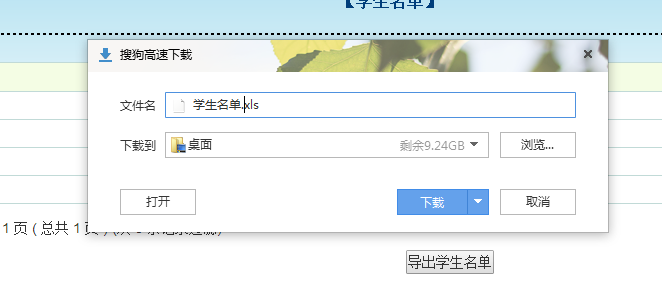


图30导出学生图

实现原理：先获取到所有选择该课的学生名单，然后构建excel表格文件，然后将该文件作为输出流，返回到前端，前端将该输出流中的字节码反序列化为文件中则可以完成excel表格下载的功能。

自定义的filed接口，实现该接口的类可以构成自定义的excel类

**public** **interface** Filed {

**public** String[] getFiled();//返回自己属性值的字符串数组

**public** **int** getNum();//返回属性的个数

}

自定义Excel 类

**public** **class** Excel {

**public** **void** setContent(List<? **extends** Filed> filed){

**if** (filed==**null**||filed.size()==0) {

**return**;

}

content = **new** Label[filed.size()][filed.get(0).getNum()];

**for** (**int** i = 0; i < filed.size(); i++) {

String[] row = filed.get(i).getFiled();

**for**(**int** j=0;j<row.length;j++){

content[i][j] = **new** Label(j, i+1, row[j]);

}

}

}

**public** **void** wirte(OutputStream os) **throws** IOException, RowsExceededException, WriteException{

**if** (header==**null**||header.length==0) {

**return**;

}

WritableWorkbook wbook = Workbook.*createWorkbook*(os);

WritableSheet wsheet = wbook.createSheet(title, 0); // sheet名称

**for** (Label label : header) {

wsheet.addCell(label);

}

**if**(content!=**null**&&content.length!=0){

**for** (Label[] labels : content) {

**for** (Label label : labels) {

wsheet.addCell(label);

}

}

}

wbook.write();

wbook.close();

}

5.5学生选课页面

学生选课页面先展示现有的课程信息，可以通过课程号查询也可以通过课程名等条件模糊匹配。实现了翻页查找不同课程



图31学生选课图

输入搜索条件搜索某个老师的课程



图32搜索课程图

实现原理：采用datatables 插件将返回的json加载成可分页表格。对返回json有格式要求，需要返回list（加载表格中的数据）该数据的总数和可显示数量。

自定义DatatablesViewPage类作为返回格式模板。

**public** **class** DatatablesViewPage<T> {

**private** List<T> aaData; //aaData 与datatales 加载的“dataSrc"对应

**private** **int** iTotalDisplayRecords;

**private** **int** iTotalRecords;

**public** DatatablesViewPage() {

}

//……getset函数

｝

Controller代码

@RequestMapping(value="/student/getArrange", method=RequestMethod.***GET***)

@ResponseBody

**public** DatatablesViewPage<Arrange> getArrangeByStudent(HttpServletRequest request,HttpServletResponse response){

response.reset();

**try** {

**int** start =Integer.*parseInt*(request.getParameter("start"));

**int** length = Integer.*parseInt*(request.getParameter("length"));

DatatablesViewPage<Arrange> view = arrangeService.getViewS(start, length, request.getParameter("cno"),

request.getParameter("cname"), request.getParameter("type"), request.getParameter("tname"),

request.getParameter("atime"), request.getParameter("address"));

**return** view;

} **catch** (Exception e) {

*logger*.info("getArrangeByStudent 出错了！",e);

**return** **null**;

}

}

学生在找到要选的课程点击选课



图33确认选课图

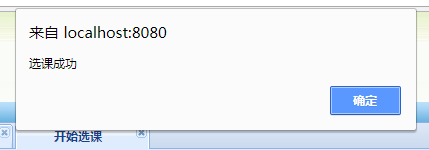


图34成功选课提示图

实现核心代码：先判断改学生同一时间有无相同课程，然后将该课程人数减一，写选课记录。若出现学生重复选择课程会抛出异常，事务回滚提醒学生不能重复选课。

Controller层代码

@RequestMapping(value="/student/selectCourse", method=RequestMethod.***POST***)

**public** **void** selectCourse(**int** id,**int** time,HttpServletRequest request,HttpServletResponse response) **throws** IOException{

PrintWriter out = response.getWriter();

HttpSession session = request.getSession();

**try** {

Student student = (Student) session.getAttribute("student");

**if**(student==**null**){

out.print("err");

**return**;

}

**if** (arrangeService.selected(student.getSno(), time)) {

arrangeService.arrangeCourse(id, student.getSno());

}**else** {

out.print("wrong");

**return**;

}

out.print("true");

} **catch**(ZeroException e){

*logger*.info(e);

out.print("fail");

}

**catch** (Exception e) {

*logger*.info("selectCourse",e);

out.print("err");

}

}

Service层代码

@Override

@Transactional //spring 事务开启

**public** **void** arrangeCourse(**int** id, String sno) **throws** ZeroException {

**int** num = arrangeDao.queryCount(id);

**if**(num<=0) **throw** **new** ZeroException();

arrangeDao.selectArrange(id);

arrangeDao.selectCourse(id, sno);

}

自定义ZeroException 在剩余人数为0时候抛出

**public** **class** ZeroException **extends** Exception {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

@Override

**public** String toString() {

**return** "选择剩余数量为0的课";

}

}

为选课记录表加上唯一索引若重复插入会抛出DuplicateKeyException 捕获该异常提醒用户。

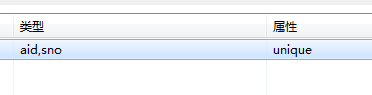


图35唯一索引图

# 6总结与展望

## 6.1总结

在开始阶段开始对学生选课系统的调研，登录学校的正方系统查询已有的教务管理系统。调研学生和教师的需求，确立好需求后开始编写程序需求。拿起大二学习的Java书属性Java的基础，待了几个星期图书馆查询相关技术资料实现。去看博客再看看比较前沿的web技术。在宿舍辛苦编码几个月之后，终于完成了学生选课系统了。本系统贴合潮流的web技术，能够更加适应时代的变迁和需求变迁，能够满足大多数高校在选课方面的教务系统要求。

在开发过程中遇到不少挫折，一开始在写这课题的时候。总觉得选课教务系统多年使用，需求之类自我比较理解。但是在深入工作之后发现之前需求凌乱不得不放弃之前工作。重新开始调研需求，并对系统的详细设计做了很多更改。在有细致详细的设计下，编码工作变得一目了然。在开发的过程中，也遇到不少问题，如前端页面不懂。通过不断查找资料，在图书馆找到不少的相关书籍。解决了前端编写问题，掌握了很多很好的开发技巧。

完成这个项目之后，对于web技术的认知更加深刻了。我对软件开发流程的认知也更加深刻了。在以后的学习工作中我会进步的更加快。

## 6.2 展望

在需求方面，由于一开始自以为已经十分熟悉教务系统。而没有认真仔细去做需求分析，在分析设计的时候，总是能找到之前需求不全面问题。会在设计某个需求的时候想到一个之前没有考虑到的需求

在设计方面，某些地方没有设计全面。例如没有设计权限管理，只用拦截器粗略地实现了简单权限限制。应该将学生、教师、管理员抽象成用户、然后分配不同的权限组。根据权限组分配权限，定义一个超级管理员管理这些权限。

对于学生选课高峰时期的并发处理方面没有做好，在处理请求前，应该加一个redis缓存处理请求。将后端服务做成无状态服务，session信息交由redis存储，所有后端服务共享session。在学生选课高峰的时候可以加多一个后端服务实例。做成无状态服务可以更加灵活扩展，应对特殊情况请求量变大（如选课高峰）

# 

# 参 考 文 献

[1] Y. Daniel Liang著；戴开宇译；Java语言程序设计（基础篇）[Z]；机械工业出版社；2016.

[2]李志秀, 张军, 陈光,等. JQuery Ajax异步处理JSON数据在项目管理系统中的应用[J]. 云南大学学报自然科学版, 2011,(s2):247-250.

[3] 钱乐秋，赵文耘，朱俊钰著；软件工程第2版[Z]；清华大学出版社；2013.

[4]《数据库系统概论第四版》---------------高等教育出版社------------- 郑若忠 王鸿武 著

[5]代威, 董运成. 基于jQuery EasyUI与Spring MVC框架的信息管理系统自动提示的设计与实现[J]. 信息与电脑, 2016,(13):33-34.

[6]《UML系统建模与分析设计》------------机械工业出版社-------------刁成嘉主编

# 

# 致 谢

在落笔之时，回首大学四年已经过去了，在这四年中感谢每一个帮过我的人。感谢那些陪我走过这一段难忘的岁月。

首先我感谢我的指导老师，张义青老师。在编写毕业设计和毕业论文的时候，张老师推荐了我不少关于技术的书，还跟我畅谈了关于java的看法和理解。使我受益匪浅。我万分感谢能够从老师中学到不少知识。

另外，感谢大学四年里的同班同学，是你们陪我一起走过这一段难忘的时光，四年中互相帮助，相亲相爱使得我大学生活更加丰富多彩。

回首四年过去得太快了，感谢遇到每一个人，使得我人生更加丰富多彩。感谢在华农生活的四年时光。感谢每一个老师的孜孜不倦。感谢我们这个可爱的校园。**华南农业大学**

**本科生毕业论文成绩评定表**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 学号 |  | | | | | 姓名 | |  | 专业 |  | | |
| 毕业论文题目 | | |  | | | | | | | | | |
| 指导教师评语  成绩（百分制）：　　　　　　　　　　　　　　　　　　　　指导教师签名：　　　　　　　　　　　　　　　年　　月　　日 | | | | | | | | | | | | |
| 评  阅  人  评  语  及  成  绩  评  定 | | 成绩  评定  标准 | | **评分项目** | | | | | | | **分值** | **得分** |
| 选题  质量  20% | 1 | | 专业培养目标 | | | | 5 |  |
| 2 | | 课题难易度与工作量 | | | | 10 |  |
| 3 | | 理论意义或生产实践意义 | | | | 5 |  |
| 能力  水平40% | 4 | | 查阅文献资料与综合运用知识能力 | | | | 10 |  |
| 5 | | 研究方案的设计能力 | | | | 10 |  |
| 6 | | 研究方法和手段的运用能力 | | | | 10 |  |
| 7 | | 外文应用能力 | | | | 10 |  |
| 成果  质量40% | 8 | | 写作水平与写作规范 | | | | 20 |  |
| 9 | | 研究结果的理论或实际应用价值 | | | | 20 |  |
| 评阅人评语：  成绩（百分制）：　　　　　　　　　　　　　评阅人签名：　　　　　　　　　　　　年　　月　　日 | | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 答  辩委员会  意见与  成  绩  评  定 | 评价项目 | 具体要求（A级标准） | 最高分 | 评分 | | | | |
| A | B | C | D | E |
| 论文质量 | 论文（设计）结构严谨，逻辑性强；有一定的学术价值或实用价值；文字表达准确流畅；论文格式规范；图表（或图纸）规范、符合要求。 | 60 | 55-60 | 49-54 | 43-48 | 37-42 | ≤36 |
|  |  |  |  |  |
| 论文报告、讲解 | 思路清晰；概念清楚，重点（创新点）突出；语言表达准确；报告时间、节奏掌握好。 | 20 | 19-20 | 17-18 | 15-16 | 13-14 | ≤12 |
|  |  |  |  |  |
| 答辩情况 | 答辩态度认真，能准确回答问题 | 20 | 19-20 | 17-18 | 15-16 | 13-14 | ≤12 |
|  |  |  |  |  |
| 是否同意通过论文答辩（打√）   1. 同意 2. 不同意   成绩（百分制）：　　　　　　　　 答辩委员会主席（签名）：  　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　 年 月 日 | | | | | | | |

续上表：

|  |  |
| --- | --- |
| 成绩总评 | 论文总评分数：  论文成绩总评等级：　　　 学院盖章：  年 月 日 |

**注：1、**论文成绩评定等级：参考评阅人的评阅、指导教师评阅情况，结合答辩情况，建议按指导教师评分、评阅人评分、答辩评分为4∶3∶3的比例评定论文总成绩分数，然后按优（90－100分）、良（80－89分）、中（70－79分）、及格（60－69分）、不及格（<60分）给出成绩等级。**2**、论文成绩以分数的形式登记到教务管理系统。

**华南农业大学**

**本科生毕业设计成绩评定表**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 学号 | |  | | 姓名 |  | 专业 |  |
| 毕业设计题目 | | |  | | | | |
| 指导教师评语  成绩（百分制）：　　　　　　　　　　　　　　　指导教师签名：　　　　　　　　　　　年　　月　　日 | | | | | | | |
| 评阅人评语  成绩（百分制）：　　　　　　　　　　　　　　　评阅人签名：　　　　　　　　　　　　年　　月　　日 | | | | | | | |
| 答辩委员会评语  成绩（百分制）：　　　　　　　　　　　　　　　答辩委员会签名：　　　　　　　　　　年　　月　　日 | | | | | | | |
| 成绩总评 | 毕业设计总评分数：  毕业设计成绩总评等级：　　　 答辩委员会主席（签名）：  年 月 日 | | | | | | |

**注：1、**毕业设计成绩评定等级：参考评阅人的评阅、指导教师评阅情况，结合答辩情况，建议按指导教师评分、评阅人评分、答辩评分为4∶3∶3的比例评定论文总成绩分数，然后按优（90－100分）、良（80－89分）、中（70－79分）、及格（60－69分）、不及格（<60分）给出成绩等级。**2**、论文成绩以分数的形式登记到教务管理系统。