

学士学位论文

题目：面向学院的教务管理系统

——培养方案管理系统

设计人：王跃跃

指导教师：谢斌红

所属系部：计算机科学与技术学院

专业班级：软件工程 142001

学 号：201420020131

2018 年 6 月 7 日

中文摘要

高等学府的规模在不断扩大，学生和教师的数量都有所增加，导致了参与教学管理工作的人员越来越多。传统的教学管理工作都是采用人工管理，以纸质文档加以辅助，这样不仅效率低下，而且随着时间的推进，对于纸质文档的管理也成了一大难题。信息管理技术的发展日趋成熟，当务之急就是开发一款适应于面向学院的教务管理系统。把学院平时教学管理工作中涉及到的教学信息有效的整合起来并且集中存储管理。把教学管理工作由传统的人工管理方式迁移到以计算机辅助人工的方式，可以有效的提高管理工作的效率。减轻教师和学生处理教学工作的负担，教师可以把更多的时间关注在教学任务中，学生也可以把更多的精力放在学习本身上。

本系统基于以上叙述，把软件工程中的“自上而下的设计，自下而上的实现”的思想应用进入系统开发的全过程。通过分析我校计算机科学与技术学院的教学管理工作内容，重点剖析了现行的管理工作的不足之处，为之量身打造了一套行之有效的教学管理系统。

关键词：教学管理；软件工程；效率；辅助

College oriented Educational Administration Management System

Author: WangYueYue Tutor: Xie Binhong

Abstract

The number of students and teachers has increased, resulting in more and more people participating in teaching management. In the traditional teaching management work, manual management is used to assist the paper documents, which is not only inefficient, but also becomes a big problem in the management of paper documents with the development of time. The development of information management technology is maturing day by day, the urgent task is to develop a college oriented educational administration system. The effective Integration of the Teaching Information involved in the normal Teaching Management of the College Get up and centralize storage management. The transfer of teaching management from traditional manual management to computer-aided manual management can effectively improve the efficiency of management. By reducing the burden on teachers and students to deal with teaching work, teachers can devote more time to teaching tasks, and students can also devote more energy to learning themselves.

Based on the above description, this system applies the idea of "top-down design, bottom-up realization" into the whole process of system development. By analyzing the contents of teaching management in our college of computer science and technology, this paper emphatically analyzes the shortcomings of the current management work, and builds a set of effective teaching management system for it.

Keywords : teaching management ;Software engineering ;Efficiency ;Auxiliary

目 录

第一章 系统概述.....	1
1.1 系统开发的背景.....	1
1.2 系统开发的意义.....	1
1.3 国内外研究现状.....	2
1.4 系统开发的内容.....	2
1.5 系统开发的目标.....	2
1.6 论文内容安排.....	3
1.7 本章小结.....	3
第二章 开发技术.....	4
2.1 前端技术.....	4
2.1.1 LayUI.....	4
2.1.2 JQuery.....	4
2.2 后端技术.....	5
2.2.1 Spring.....	5
2.2.2 SpringMVC.....	5
2.2.3 MyBatis.....	6
2.2.4 Shiro.....	6
2.3 数据库技术.....	7
2.3.1 MySql.....	7
2.3.2 Redis.....	7
2.4 本章小结.....	8
第三章 需求分析.....	9
3.1 需求分析.....	9
3.1.1 功能性需求分析.....	9
3.1.2 非功能性需求分析.....	12
3.1.3 建立需求模型.....	12
3.1.4 编写数据字典.....	14
3.2 可行性分析.....	16
3.3 本章小结.....	17
第四章 总体设计.....	18
4.1 功能模块设计.....	18
4.2 数据库设计.....	20
4.2.1 概念数据模型设计.....	20
4.2.2 物理数据模型设计.....	20
4.2.3 数据库表结构设计.....	21
4.3 物理配置方案设计.....	25
4.3.1 计算机硬件选择.....	25
4.3.2 计算机软件选择.....	25
4.4 系统结构设计.....	26
4.4.1 前端结构设计.....	26
4.4.2 后端结构设计.....	27

4.5 本章小结.....	28
第五章 详细设计.....	29
5.1 界面设计.....	29
5.1.1 界面设计原则.....	29
5.1.2 界面设计原型.....	29
5.2 功能结构设计.....	30
5.2.1 课程管理.....	30
5.2.2 课程资料管理.....	31
5.1.2 课程库导入导出.....	31
5.1.2 培养方案管理.....	32
5.1.2 培养方案课程管理.....	32
5.1.2 培养方案能力管理.....	32
5.3 本章小结.....	33
第六章 编码实现.....	34
6.1 课程管理.....	34
6.1.1 新增课程.....	34
6.2 课程资料管理.....	37
6.2.1 上传课程资料.....	37
6.3 培养方案管理.....	40
6.3.1 查看培养方案.....	40
6.4 培养方案课程管理.....	42
6.4.1 培养方案排课.....	42
6.5 培养方案能力管理.....	44
6.6 本章小结.....	46
第七章 系统测试.....	47
7.1 课程新增测试.....	47
7.2 导出课程测试.....	47
7.3 上传课程结构图测试.....	48
7.4 课程类别树测试.....	48
7.5 删除培养方案课程测试.....	49
7.6 本章小结.....	49
结束语.....	50
致 谢.....	51
参考文献.....	52

第一章 系统概述

本章对系统进行了简要的概述,确定了为什么要开发本系统以及开发本系统的目标。

1.1 系统开发的背景

中国高等教育的水平越来越高,教学环境也随之日渐复杂。平时教学管理工作中需要接触到的信息也越来越多。但是现行的教学管理工作都是采用以人为主,纸质文档为辅的工作模式。管理人员且多是主张教学的教师,繁重的教学管理工作让他们的工作脱离了自己工作的本质,每天不得已要去处理繁琐的教学管理工作。严重剥夺了教师的教学工作时间,教师每天需要更多的时间去备课、处理教学任务。由于对于信息的管理模式是采用纸质文档的模式,随着时间的推移,每一年都会留下大量的教学管理资料,对于这些纸质文档的管理也成为了一项耗时耗资的工作,而且这些资料中也有大量的重复性文档,管理人员也无法去对这些资料删繁就简。当管理人员想要查询往年的教学资料时,还要从这些海量的资料中去一层一层的寻找,整个工作量极为繁重。而且对于这些信息而言保密性极差,安全性也特别低。一旦这些信息泄露出去将造成不可估量的损失,更为严重的是一旦遇到非人为因素的原因时,例如自然灾害时。这些信息将会受到毁灭性的冲击,直接造成了信息的断层,损失将会更大。

1.2 系统开发的意义

教学管理工作中相关管理人员工作任务的繁重困扰着高校教学管理工作人员很长时间,将管理人员从繁琐的管理工作中解救出来成为当前高校教学管理工作的首要任务。本系统可以有效的解决教学管理人员的许多重复性工作,让他们对于信息的处理方式更加集中简便。把管理人员的工作高效化、简单化,这样就可以为他们提供更多的时间专注于教学任务的工作当中去。同时,当管理工作中涉及到的教学信息积累的越来越多时,这些资料的管理也成为了一个比较头疼的问题。本系统可以为这些教学资料提供一个集中管理的方式,并且更加安全,资料的保密性更高。把全部教学工作中涉及到的教学资料都集中在中央服务器中存储管理,并且会对这些信息进行备份,很大程度上保证了信息的安全性。把管理工作的形式由以人为主转变为以计算机为主。

1.3 国内外研究现状

在国外的一些发达国家之中。他们的科学技术水平相对来说比国内的普遍要高，特别是信息技术的发展，更是引领着世界的潮流。高校工作特别注重效率问题，他们拒绝处理一些简单重复的工作。像类似的教学管理工作他们都交付给计算机软件系统去管理，让计算机帮他们处理繁琐的工作。他们的教学管理体系也做的相对于成熟完美。所以说在国外的发达国家中的教学管理体系还是非常成熟的。

在国内，由于我国近年的教育水平有着迅猛的成长，且我国的互联网成长有着突飞猛进的提高。教育水平的进步直接促使高校的不断扩招，随着高校的不停扩招，教学环境也日渐复杂。已有的教学管理模式已不能适应当前的急速变化的环境。还停留在传统的以纸质管理的方式上，不仅工作效率极为低下，而且安全性和保密性都非常低。还有一些类似的教学管理软件，但是其需求覆盖面不够广，只能处理一些最基本的教学任务需求，无法适应一些需求相对复杂的功能。

1.4 系统开发的内容

本系统核心实现语言使用 JAVA，关系型数据库管理系统使用了 MySQL，数据库建模使用了 PowerDesigner，非关系型数据库采用了业界流行的 Redis 数据库。前端界面设计采取了 LayUI 框架。系统整体开发的内容分为如下六部分。

- 1) 课程库管理：对课程库里的全部课程信息的维护。
- 2) 课程相关资料的维护。这些资料的维护工作以上传附件的形式体现。
- 3) 课程库的导入导出功能：对课程库里的课程进行导入导出功能的实现
- 4) 培养方案管理：对培养方案信息进行管理。
- 5) 培养方案能力管理：管理培养方案中的能力，也涉及到培养方案课程能力的管理。
- 6) 培养方案课程管理：管理培养方案中的课程相关信息。

1.5 系统开发的目标

此次开发的面向学院的教研室管理系统主要可以解决现行教研室管理工作中的管理人员工作繁琐重复的问题，教学资料管理困难的问题也可得到有效的解决，把管理方式集中到互联网平台之中，做到管理方式集中化，高效化，网络化。

最终要做到提升整个教学管理工作的效率。

1.6 论文内容安排

本篇文章围绕着教研管理系统中的课程与培养方案管理模块展开，共分为 7 个章节，内容如下：

第一章 系统概述：对系统进行了比较简要的描述。

第二章 开发技术：简单介绍了本开发本系统所使用到的主要技术。

第三章 需求分析：分析系统的需求

第四章 总体设计：结合需求分析对系统进行总体设计

第五章 详细设计：针对总体设计对系统进行详细设计

第六章 编码实现：参考总体设计与详细设计的内容对系统进行编码实现。

第七章 系统测试：对系统各功能进行测试

1.7 本章小结

本章从系统的现行背景与系统的开发意义开始研究，然后确定了本系统国内外的研究现状。最后确定了系统开发的目标和论文的内容安排。

第二章 开发技术

本系统采用 Java 语言实现其核心代码, Java 是一门优秀的面向对象编程语言, 因为 Java 在吸收了 C++ 的各种优点的同时还去除了 C++ 中许多晦涩难懂的概念, 例如多继承、指针等难以理解的概念。所以使用 Java 在实现功能的时候, 不仅仅是简单易用, 其功能也是非常强大。Java 语言作为一门非常优秀的静态语言, 对于面向对象编程理论有很好的实现, 可以容许开发者在开发程序的过程中兼顾优雅的头脑体例与庞大的编程实现。

本系统最终成果是以 Web 应用程序展示出来, 采用 Java 语言实现本系统的编码实现是一种正确的选择。

本系统的开发环境如下:

- 1) 系统环境: Windows10。
- 2) 开发工具: IntelliJ IDEA, HBuilder。
- 3) 后端框架: Spring, Spring MVC, Mybatis, Shiro。
- 4) 前端技术: LayUI, JQuery。
- 4) 数据库: MySQL, Redis。
- 5) 服务器: Tomcat。
- 6) 管理工具: Git, Maven。
- 7) 建模工具: PowerDesigner。
- 8) 测试工具: Junit。
- 9) JDK: JDK1.7。

2.1 前端技术

本系统前端开发使用到的主要技术有 LayUI 和 JQuery。接下来将介绍这两项技术。

2.1.1 LayUI

Layui 的起点并不是专门为前端开发者服务。LayUI 仍在坚持使用最简单的 HTML/CSS/JavaScript。这样做的优点可以或许让开发者避开非常繁琐的工具配置。LayUI 不仅应用于服务端程序员, 在整个前端社区也受到诸多前端开发人员的热捧与追爱。

2.1.2 JQuery

jQuery 是一个已经封装好的 JavaScript 框架，jQuery 设计的主旨让你做到在写最少的代码容许你做更多的事情。它把 JavaScript 中经常使用的功能代码做到了有用的封装，并且提供了一种简单有用的设计模式。

jQuery 可以兼容各种主流浏览器，如 IE6.0+、FireFox 1.5+、Safari 2.0+、Opera 9.0+等。jQuery 很快成为时下最为流行的 JavaScript 库。并且微软、Google 等业界的大公司都对其进行了相应的支持。正是由于社区活跃的贡献和业界主流公司的支持，jQuery 才可以走的更高、更远。

2.2 后端技术

本系统后端开发使用的核心技术有 Spring，SpringMVC，MyBatis，Shiro。接下来将介绍这四项技术。

2.2.1 Spring

Spring 是一个开源的 Java 开发框架，她的初志是为了取代在那时十分粗笨的 Java EE（当时还称为 J2EE）组件技术 EJB（Enterprise Java Beans），也是为了让 Java EE 开发流程更加简朴灵活，提高开发人员的开发效力。

Spring 官方给出的概览图如下图 2.1 所示

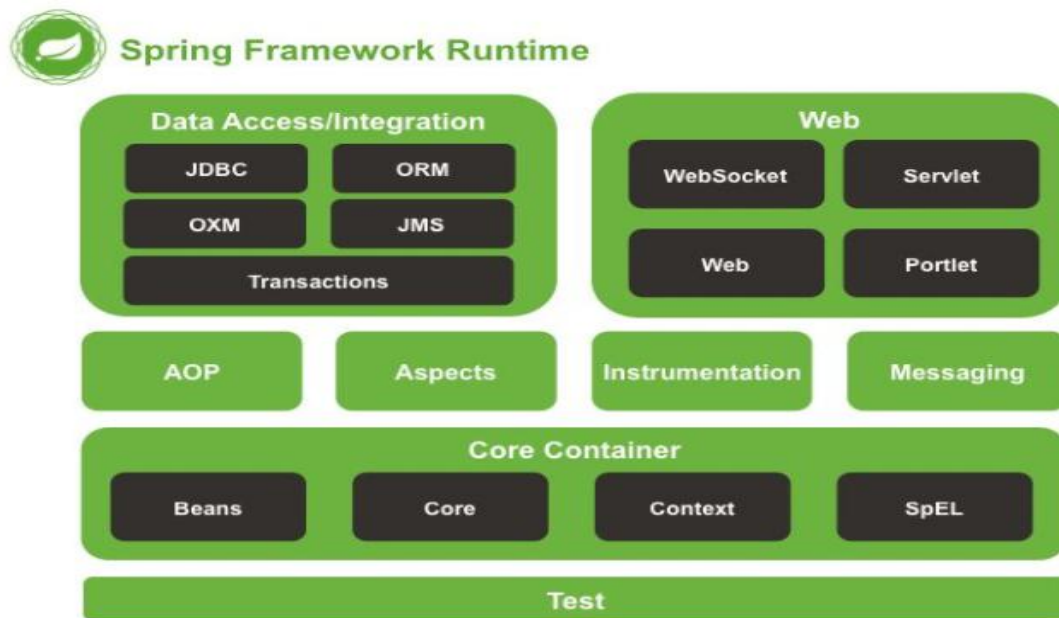
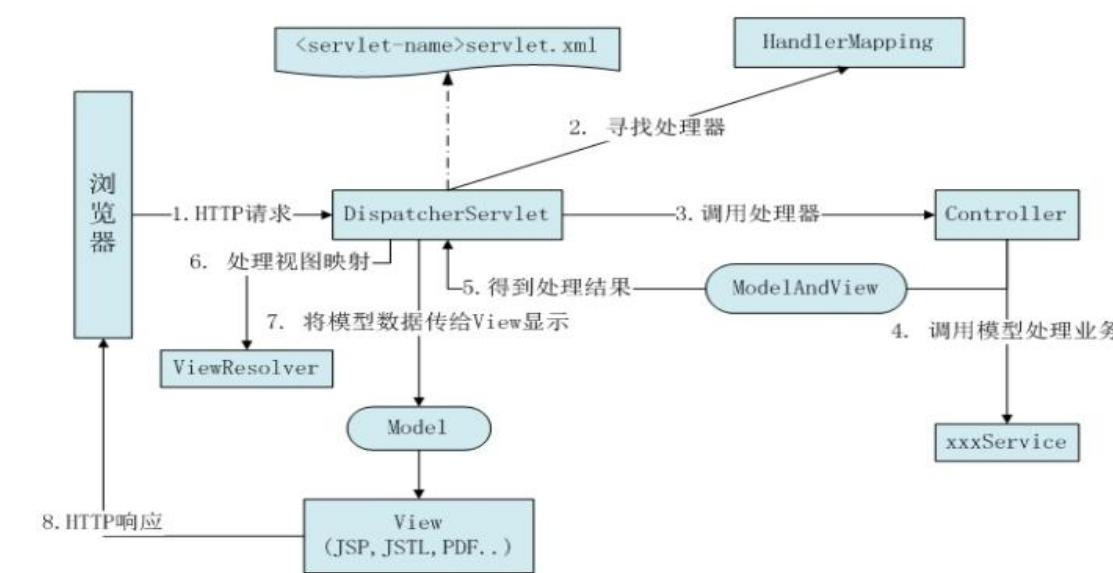


图 2.1Spring 概览图

2.2.2 SpringMVC

SpringMVC 主要在控制层起到接收参数，调用 Service 层的作用。SpringMVC 框架的工作原理如下图 2.2 所示。

图 2.2SpringMVC 工作原理



2.2.3 MyBatis

MyBatis 是一款非常优秀的 ORM 框架，我们使用传统的 JDBC 编程，其虽然提供了连接数据库的功能，但是有着非常严重的缺陷。最大的弊端就是工作量相对较大，我们首先需要连接数据库，接着再去处理 JDBC 底层的事务，同时还要处理数据类型，在处理以上任务的同时还需要对可能产生的异常进行捕捉处理，最后我们还要确保正确的关闭资源的连接。MyBatis 的功能架构如下图 2.3 所示。

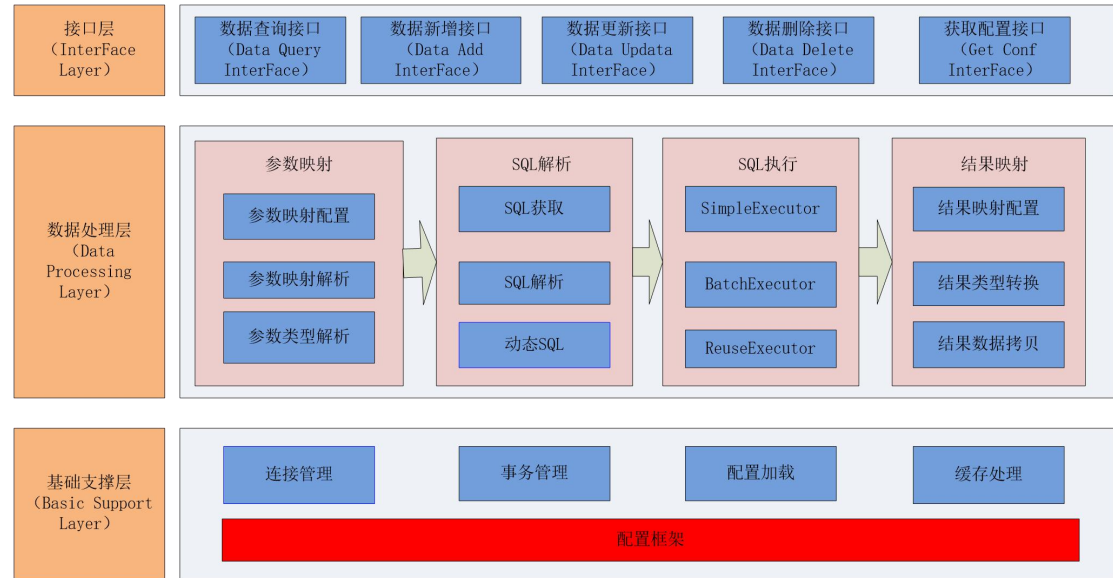


图 2.3 MyBatis 功能架构

2.2.4 Shiro

Shiro 是开源组织 Apache 下的一款进行安全管理的且功能非常强大的框架。其以强盛、灵活的特性保持着鲜活的生命力。在安全管理领域中，软件的实现往往会涉及到非常晦涩难懂的概念，导致整个软件的实现会特别复杂。这对于开发人员而言是一件很痛苦的事情。以下是 Apache Shiro 可以实现的功能：

- 1)：鉴别用户身份信息
- 2)：管理用户权限
- 3)：在没有 web 或 EJB 容器情况下仍可以使用 session API
- 4)：鉴别用户身份、进行权限管理或 session 生命周期时进行相应操作
- 5)：聚合一个或多个用户权限数据源并且以用户视图的形式统一表现出来
- 6)：实现了单点登录功能（SSO）

..... 下图 2.4 展示了 Shiro 的整体架构

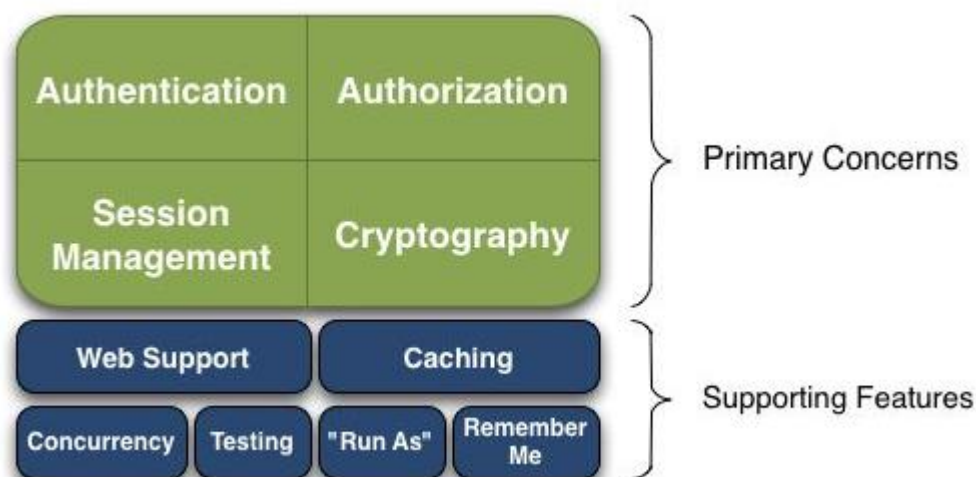


图 2.4 Shiro 整体架构图

2.3 数据库技术

本系统数据库系统采用了 MySQL 和 Redis 相结合的方案。接下来将详细介绍两种数据库系统。

2.3.1 MySQL

MySQL 使用的版本是 5.7，此版本相比较于之前的版本性能上有很大的提升，并且新增了众多全新的特性，例如：支持 JSON、GIS 地理空间数据库、更加便携的 SSL 安全访问配置。在本系统中使用 MySQL 作为系统数据的主存储系统，数据会存储到磁盘上，作为永久数据保存。

2.3.2 Redis

Redis 是一种非常盛行的非关系型数据库，她是基于特殊的结构，并将数据存储到内存的数据库。从性能看两者，Redis 数据库要优于 MySQL 数据库，从安全性上而言 MySQL 数据库要优于 Redis 数据库，Redis 和 MySQL 数据库一起使用，需要性能的地方使用 Redis，在不需要高性能的地方使用 MySQL 达到性能和安全性的双保证。

2.4 本章小结

本章从前端到后端框架以及数据库系统都比较详细的介绍。

第三章 需求分析

系统分析阶段是软件开发工程生命周期里的第一个环节,所以说系统分析阶段要解决的问题是我们的软件系统需要做什么的问题。本章将会从系统的需求说明、资源说明以及可行性分析展开系统分析的研究。

3.1 需求分析

需求分析阶段需要确立本系统需要实现哪些功能。如下图 3.1 简单描述了需求分析

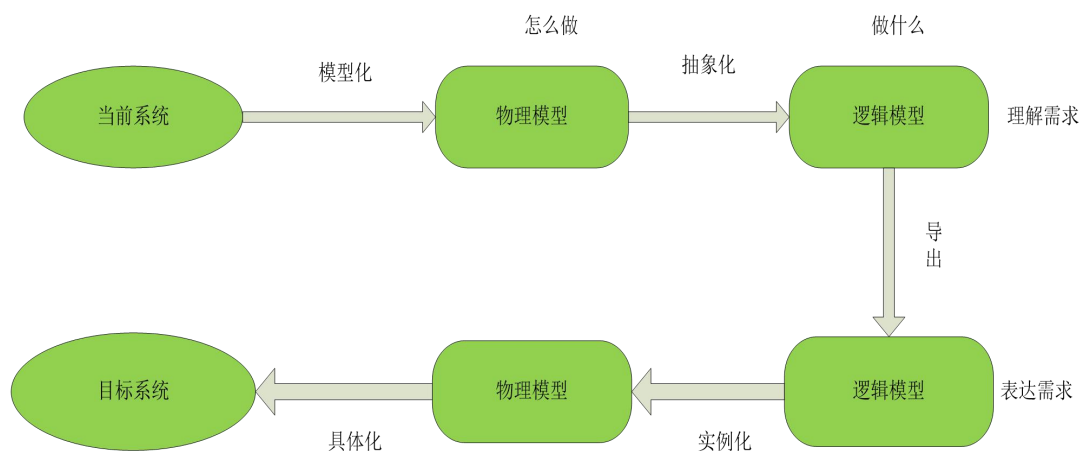


图 3.1 需求分析

3.1.1 功能性需求分析

想要开发一个面向学院的教务管理系统,首先要确定其功能性需求都有哪些。

用户首先需要管理课程库里的课程信息。课程库初始化的时候是没有课程信息的,用户需要手动添加课程或者通过导入表格的形式为课程库批量添加课程。用户增加课程信息的方式也可以选择复制性新增的方式进行,以此达到快速增加一门课程的目的。每一门课程都会有课程相关资料需要维护,用户可以选择以上传附件的形式为每门课程上传相应的课程相关资料。当用户需要导出课程库里的课程信息时也可选择以导出 Excel 表格文件的形式导出课程库里的所有课程信息。

当用户维护好课程信息后就可以对培养方案进行管理。系统初始化的时候是默认没有培养方案的存在。此时用户需要手动在本系统中添加培养方案信息,添加完培养方案基本信息后要求可以对已存在的培养方案信息进行操作功能。用

户在添加培养方案的同时需要为每一个培养方案设定所对应的能力等级，每个培养方案可以有多个对应的能力等级于等级描述。当为培养方案分配课程时要求课程要以课程类别树的形式去管理维护。每一门课程可以通过在课程类别树上进行的操作。为培养方案添加好课程后可以选中相对应的课程为选中的课程添加相对应的课程能力，此时选择的课程能力即在添加培养方案时维护的培养方案能力信息。

确定好使用本系统的用户需求后，决定了他们将要使用的功能，下面将会对功能性需求进行具体的用例分析。

基于用户的角度出发并结合于系统的整体需求，将本系统分为 6 个模块。本系统的总用例图如下图 3.2 所示。

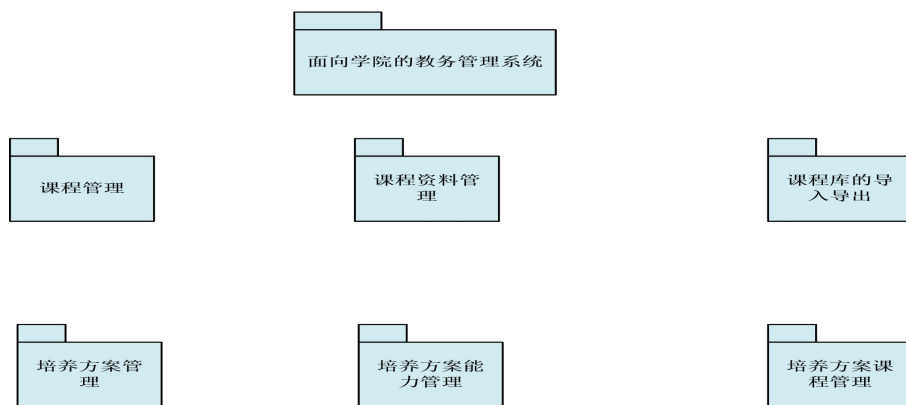


图 3.2 系统总用例图

上图 3.2 描述了系统总用例关系。接下来将会以系统总体用例关系为基础描述各个模块的用例关系。

1) 课程信息管理部分的用例图。

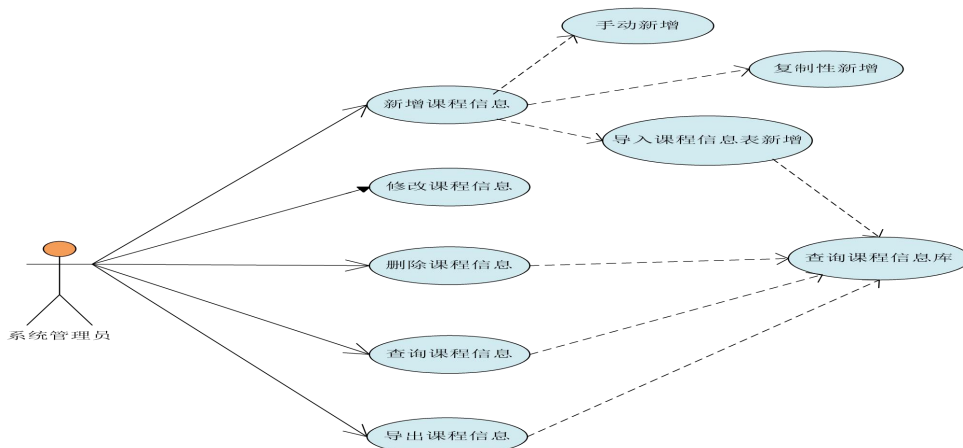


图 3.3 课程信息管理用例图

2) 课程资料管理用例图

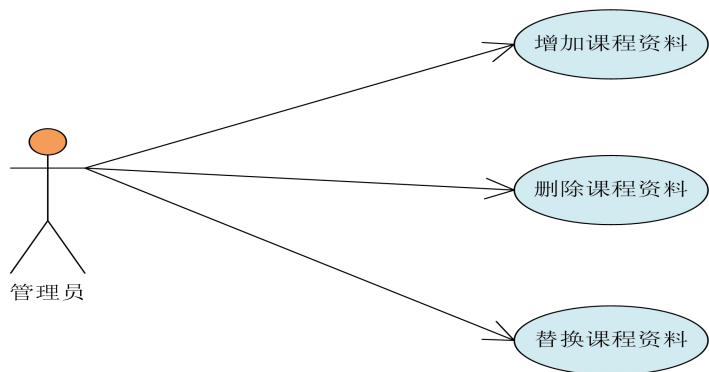


图 3.4 课程资料管理用例图

3) 培养方案管理用例图

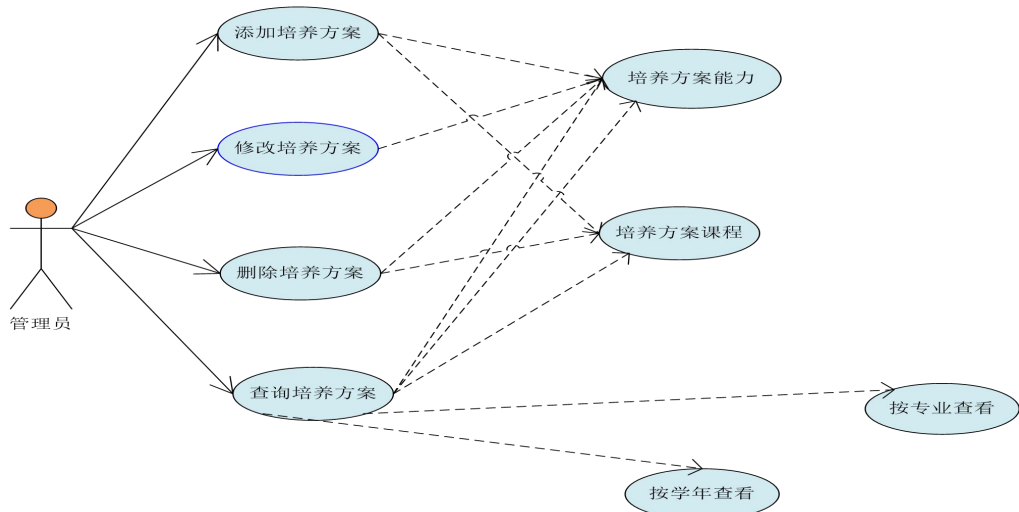


图 3.5 培养方案管理用例图

4) 培养方案课程管理用例图

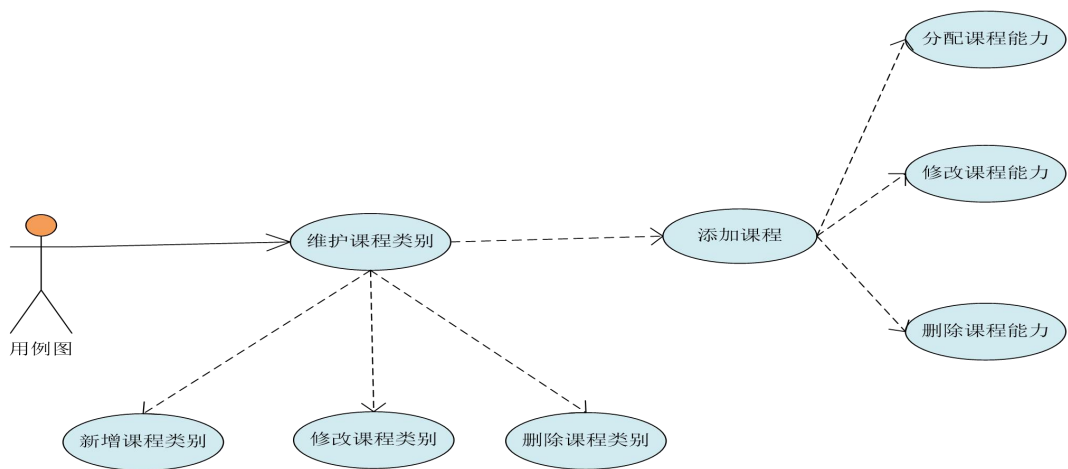


图 3.6 培养方案课程管理用例图

3.1.2 非功能性需求分析

进行完功能性需求分析后，还需要考虑到一些非功能性需求（Non-functional requirement）的因素。

非功能性需求往往会被称作为系统的品质特性，在有的时候也通常会被称作“限制”或“品质服务需求”，非功能性需求有的时候也会被称作英文“ilities”。

下述将会从 7 个方面对非功能性需求进行分类分析：

1) 性能：本系统支持并发的用户数量需要足以支撑实际情况下的最大用户访问数量。

2) 可靠性：存储数据的方式需要采用基于同享存储的方法。在数据存储的底层原理上采用了事务回滚的机制，保证了事务的一致性。

3) 可维护性：系统在没有人为干预的情况下如果出现了系统错误的情况可以做到自己记录下错误日志的功能。系统操作人员对系统的每一个操作按钮都会在系统中详细记录下来，做到为系统的维护提供了最基础的日志信息。

4) 安全性：需要加强系统的可破解性，要做到控制各种未被授权用户的操作行为。针对系统用户，不同级别的用户要做到控制不同的权限处理，大到控制操作每个功能模块，小到控制到每个按钮和输入表单的权限。

5) 易用性：系统要求操作简单明了，用户界面设计要符合最基本的人机交互原理。要符合大多数人的审美观点，不可只把功能实现，用户设计却做的不伦不类。要做到用户所看到的界面就可明白这些界面将要实现什么样的功能。

6) 数据一致性：系统要保证当某个数据在一处发生变动时，整个系统涉及到变动的部分都要发生相应的变化。

7) 系统条件及限制：要求软硬件环境足以支持系统的运行。

3.1.3 建立需求模型

本系统的总需求模型如下图 3.7 所示

	Title ID	Full Description	Code	Priority
1	1.	课程管理 此模块包括了课程信息新增、复制性新增课程信息、导入课程表格新增课程信息、修改课程信息、删除课程信息的功能。	CourseManage	1
7	2.	课程资料管理 此模块包括了增加课程资料、删除课程资料、替换课程资料的功能。	CourseMaterialManage	1.5
12	3.	课程库课程导入导出 此模块包括了导入课程信息、导出课程信息的功能。	ImportExport	2
→	4.	培养方案管理 此模块包括了添加培养方案、修改培养方案、删除培养方案、查询培养方案信息、添加培养方案能力的功能。	TrainingSchemeManage	3
21	5.	培养方案能力管理 此模块包括了新增培养方案能力、删除培养方案能力的功能。	TrainingSchemePower	4
24	6.	培养方案课程管理 此模块包括了维护培养方案课程树信息、新增培养方案课程、删除培养方案课程信息、为培养方案课程添加课程能力、修改培养方案课程的课程相关能力的功能。	TrainingSchemeCourse	5

图 3.7 总需求模型

课程管理模块的需求模型如下图 3.8 所示。

	Title ID	Full Description	Code	Priority
→	1.	课程管理 此模块包括了课程信息新增、复制性新增课程信息、导入课程表格新增课程信息、修改课程信息、删除课程信息的功能。	CourseManage	1
2	1.1	课程信息新增 新增课程信息	CourseManage_1	Undefined
3	1.2	复制性新增课程信息 选中一门已存在的课程，以这门课程信息为基础，在此信息基础上做微小修改，从而达到快速新增一门课程信息的功能。	CourseManage_2	Undefined
4	1.3	导入课程表格新增课程信息 通过导入已有的课程信息表格来向课程库批量增加多门课程信息。	CourseManage_3	Undefined
5	1.4	修改课程信息 修改课程基本信息。	CourseManage_4	Undefined
6	1.5	删除课程信息 删除一门课程相关信息。	CourseManage_5	Undefined

图 3.8 课程管理需求模型

课程资料管理模块的需求模型如下图 3.9 所示。

7	2.	课程资料管理 此模块包括了增加课程资料、删除课程资料、替换课程资料的功能。	CourseMaterialManage	1.5
8	2.1	增加课程资料 为一门课程增加相关的课程资料，这些课程资料的维护以上传附件的形式体现，如果上传的word文档则需要后台转换为PDF格式的文档形式，为了便于浏览。	CourseMaterialManage_1	Undefined
9	2.2	删除课程资料 删除课程相关附件资料	CourseMaterialManage_2	Undefined
10	2.3	替换课程资料 替换课程相关资料	CourseMaterialManage_3	Undefined
11	2.4	课程资料查看 查看课程相关资料，调用浏览器内置的PDF解析器浏览文档	CourseMaterialManage_4	1.5

图 3.9 课程资料管理需求模型

课程库导入导出模块的需求模型如下图 3.10 所示。

12	3.	课程库课程导入导出 此模块包括了导入课程信息、导出课程信息的功能。	ImportExport	2
13	3.1	导入课程信息 通过导入课程信息的表格达到快速向课程库增加课程信息	ImportExport_1	Undefined
14	3.2	导出课程信息 选择导出当前课程库中的所有课程信息	ImportExport_2	Undefined

图 3.10 课程库导入导出需求模型

培养方案管理模块的需求模型如下图 3.11 所示。

15	4.	培养方案管理 此模块包括了添加培养方案、修改培养方案、删除培养方案、查询培养方案信息、添加培养方案能力的功能。	TrainingSchemeManage	3
16	4.1	添加培养方案 添加培养方案的基本信息	TrainingSchemeManage_1	Undefined
17	4.2	修改培养方案 修改培养方案的基本信息	TrainingSchemeManage_2	Undefined
18	4.3	删除培养方案 删除一个培养方案	TrainingSchemeManage_3	Undefined
19	4.4	查询培养方案信息 通过条件的组合查询培养方案信息	TrainingSchemeManage_4	Undefined
20	4.5	添加培养方案能力 在添加培养方案的时候为培养方案添加培养方案能力	TrainingSchemeManage_5	Undefined

图 3.11 培养方案管理需求模型

培养方案能力管理模块的需求模型如下图 3.12 所示。

21	5.	培养方案能力管理 此模块包括了新增培养方案能力、删除培养方案能力的功能。	TrainingSchemePower	4
22	5.1	新增培养方案能力 新增培养方案能力	TrainingSchemePower_1	Undefined
23	5.2	删除培养方案能力 删除培养方案能力	TrainingSchemePower_2	Undefined

图 3.12 培养方案能力管理需求模型

培养方案课程管理模块的需求模型如下图 3.12 所示。

24	6.	培养方案课程管理 此模块包括了维护培养方案课程树信息、新增培养方案课程、删除培养方案课程信息、为培养方案课程添加课程能力、修改培养方案课程的课程相关能力的功能。	TrainingSchemeCourse	5
25	6.1	维护培养方案课程树信息 通过课程树来维护培养方案课程信息	TrainingSchemeCourse_1	Undefined
26	6.2	新增培养方案课程 维护好课程树的关系后，选择一种课程类别，为此课程类别增加课程信息	TrainingSchemeCourse_2	Undefined
27	6.3	删除培养方案课程信息 删除培养方案相关课程	TrainingSchemeCourse_3	Undefined
28	6.4	为培养方案课程添加课程能力 选中一门课程后，为此课程添加相对应的课程能力。	TrainingSchemeCourse_4	Undefined
→	6.5	修改培养方案课程的课程相关能力 修改培养方案课程的课程相关能力	TrainingSchemeCourse_5	Undefined

图 3.12 培养方案课程管理需求模型

3.1.4 编写数据字典

编写数据字典可以使数据库表结构和程序在结构条理上更加清晰、更加容易理解。使系统在可开发性、可维护性、可拓展性以及系统的强壮性上都有更大的优势。

1) 课程基本信息表的数据字典如下表 3-1 所示。

表 3-1 课程基本信息数据字典

名称：课程基本信息表
说明：描述了课程的基本信息
结构：课程 ID+课程编号+课程平台+课程性质+课程中文名称+课程英文名称+学分+学时+讲课时常+上机时常+实验时常+实践时常+周学时分配+记分方式+学时单位
总编号：001
数据流向：有关数据、数据存储、课程基本信息表、培养方案课程表、课程教学资料表、教材订购表

2) 课程教学资料表的数据字典如下表 3-2 所示。

表 3-2 课程教学资料数据字典

名称：课程教学资料表
说明：描述了与课程教资料的基本信息
结构：资料 ID+课程 ID+资料原来名称+资料系统生成名称+资料类别
总编号：002
数据流向：有关数据、数据存储、课程教学资料表、课程基本信息表

3) 课程类别表的数据字典如下表 3-3 所示。

表 3-3 课程类别数据字典

名称：课程类别表
说明：描述了与课程类别相关的基本信息
结构：类别 ID+类别编号+上级 ID+类别类别
总编号：003
数据流向：有关数据、数据存储、课程类别表、课程基本信息表

4) 课程能力要求对应关系表的数据字典如下表 3-4 所示。

表 3-4 课程能力要求对应关系数据字典

名称：课程能力要求对应关系表
说明：描述了与课程能力要求对应关系相关的基本信息
结构：课程能力要求编号+培养方案编号+课程 ID+能力编号
总编号：004
数据流向：有关数据、数据存储、课程能力要求对应关系表、课程基本信息表

5) 专业培养方案基本信息表的数据字典如下表 3-5 所示。

表 3-5 专业培养方案基本信息数据字典

名称：专业培养方案基本信息表
说明：描述了与专业培养方案对应关系相关的基本信息
结构：培养方案名称+培养方案编号+专业名称+专业代码+修订人+审核人+专业负责人+学科门类+是否使用+培养目标+培养要求
总编号：005
数据流向：有关数据、数据存储、专业培养方案基本信息表、专业基本信息表

6) 培养方案基本信息表的数据字典如下表 3-6 所示。

表 3-6 培养方案基本信息数据字典

名称：培养方案基本信息表
说明：描述了与培养方案相关的基本信息
结构：基本信息编号+培养方案编号+修业年限+授予学位+主干学科+核心课程+专业特色+主要实践性教学环节+主要专业实验
总编号：006
数据流向：有关数据、数据存储、培养方案基本信息表

7) 培养方案图片表的数据字典如下表 3-7 所示。

表 3-7 培养方案图片数据字典

名称：培养方案图片表
说明：描述了培养方案图片信息
结构：图片编号+培养方案编号+图片类型+图片原始名称+图片现在名称+图片上传时间
总编号：007
数据流向：有关数据、数据存储、培养方案图片表

8) 培养方案课程表的数据字典如下表 3-8 所示，其主要描述了与培养方案课程相关的信息。

表 3-8 培养方案课程数据字典

名称：培养方案课程表
说明：描述了培养方案课程信息
结构：培养方案课程编号+培养方案编号+课程 ID+课程类别 ID+课程性质+学期
总编号：008
数据流向：有关数据、数据存储、培养方案课程表、课程基本信息表

3.2 可行性分析

针对于面向教务的管理系统的现行条件进行分析，本小节将会从系统可行性上去分析得到本系统在现有条件下是否有条件去进行开发工作。接下来将会进行详细的可行性分析。

1) 经济可行性

系统开发完成后将会在很大的水平上提高教务管理人员的工作效率，学生也可高效率的完成教学工作任务。本系统的投资回报将会远大于投入支出，所以本系统在经济条件上完全可行。

2) 技术可行性

通过研究对比发现本系统的同类软件在国内的开发并不是特别成熟，且他们做出的产品在功能上都比较单一，无法应对现行教学环境中复杂多变的情形。本系统的开发产品功能能做到功能全面覆盖当前的教学环境的需求因素。所以本系统在开发技术上没有问题，可以正常进行开发。

3) 社会可行性

本系统在开发完成后将会提升整个社会的工作效率，这将会降低社会矛盾，为社会带来隐藏的社会收益。

4) 操作可行性

本系统的人机交互符合绝大多数人的审美观点。所以本系统在操作上具有可行性。

3.3 本章小结

本章从系统的需求分析出发，详细分析了系统的功能性需求与非功能性需求，然后又建立了需求模型与编写数据字典，从数据层对系统进行了分析。最后对系统的可行性进行了分析研究，最终得出了本系统可以进行开发的结论。

第四章 总体设计

基于需求分析阶段的需求描述，本章开始进行系统的总体设计。

4.1 功能模块设计

本系统设计成了六个模块，功能模块图如下图 4.1 所示。

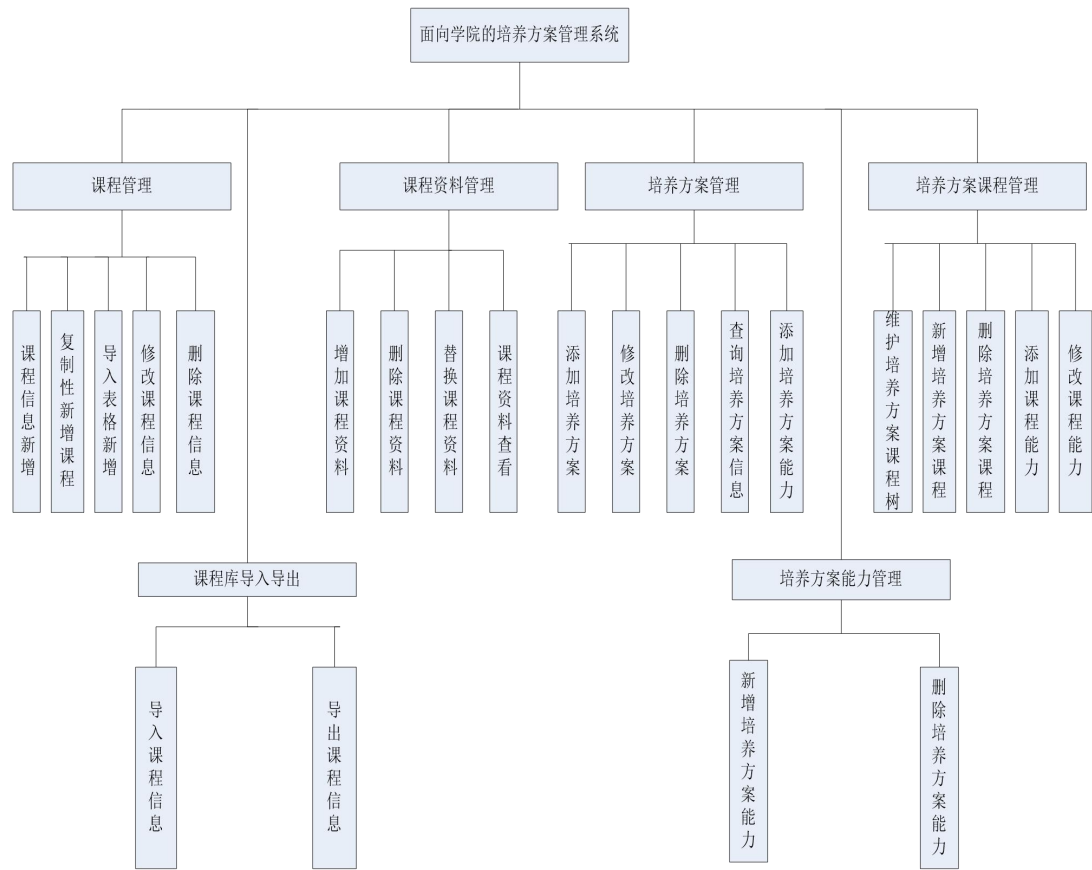


图 4.1 功能模块图

功能模块图从宏观上将系统设计成了六个模块，并且把每个模块的子功能也罗列出来。根据系统的功能模块图可进行各个功能模块以及子功能的简要叙述。接下来将会对系统各个功能模块进行简要的模块功能分析。

表 4-1 课程管理模块

一级功能模块	二级功能模块	功能简要描述
课程管理	课程信息新增	手动新增课程信息
	复制性新增课程	选择一门课程，以此信息为基础达到复制性新增的目的
	导入表格新增课程	通过导入课程信息表格快速为课程库增加课程
	修改课程信息	修改课程基本信息，以及修改课程相关资料
	删除课程信息	删除课程信息

表 4-2 课程资料管理模块

一级功能模块	二级功能模块	功能简要描述
课程资料管理	增加课程资料	为一门课程增加相关的课程资料，以上传附件的形式体现出来
	删除课程资料	删除一门课程的课程相关资料
	替换课程资料	当课程相关资料不适用于当前课程时可以选择替换课程相关资料
	课程资料查看	课程资料上传成功后可选择查看对应的课程相关资料，要求实现方式为直接调用浏览器的 PDF 解析器直接实现

表 4-3 课程库导入导出模块

一级功能模块	二级功能模块	功能简要描述
课程库导入导出	导入	导入课程信息表格。达到快速为课程库添加课程信息的目的。
	导出	导出课程信息可将当前课程库里的课程信息以 Excel 表格的形式导出到用户电脑的文件系统中。

表 4-4 培养方案管理模块

一级功能模块	二级功能模块	功能简要描述
培养方案管理	添加培养方案	添加一个培养方案
	修改培养方案	修改培养方案的信息
	删除培养方案	删除一个培养方案，在删除培养方案信息时需要判断和培养方案关联的部分。
	查询培养方案信息	通过各种组合条件查询培养方案信息
	添加培养方案能力	每个培养方案都有相对应的培养方案能力，添加培养方案的时候应该为此培养方案确定其对应的培养方案能力。

表 4-5 培养方案课程管理模块

一级功能模块	二级功能模块	功能简要描述
培养方案课程管理	课程树维护	每个培养方案需要有其对应的课程信息。管理这些课程信息的方法是通过维护一个培养方案课程相关树的形式，做到课程分类的机制，每门培养方案对应的课程信息都有其对应的课程类别信息。
	新增培养方案课程	通过培养方案课程类别树去选择一个课程类别，在此类别基础上新增培养方案课程
	删除培养方案课程	删除一门培养方案课程，删除课程信息时需要查询培养信息。
	添加培养方案课程能力	选中一门课程，为此课程添加对应的课程能力，可以为一门课程同时添加多个能力。
	修改培养方案课程能力	修改培养方案课程对应的课程能力信息

表 4-6 培养方案能力管理模块

一级功能模块	二级功能模块	功能简要描述
培养方案能力管理	新增培养方案能力	在添加培养方案基本信息时需要为此培养方案新增对应的培养方案能力。
	删除培养方案能力	删除培养方案能力

4.2 数据库设计

数据库系统的角色在系统中承担着至关重要的作用。她为系统中的数据存储提供了一个“数据仓库”。接下来将会比较详细的数据库设计。

4.2.1 概念数据模型设计

概念数据模型的设计是全部数据库设计的重要组成部分，概念数据模型可以起到将需求分析阶段得到的用户需求进一步抽象化，得到信息结构的作用。他可以将显示世界真真实实的模拟出来，为接下来的物理数据模型设计和数据库表结构的设计打下坚实的基础。概念数据模型的设计如下图 4.2 所示。

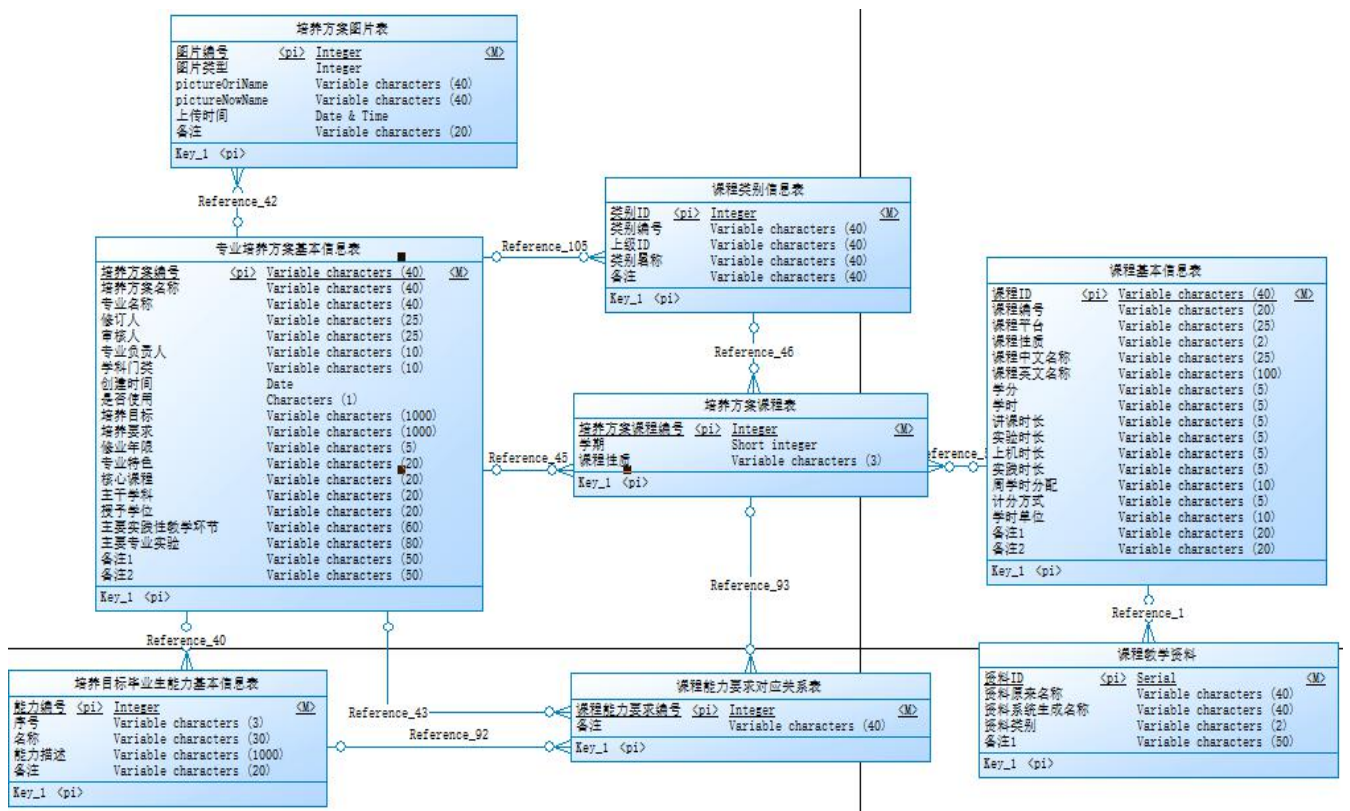


图 4.2 概念数据模型

4.2.2 物理数据模型设计

物理数据模型提供了初始设计阶段需要的基础元素。物理数据模型的设计是基于概念模型的设计展开的，其目的是为了更好的为数据库表结构的设计服务。

课程性质	courseNature	varchar(2)	FALSE	FALSE	必修、选修
课程中文名称	courseNameCN	varchar(25)	FALSE	FALSE	
课程英文名称	courseNameEN	varchar(100)	FALSE	FALSE	
学分	credit	varchar(5)	FALSE	FALSE	
学时	courseHour	varchar(5)	FALSE	FALSE	
讲课时长	teachHour	varchar(5)	FALSE	FALSE	
实践时长	practiceHour	varchar(5)	FALSE	FALSE	
实验时长	experimentHour	varchar(5)	FALSE	FALSE	
上机时长	computerHour	varchar(5)	FALSE	FALSE	
周学时分配	weeklyHour	varchar(10)	FALSE	FALSE	
计分方式	scoringWay	varchar(5)	FALSE	FALSE	
学时单位	courseHourMethod	varchar(10)	FALSE	FALSE	

2) t_teachingProgram_info 是课程教学资料表, 主要用于存储课程教学资料相关信息。

表 4-8 t_teachingProgram_info (课程教学资料表)

中文名称	代码	数据类型	主键	外键	注释
资料 ID	courseFileID	int	TRUE	FALSE	
课程 ID	courseID	varchar(40)	FALSE	TRUE	
资料原来名称	fileOriName	varchar(40)	FALSE	FALSE	
资料系统生成名称	fileNowName	varchar(40)	FALSE	FALSE	
资料类别	Filetype	varchar(2)	FALSE	FALSE	

3) courseTypeInfo 是课程类别信息表, 其主要存储着与课程类别相关的数据信息。

表 4-9 courseTypeInfo (课程类别信息表)

中文名称	代码	数据类型	主键	外键	注释
类别 ID	courseTypeID	int	TRUE	FALSE	

类别编号	typeNum	varchar(40)	FALSE	FALSE	
上级 ID	upTypeNum	varchar(40)	FALSE	FALSE	
类别名称	typeName	varchar(40)	FALSE	FALSE	

4) trainSchemeInfo 是专业培养方案基本信息表，其主要存储着与培养方案基本信息相关的数据信息。

表 4-10 trainSchemeInfo (专业培养方案基本信息表)

中文名称	代码	数据类型	主键	外键	注释
培养方案编号	trainingSchemeID	varchar(40)	TRUE	FALSE	
培养方案名称	trainingSchemaName	varchar(40)	FALSE	FALSE	
专业名称	majorName	varchar(40)	FALSE	FALSE	
专业代码	majorID	varchar(40)	FALSE	TRUE	
修订人	revisePerson	varchar(25)	FALSE	FALSE	
审核人	checkPerson	varchar(25)	FALSE	FALSE	
专业负责人	majorManager	varchar(10)	FALSE	FALSE	
学科门类	majorType	varchar(10)	FALSE	FALSE	
创建时间	createTime	date	FALSE	FALSE	
是否使用	isUse	char(1)	FALSE	FALSE	
培养目标	trainingtarget	varchar(1000)	FALSE	FALSE	
培养要求	trainingRequire	varchar(1000)	FALSE	FALSE	
培养方案编号	trainingSchemeID	varchar(40)	FALSE	TRUE	
修业年限	trainYears	varchar(5)	FALSE	FALSE	
授予学位	trainDegree	varchar(20)	FALSE	FALSE	
主干学科	mainCourse	varchar(20)	FALSE	FALSE	
核心课程	coreCourse	varchar(300)	FALSE	FALSE	
专业特色	majorfeature	varchar(400)	FALSE	FALSE	
主要实践性教学环节	mainTestItem	varchar(500)	FALSE	FALSE	
主要专业实验	mainMajorExperience	varchar(500)	FALSE	FALSE	

5) train_course 是培养方案课程表,其主要存储了与培养方案课程相关的信息。

表 4-11train_course (培养方案课程表)

中文名称	代码	数据类型	主键	外键	注释
培养方案课程编号	trainCourseID	int	TRUE	FALSE	
培养方案编号	trainingSchemeID	varchar(40)	FALSE	TRUE	
课程 ID	courseID	varchar(40)	FALSE	TRUE	
课程类别 ID	courseTypeID	int	FALSE	TRUE	
学期	semester	smallint	FALSE	FALSE	

6) CourseCapacityRequireInfo 是课程能力要求对应关系表,其主要存储了培养方案课程与课程要求能力的关系。

表 4-12CourseCapacityRequireInfo (课程能力要求对应关系表)

中文名称	代码	数据类型	主键	外键	注释
课程能力要求编号	courseCapacityId	int	TRUE	FALSE	
培养方案编号	trainingSchemeID	varchar(40)	FALSE	TRUE	
课程 ID	courseID	varchar(40)	FALSE	TRUE	
能力编号	capacityId	int	FALSE	TRUE	
课程能力要求编号	courseCapacityId	int	TRUE	FALSE	

7) trainingCapacityBaseInfo 是培养目标毕业生能力基本信息表,其主要存储了培养方案与培养方案能力之间的关系。

表 4-13trainingCapacityBaseInfo (培养目标毕业生能力基本信息表)

中文名称	代码	数据类型	主键	外键	注释
能力编号	capacityId	int	TRUE	FALSE	
培养方案编号	trainingSchemeID	varchar(40)	FALSE	TRUE	
序号	capacitySort	varchar(3)	FALSE	FALSE	
名称	capacityName	varchar(30)	FALSE	FALSE	
能力描述	capacityDescription	varchar(1000)	FALSE	FALSE	

8) trainPicture 是培养方案图片表,其主要存储着与培养方案图片相关的数据信息.

表 4-14trainPicture (培养方案图片表)

中文名称	代码	数据类型	主键	外键	注释
图片编号	pictureId	int	TRUE	FALSE	
培养方案编号	trainingSchemeID	varchar(40)	FALSE	TRUE	
图片类型	pictureType	int	FALSE	FALSE	
图片原始名称	pictureOriName	varchar(40)	FALSE	FALSE	
图片现在名称	pictureNowName	varchar(40)	FALSE	FALSE	
上传时间	upTime	datetime	FALSE	FALSE	

4.3 物理配置方案设计

通过系统的总体设计的目标,需要为系统构建出足以支持系统运行的硬件环境.新环境的构建需要满足系统总体设计的各个功能模块与模块设计的功能需求和技术需求.物理配置方案的设计主要考虑系统运行的硬件平台、软件平台.本系统物理配置方案的设计考虑到了以下六项原则依据:

- 1) 系统吞吐量: 系统要求吞吐量的设计足以支持当前的教学管理环境
- 2) 系统响应时间: 系统的系统响应时间要充分符合用户的心理反应时间。
- 3) 系统可靠性: 可靠性要求系统长时间运行不崩溃。
- 4) 系统的处理方式: 系统的处理方式要求以分布式的处理方式。
- 5) 地域范围: 本系统要求采用广域网布局的方式。
- 6) 数据管理方式: 数据管理方式以数据库管理方式进行。

依据以上六项设计原则, 接下来将会从计算机硬件选择、计算机软件选择、等方面具体阐述物理配置方案的设计。

4.3.1 计算机硬件选择

本系统在计算机硬件上采用了主机-终端系统。

内存	硬盘	处理器	显卡
16G	4T	Core i7	GeForce GTX970

4.3.2 计算机软件选择

本系统的数据信息以数据库系统为基础,在数据库管理系统的选择上选用了

MySQL 数据库管理系统。其性能强大、系统的安全保密性高、支持多种系统操作平台、支持多种数据类型，正是基于 MySQL 的以上特性决定使用其作为本系统的数据管理系统。

由于 Windows 操作系统具有强大的软件平台集成能力以及 Windows 操作系统的发展和其向客户机/服务器模式（C/S）和浏览器/服务器模式（B/S）的延申和拓展。故本系统的网络操作系统选择使用 Windows 操作系统。

4.4 系统结构设计

本小结将从前端与后端的结构设计分析系统的结构设计部分。

4.4.1 前端结构设计

本系统前端结构设计如下页图 4.8 所示。整个前端界面代码组织在 webapp 文件夹中。其下的文件结构依次为 css 文件夹，里面存放的是组织系统的样式信息的代码。js 文件夹中存放了系统中的 js 代码。lib 文件夹中存放了一些与前端控件相关的代码，比如有 layui、echarts、bootstrap 等控件的源码。pages 文件夹中存放了各个功能模块的前端详细设计的代码。pages 文件夹中的详细设计如右下图 4.9 所示。我所设计的模块是 courseManage 即课程库管理模块的前端界面的设计代码，trainingScheme 即培养方案管理模块的前端界面的设计设计代码。

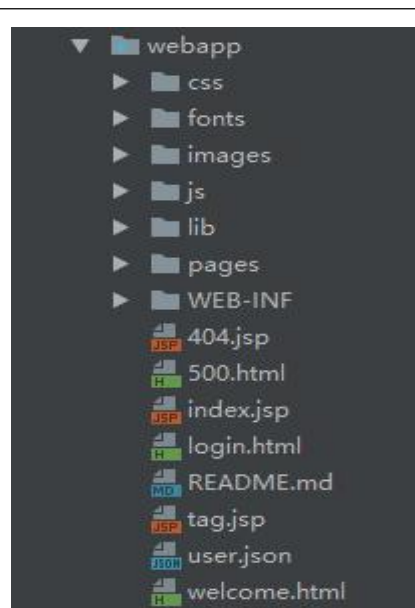


图 4.8 前端结构设计图

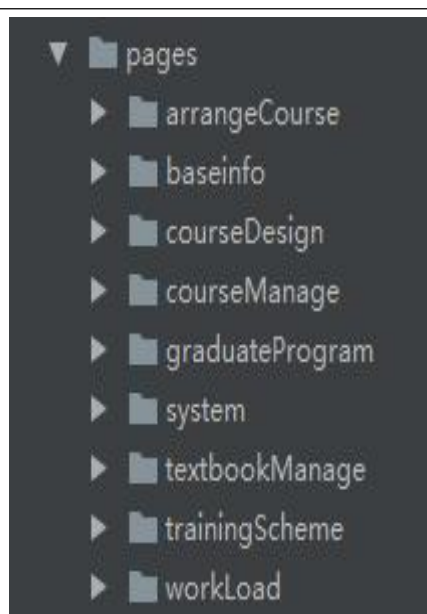


图 4.9 各功能模块前端结构设计图

4.4.2 后端结构设计

本系统的后端结构设计如下图 4.10 所示。resource 包的设计如右下图 4.11 所示，包中存放着系统的各种配置文件以及 mapper 接口的实现，也即 dao 层的代码实现，就是各个功能模块的 sql 语句编写的地方。接下来详细介绍后端结构总体设计

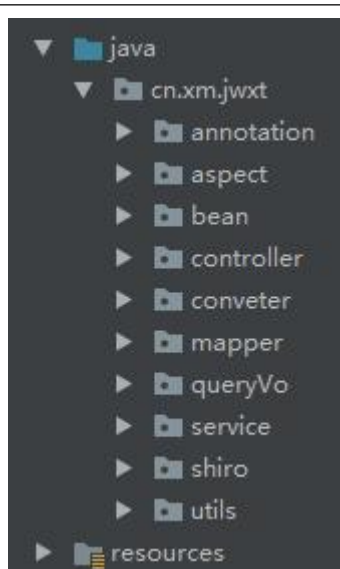


图 4.10 后端结构总体设计

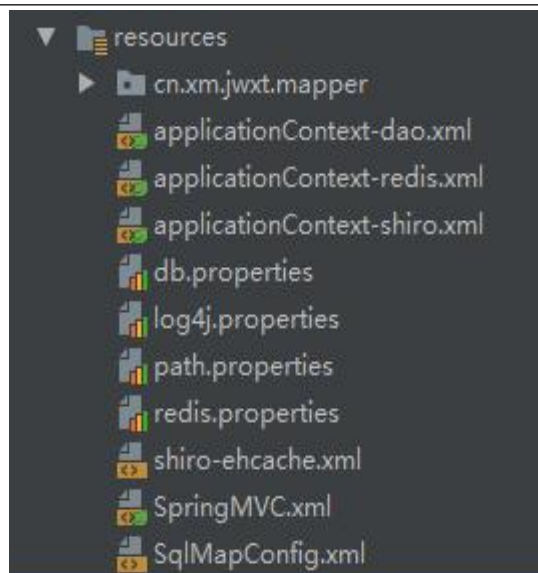


图 4.11 resource 设计

1) annotation 包中主要存放了日志注解的代码，用于日志管理时的注解使用。

2) aspect 包中存放了切面注解的代码。

3) bean 包中存放了系统中各个功能模块设计到的表结构的实体类，实体类中还有 get/set 方法，用于和数据库层面的交流，主要是和 dao 层的交互。

4) controller 包中存放的是系统中控制层相关的代码，主要作用是为了从页面接收参数然后将接收到的参数传递给 service 层让 service 层去进行下一步的处理。

5) conveter 包中转换日期类型的代码，主要用于系统中需要用到处理日期类型的功能。

6) mapper 包中存放的是系统中各个功能模块的 mapper 接口设计。

7) queryVo 包中存放了接收 list 参数的 vo。

8) service 包中存放了系统各个功能模块的 service 层的接口，以及 service 层的接口实现设计的代码。service 层代码主要负责从控制层接收参数，

然后调用 dao 层获取数据后再将数据返回给控制层。

9) shiro 包中存放了安全管理框架相关的代码。

10) utils 包中存放了系统中一些通用的工具类相关的代码设计。

4.5 本章小结

本章从对系统进行了总体上的设计。确定了系统的总体架构。

第五章 详细设计

本章对系统进行详细的设计，主要设计了用户界面与各功能结构的设计。

5.1 界面设计

5.1.1 界面设计原则

用户界面是系统与用户进行交互的第一入口，用户无时无刻不在跟其打交道。用户界面的表现会深刻影响用户的体验好坏。

本系统的用户界面设计遵循了 CRAP 原则，CRAP 原则的具体描述如下

Contrast（对比）：对于两个不同的元素，应设计其表现形式不一。对比有两个目的，一是进一步增强页面的表现效果，而是可以更好的进行页面信息的组织。

Repetition（重复）：相同的元素需要不断的重复，重复的内容可以是某种颜色、空间关系、设计要素、某种格式等内容。

Alignment（对齐）：任何元素都不可随意摆放，而要以一种基准原则进行对齐。

Proximity（亲密性）：将相关的元素尽可能组织在一起，让他们的位置更加靠近。因为在用户看来如果两个元素的位置越接近便会认为他们之间存在着某种联系。遵循亲密性原则会带给用户一种宁静祥和的用户体验。

优秀的用户界面设计通常有以下特征：

- 1) 引导用户视觉
- 2) 配色原理要符合产品功能
- 3) 充分考虑用户场景
- 4) 以用户为中心和出发点

5.1.2 界面设计原型

如下 5.1 所示是课程管理模块的界面原型。

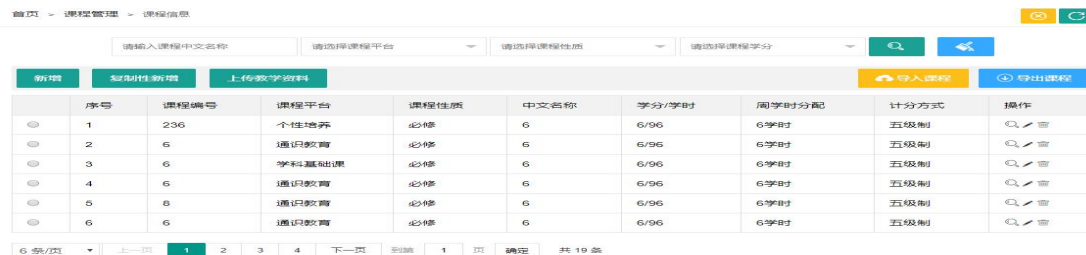


图 5.1 课程管理界面原型

下图 5.2 所示是添加培养方案部分界面原型。

基本信息

培养方案名称

太原科技大学本科专业人才培养方案

必填

专业名称

软件工程

必填

专业代码

必填

学科门类

工学

必填

专业负责人

必填

修订人

必填

修订时间

yyyy-MM-dd

请选择年月

培养目标

培养目标

目录

- 基本信息
- 培养目标
- 培养要求
- 毕业生能力
- 修业年限
- 核心课程
- 专业特色
- 主要实践性教学环节
- 主要专业实验
- 主要课程关系结构图

图 5.2 添加培养方案界面原型

下图 5.3 所示是培养方案管理部分的界面原型图。

培养方案名称

专业名称

专业代码

编辑状态

🔍

🔧

上传课程结构图

安排课程与课程能力

下载培养方案

	序号	培养方案名称	专业名称	专业代码	学科门类	专业负责人	修业年限	修订人	修订时间	状态	操作
●	1	太原科技大学本科专业人才培养方案2	网络工程	1002	工学	张三	4年	修订人	2018-05	保存	🔍🔧🗑
●	2	太原科技大学本科专业人才培养方案1	软件工程	1001	工学	陈立潮	四年。	修订人1	2018-05	保存	🔍🔧🗑

图 5.3 培养方案管理界面原型

本小节展示了部分功能模块的界面设计原型，我们的界面原型设计是基于 ARAP 准则的，所以就不一一展示所有功能模块的界面设计原型。

5.2 功能结构设计

功能结构的详细设计需要对系统中各个功能模块的功能进行详细的设计，再系统设计中属于详细设计阶段，有利于我们梳理各模块的功能。

5.2.1 课程管理

课程管理模块主的功能结构设计如下图 5.4 所示。

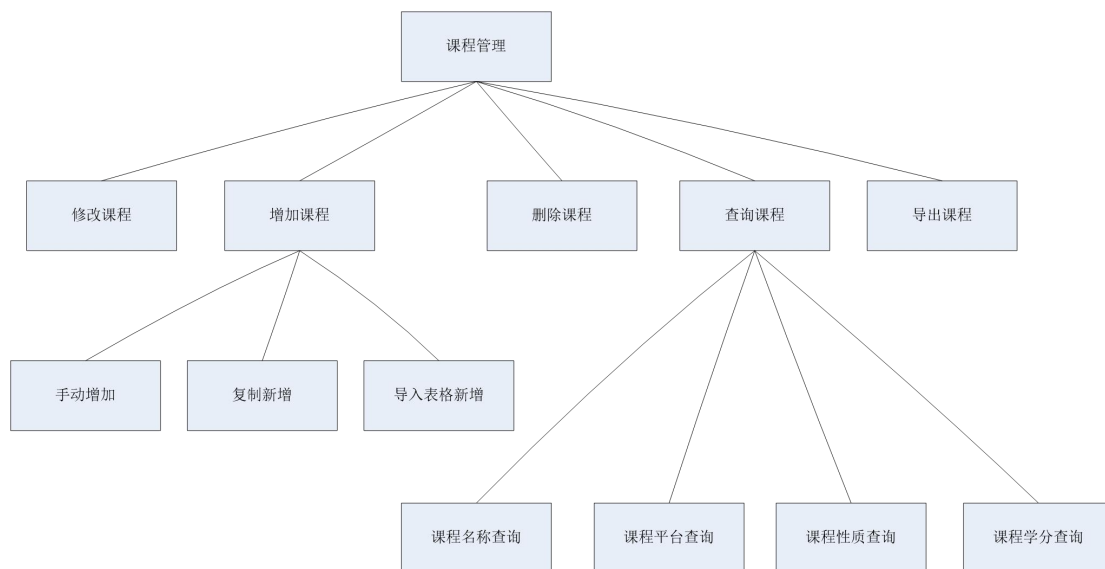


图 5.4 课程管理功能结构设计图

5.2.2 课程资料管理

课程资料管理模块的功能结构设计如下图 5.5 所示。

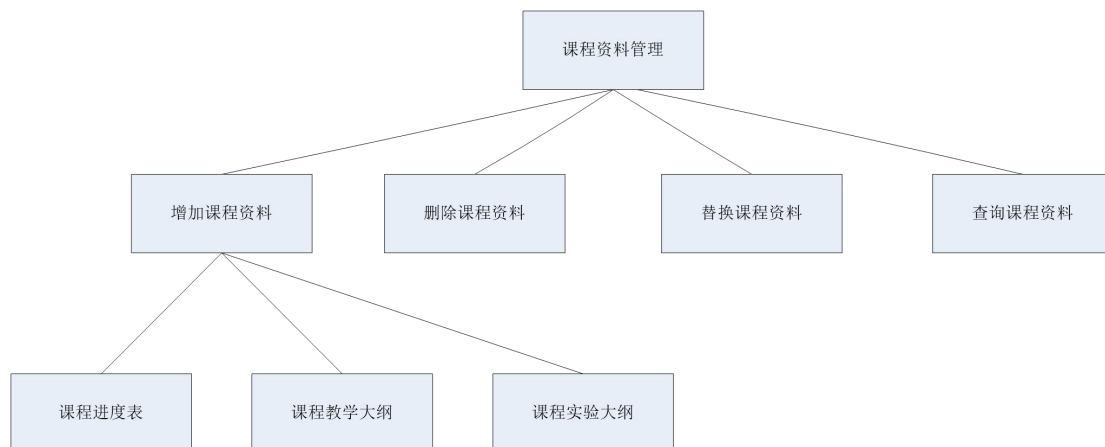


图 5.5 课程资料管理功能结构设计图

5.1.2 课程库导入导出

课程库导入导出的功能结构设计图如下图 5.6 所示

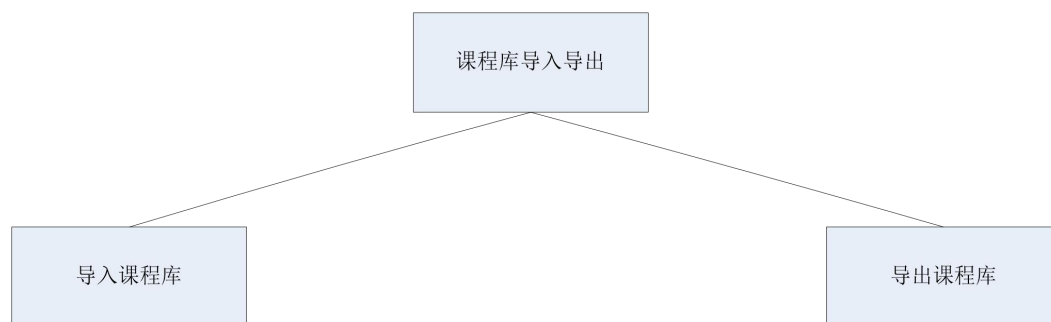


图 5.6 课程库导入导出功能结构设计图

5.1.2 培养方案管理

培养方案管理功能结构设计图如下图 5.7 所示。

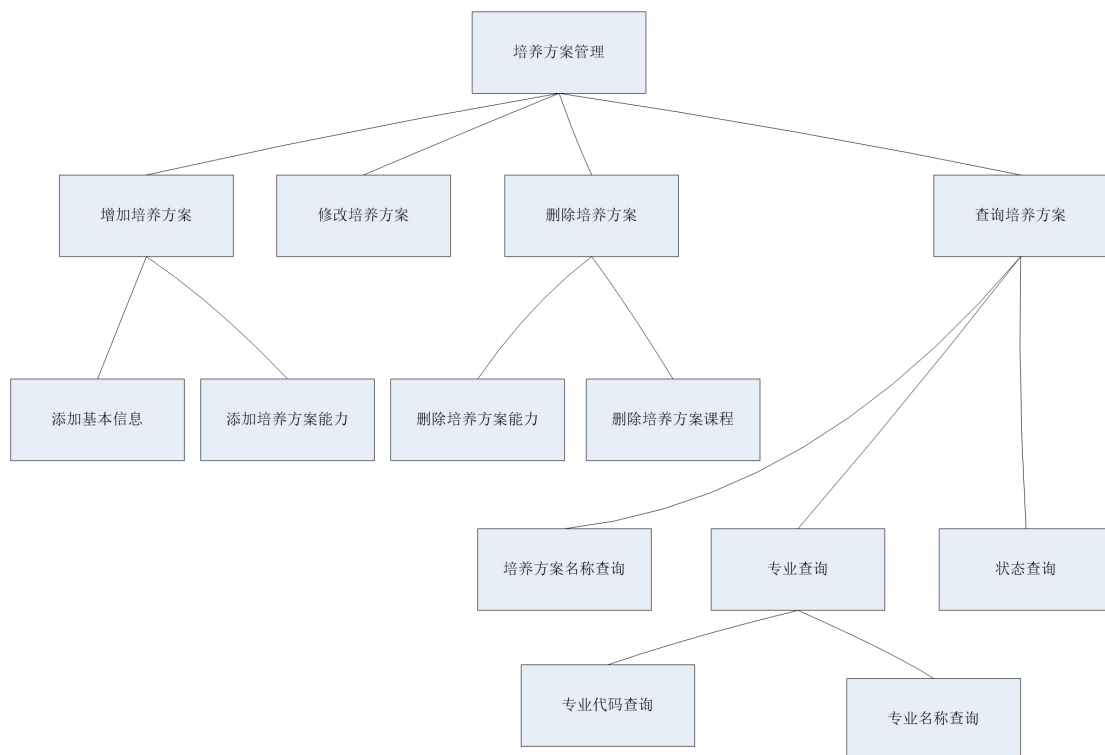


图 5.7 培养方案管理功能结构设计图

5.1.2 培养方案课程管理

培养方案课程管理功能结构设计图如下图 5.8 所示。

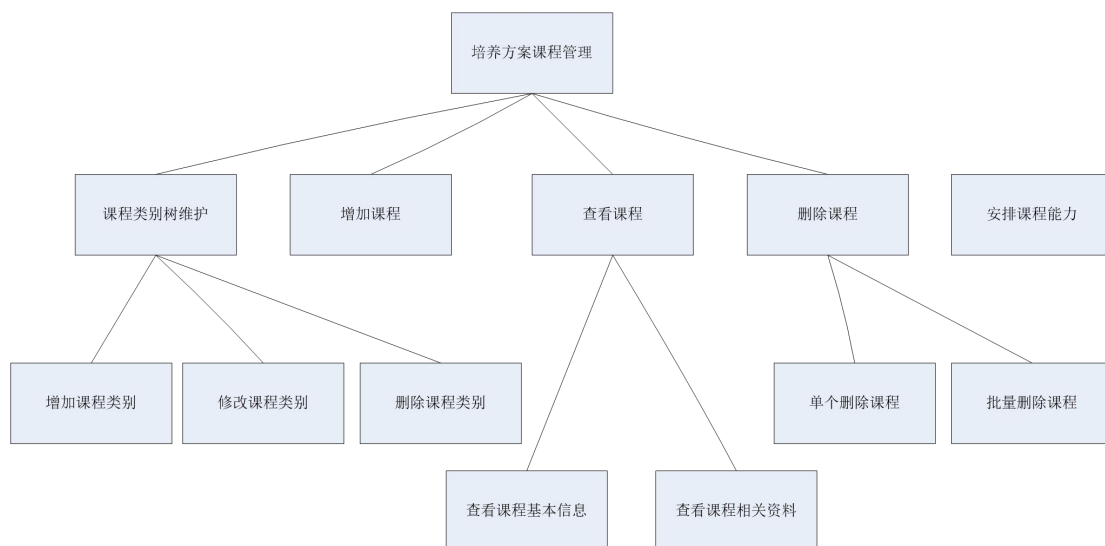


图 5.8 培养方案课程管理功能结构设计图

5.1.2 培养方案能力管理

培养方案能力管理功能结构设计如下图 5.9 所示。

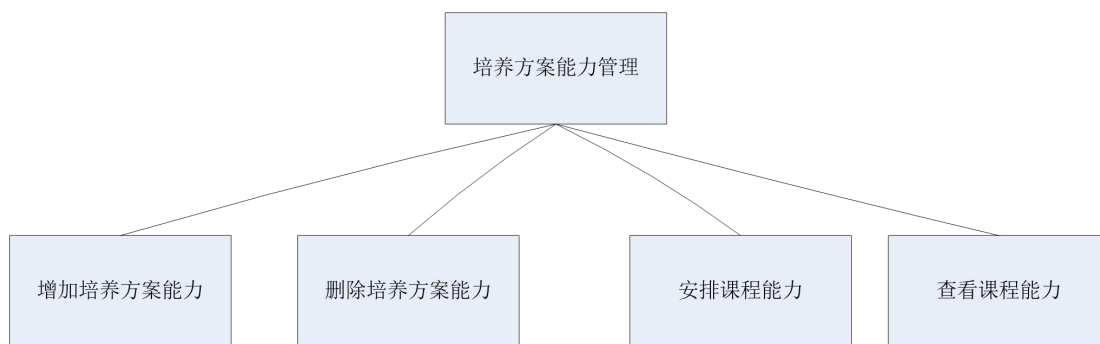


图 5.9 培养方案能力管理功能结构设计图

5.3 本章小结

本章通过两部分的详细设计，进一步加深了对于系统的需求理解，为系统的编码实现打下了坚实的基础。

第六章 编码实现

本章将会面对各个功能模块介绍系统的编码实现部分。其中编码部分每个模块只介绍部分核心功能编码设计。

6.1 课程管理

本节将会简要介绍课程管理部分的页面设计与核心代码实现部分。课程管理部分的界面设计如下图 6.1 所示，课程资料管理与课程库导入导出的入口也在其中，符合充分以用户为中心和出发点的特征。



图 6.1 课程管理界面设计

6.1.1 新增课程

新增课程界面设计如下图 6.2 所示。

新增课程

课程编号

*必填填写

课程平台

请选择课程平台

*必填填写

课程名称

*必填填写

英文名称

*必填填写

课程性质

必修

*必填填写

学分

*必填填写

学时

*必填填写

讲课时长

*必填填写

实验时长

*必填填写

图 6.2 新增课程界面设计

1) JS 代码：课程信息提交采用 ajax 方式提交。提交的时候需要调用方法

验证所提交的课程的课程编号在数据库中是否已存在。

```
layui.use(['form', 'layer'], function () {
    $ = layui.jquery;
    var form = layui.form, layer = layui.layer; //获取所需要的模块

    //监听提交
    form.on('submit(addCourse)', function (data) {
        if(!checkoutCourseNum()){
            //layui 的弹出层提示消息
            layer.msg("课程编号已经存在，请勿重复添加!");
            return false;
        }
        //第一种($.ajax 提交)
        $.ajax({
            url: contextPath+"/courseLibrary/addCourse.do",
            data:data.field,
            type:"POST",
            async:false,
            dataType:'text',
            success:function (response) {
                alert(response);
                if(response == "添加成功"){
                    window.parent.location.reload(); //刷新父窗口
                    var index = parent.layer.getFrameIndex(window.name);
                    parent.layer.close(index); //再执行关闭
                }
            },
        });
    });
});

function checkoutCourseNum() {
    var value = $("[name='coursenum']").val();
    var result = true;
    $.ajax({url:contextPath+"/courseLibrary/getCountByCourseNum.do",
        async:false,
        type:"POST",
        data:{"courseNum":value},
        success:function (response) {
            if(parseInt(response)> 0){
                result=false;
            }
        },
        dataType:'text'})

    return result;
}
```


2) Controller 代码

```

@RequestMapping("/addCourse")
public @ResponseBody
String addCourseBaseInfo(TCourseBaseInfo tCourseBaseInfo) {
    if(tCourseBaseInfo == null) {
        return "添加失败!";
    }
    try {
        courseBaseInfoService.addCourseBaseInfo(tCourseBaseInfo);
    } catch (SQLException e) {
        logger.error("添加课程失败", e);
        return "添加失败!";
    }
    return "添加成功";
}

@RequestMapping("/getCountByCourseNum")
public @ResponseBody String getCountByCourseNum(String courseNum) {
    int result = 0;
    if(ValidateCheck.isNull(courseNum)) {
        result=1;
    }
    try {
        result = courseBaseInfoService.getCountByCourseNum(courseNum);
    } catch (SQLException e) {
        logger.error("根据课程编号查询课程总数出错", e);
    }
    return String.valueOf(result);
}

```

3) Service 层代码

```

@CacheEvict(value = "coursesFy",allEntries =true )//清掉分页的 redis 缓存
@Override
public boolean addCourseBaseInfo(TCourseBaseInfo courseBaseInfo) throws
SQLException {
    if(ValidateCheck.isNull(courseBaseInfo.getCourseid())) {
        courseBaseInfo.setCourseid(UUIDUtil.getUUID2());
    }
    if(ValidateCheck.isNull(courseBaseInfo.getRemark1())) {
        courseBaseInfo.setRemark1(DefaultValue.IS_USE);
    }
    return tCourseBaseInfoMapper.insert(courseBaseInfo)>0?true:false;
}

@Override

```

```

public List<String> addCourseBaseInfoBatch(List<TCourseBaseInfo>
courseBaseInfos) throws SQLException {
    List<String> repeatCourseNums = new ArrayList<String>();
    for(TCourseBaseInfo tCourseBaseInfo :courseBaseInfos){
        if(ValidateCheck.isNull(tCourseBaseInfo.getCoursenum())){
            continue;
        }

        int result =
this.getCountByCourseNum(tCourseBaseInfo.getCoursenum());
        if(result >=
1){
            repeatCourseNums.add(tCourseBaseInfo.getCoursenum());
        }else {//不存在就可以添加
            this.addCourseBaseInfo(tCourseBaseInfo);
        }
    }
    return repeatCourseNums;
}

```

4) DAO 层代码

```

<insert id="insert"
parameterType="cn.xm.jwxt.bean.trainScheme.TCourseBaseInfo" >
    insert into t_course_base_info (courseId, courseNum, coursePlatform,
        courseNature, courseNameCN, courseNameEN,
        credit, courseHour, teachHour,
        experimentHour, computerHour, practiceHour,
        weeklyHour, scoringWay, courseHourMethod,
        remark1, remark2)
    values ({courseid,jdbcType=VARCHAR}, #{coursenum,jdbcType=VARCHAR},
#{courseplatform,jdbcType=VARCHAR},
        #{coursenature,jdbcType=VARCHAR}, #{coursenamecn,jdbcType=VARCHAR},
#{coursenameen,jdbcType=VARCHAR},
        #{credit,jdbcType=VARCHAR}, #{coursehour,jdbcType=VARCHAR},
#{teachhour,jdbcType=VARCHAR},
        #{experimenthour,jdbcType=VARCHAR}, #{computerhour,jdbcType=VARCHAR},
#{practicehour,jdbcType=VARCHAR},
        #{weeklyhour,jdbcType=VARCHAR}, #{scoringway,jdbcType=VARCHAR},
#{coursehourmethod,jdbcType=VARCHAR},
        #{remark1,jdbcType=VARCHAR}, #{remark2,jdbcType=VARCHAR})
</insert>

```

6.2 课程资料管理

6.2.1 上传课程资料

上传课程资料界面设计如下图 6.3 所示

图 6.3 上传课程资料界面设计

1) JS 代码

```
function initUploadTeachingProgram() {
    layui.use(['upload', 'layer'], function() {
        var upload = layui.upload, layer = layui.layer;
        upload.render({
            elem: '#uploadCourseMaterial',
            url: contextPath + '/teachingFile/upload.do',
            auto: false, //是否自动上传
            accept: 'file', //普通文件,
            multiple: false, //支持多文件上传
            exts: 'doc|docx|pdf', //只允许上传 doc, docx 和 pdf 文件
            bindAction: '#uploadTeachingFile', //绑定到哪个按钮进行提交(提交按钮进行提交)
            data: { //另外携带的数据
                courseid: $('[name=' courseid ']').val(), //携带课程编号
                filetype: $('[name=' filetype ']').val() //携带文件类型
            },
            choose: function (obj, file) { //选中文件执行的操作
                var file = this.files ; //获取到选择的文件
                obj.preview(function(index, file, result){
                    // alert(file.name+file.size)//获取文件的名字和大小
                    $('#uploadCourseMaterial').val(file.name+"("+(file.size/1024)+"kb)")
                });
            },
            before: function (obj) { //文件上传前
                $('#uploadTeachingFile').addClass("layui-btn-disabled");//设置按钮不可以点击(增加 layui 的禁用按钮属性)
                var index = layer.load(); //开始上传之后打开 load 层
                $('#hidden_index').val(index); //隐藏到页面中
```

```

    },
    done: function(res, index,
upload) {
        layer.close(layer.index); //它获取的始终是最新弹出的某
        个层, 值是由 layer 内部动态递增计算的
        if(res !=null && res.success==true) { //上传成功
            layer.msg(res.msg, {time: 1500}, function () { //弹出消息三秒后
                layer.close($("#hidden_index").val());
                var index = parent.layer.getFrameIndex(window.name);
                parent.layer.close(index); //再执行关闭
            });
        }
    }
};

```

2) Controller 代码

```

@RequestMapping("/upload")
public @ResponseBody
ResponseResult uploadTeachingProgramFile(TTeachingprogramInfo
teachingprogramInfo, MultipartFile file) {
    //1. 将文件保存到磁盘
    String fileoriname=null; //原名称
    String filenowname=null; //系统生成的名称
    if(file!=null) {
        fileoriname=file.getOriginalFilename(); //获取文件原始名字
        filenowname = UUIDUtil.getUUID2()+"."+
        FilenameUtils.getExtension(fileoriname); //新生产一个文件
    }
    try {
        filenowname =
        FileHandleUtil.uploadSpringMVCFile(file, "teachingFile", filenowname); //获取传完
        之后新的名字
    } catch (Exception e) {
        logger.error("保存文件出错", e);
    }
    //2. 保存数据库
    teachingprogramInfo.setFilenowname(filenowname); //设置 UUID 生成的名字
    teachingprogramInfo.setFileoriname(fileoriname); //设置原始名字
    String result=null;
    //返回 ajax 请求的工具类
    ResponseResult responseResult = new ResponseResult();
    try {
        result =
        teachingProgramInfoService.addTeachingProgramInfo(teachingprogramInfo)? "上传成
        功": "保存失败";
    } catch (SQLException e) {
        result = "保存失败";
        responseResult.setSuccess(false);
        logger.error("保存课程资料到数据库出错");
    }
}

```

```
}  
responseResult.setMsg(result);  
return responseResult;  
}
```

3) Service 层代码

```
@Override  
public boolean addTeachingProgramInfo(TTeachingprogramInfo teachingprogramInfo)  
throws SQLException {  
    return tTeachingprogramInfoMapper.insert(teachingprogramInfo)>0?true:false;  
}
```

4) DAO 层代码

```
<insert id="insert"  
parameterType="cn.xm.jwxt.bean.trainScheme.TTeachingprogramInfo" >  
    insert into t_teachingprogram_info (courseFileID, courseID, fileOriName,  
        fileNowName, FileType, remark1  
    )  
    values (#{coursefileid,jdbcType=INTEGER}, #{courseid,jdbcType=VARCHAR},  
        #{fileoriname,jdbcType=VARCHAR},  
        #{filenowname,jdbcType=VARCHAR}, #{filetype,jdbcType=VARCHAR},  
        #{remark1,jdbcType=VARCHAR}  
    )  
</insert>
```

6.3 培养方案管理

培养方案管理界面设计如下图 6.4 所示。可组合条件查询培养方案，选择培养方案上传课程结构图，为培养方案安排课程与课程能力。

培养方案名称	专业名称	专业代码	保存	Q	+					
<div>上传课程结构图 安排课程与课程能力 下载培养方案</div>										
序号	培养方案名称	专业名称	专业代码	学科门类	专业负责人	修业年限	修订人	修订时间	状态	操作
1	太原科技大学本科专业人才培养方案2	网络工程	1002	工学	张三	4年	修订人	2018-05	保存	Q / 笔
2	太原科技大学本科专业人才培养方案1	软件工程	1001	工学	陈立潮	四年。	修订人1	2018-05	保存	Q / 笔

6 条/页 上一页 1 下一页 到第 1 页 确定 共 2 条

图 6.4 培养方案管理界面设计

6.3.1 查看培养方案

查看培养方案界面设计如下图 6.5 所示。

11、毕业总学分及总学时基本要求与分配							
课程类别		课程性质	学分	占总学分比例	学时	占总学时比例	
通识课程类别		必修	2.5	15%	140	14%	
		必修	2.5	15%	140	14%	
学科基础课程	数理基础	必修	2.5	15%	140	<div>目录</div> <div><div>• 基本信息</div><div>• 培养目标</div><div>• 培养要求</div><div>• 毕业生能力</div><div>• 修业年限</div><div>• 核心课程</div><div>• 专业特色</div><div>• 主要实践性教学环节</div><div>• 主要专业实验</div><div>• 毕业总学分及总学时基本要求与分配</div><div>• 主要课程关系结构图</div><div>• 毕业总学分及总学时基本要求与分配</div><div>• 毕业总学分及总学时基本要求与分配</div><div>• 每学期教学安排一览表</div><div>• 课程分类及学分比例</div></div>	
	大类基础	必修	2.5	15%	140		
	专业基础	必修	2.5	15%	140		
专业课程		必修	2.5	15%	140		
		选修	2.5	15%	140		
个性培养		选修	2.5	15%	140		
教学环节	通识实践	必修	2.5	15%	140		
	专业实验与专业实践	必修	2.5	15%	140		
毕业总学分（总学时）			2.5	15%	140		
—— 主要课程关系结构图 ——							
12、主要课程关系结构图							

图 6.5 查看培养方案界面设计

1) JS 代码

```
function viewTrainschemeDetail(obj) {
    var tr = $(obj).parent().parent();//获取到 tr 元素
    var value = tr.find("input:hidden").val();//获取到培养方案 ID

    window.open(contextPath+"/forward/2TrainSchemeDatail.do?trainSchemeId="+value);
}
```

2) Controller 层代码

```
@RequestMapping("/2TrainSchemeDatail")
public String forward2TrainSchemeDetail(@RequestParam(defaultValue = "1") String
trainSchemeId, Model model){
    //1. 将数据塞进去
    model.addAttribute("trainSchemeId",trainSchemeId);
    //2. 跳转到查看详情页面
    return "pages/trainingScheme/trainingScheme-view";
}
```

3) Service 层代码

```
public Map<String, Object> getTrainschemeinfoById(String trainschemeinfoId) throws
SQLException;
```

4) DAO 层代码

```
<select id="getTrainschemeinfoById" parameterType="string" resultType="map">
    select * from trainschemeinfo
    <where>
        <if test="1 == 1">
            and isUse = '1'
        </if>
    </where>
</select>
```

```
<if test="1 == 1">
    and trainingSchemeID = #{trainschemeinfoId}
</if>
</where>
</select>
```

6.4 培养方案课程管理

培养方案课程管理部分如下图 6.6 所示，左侧是培养方案课程类别树。本小节编码实现将会对排课部分详细介绍。



图 6.6 培养方案课程管理

6.4.1 培养方案排课

培养方案排课界面设计如下图 6.7 所示，接下来将会进行编码实现的详细介绍。



图 6.7 培养方案排课界面设计

1) JS 代码

```
function queryCourseFY() {
```

```
$.ajax({
    url:contextPath+"/TrainCourse/getCoursesBseInfoByCondition.do",
    data:$("#queryCourseForm").serialize(),
    dataType:'json',
    type:'post',
    async:true,
    success:showCourseInfo
});
}
```

2) Controller 层代码

```
@RequestMapping("/getCoursesBseInfoByCondition")
public PageInfo<TCourseBaseInfo> getCoursesBseInfoByCondition(@RequestParam Map
condition) {
    int pageNum = 1;
    if(ValidateCheck.isNull((String) condition.get("pageNum"))){ //如果不为空
的话改变当前页号
        pageNum = Integer.parseInt((String) condition.get("pageNum"));
    }
    int pageSize = DefaultValue.PAGE_SIZE;
    if(ValidateCheck.isNull((String) condition.get("pageSize"))){ //如果不为
空的话改变当前页大小
        pageSize = Integer.parseInt((String) condition.get("pageSize"));
    }
    //开始分页
    PageHelper.startPage(pageNum, pageSize);
    List<TCourseBaseInfo> coursesBseInfos = null;
    try {
        coursesBseInfos =
trainCourseService.getCoursesBseInfoByCondition(condition);
    } catch (SQLException e) {
        logger.error("查询未排的课程失败", e);
        return null;
    }
    PageInfo<TCourseBaseInfo> pageInfo = new
PageInfo<TCourseBaseInfo>(coursesBseInfos);
    return pageInfo;
}
```

3) Service 层代码

```
@Override
public List<TCourseBaseInfo> getCoursesBseInfoByCondition(Map condition) throws
SQLException {
    return trainCourseCustomMapper.getCoursesBseInfoByCondition(condition);
}
```


4) DAO 层代码

```

<select id="getCoursesBseInfoByCondition" parameterType="map"
resultType="TCourseBaseInfo">
    SELECT * FROM t_course_base_info
    <where>
        <if test="1 == 1">
            and remark1 = '1' AND courseId NOT IN (SELECT courseId FROM train_course
WHERE trainingSchemeID =
            #{trainSchemeId} and courseType='1')
        </if>
        <if test="courseamecn != null and courseamecn != ''">
            and courseamecn like '%${courseamecn}%'
        </if>
        <if test="courseplatform != null and courseplatform != ''">
            and courseplatform = #{courseplatform}
        </if>
        <if test="coursenature != null and coursenature != ''">
            and coursenature = #{coursenature}
        </if>
    </where>
</select>

```

6.5 培养方案能力管理

培养方案能力管理部分以分配课程能力为例进行编码实现, 分配课程能力界面设计如下图 6.8 所示。

安排课程能力信息

确认分配课程能力

序号	课程名称	课程编号	G 1	G 2	G 3	G 4	G 5	G 6	G 7	G 8	G 9	G 10	G 11	G 12
1	6666	6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	6666	6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

图 6.8 分配课程能力界面设计

1) JS 代码

```

function getTrainCourseCapacity() {
    $(".trainCourseCheckbox:checked").each(function (index) { //遍历选中的多选框
        ids[index] = $(this).val();
    });
}

```

```
$.post(contextPath+"/TrainCourseCapacity/getTrainCourseByTrainCourseIds.do", {"ids":ids.toString()},selectTraincapacity,'json');
```

2) Controller 层代码

```
@RequestMapping("/getTrainCourseByTrainCourseIds")
public List<Coursecapacityrequireinfo>
getTrainCourseCapacityByTrainCourseIds(@RequestParam(defaultValue =
"100000,1000000") String ids){
    String[] split = ids.split(",");//逗号分隔形成数组
    List<Integer> ids_int = new ArrayList<>();
    for(String id_str : split){
        ids_int.add(Integer.parseInt(id_str));
    }
    List<Coursecapacityrequireinfo> trainCourseCapacitys = null;
    try {
        trainCourseCapacitys =
courseCapacityRequireInfoService.getTrainCourseCapacityByTrainCourseIds(ids_int
);
    } catch (SQLException e) {
        logger.error("查询培养方案课程信息出错",e);
        return null;
    }
    return trainCourseCapacitys;
}
```

3) Service 层代码

```
@Override
public List<Coursecapacityrequireinfo>
getTrainCourseCapacityByTrainCourseIds(List<Integer> trainCourseIds) throws
SQLException {
    return
coursecapacityrequireinfoCustomMapper.getTrainCourseCapacityByTrainCourseIds(tr
ainCourseIds);
}
```

4) DAO 层设计

```
<select id="getTrainCourseCapacityByTrainCourseIds" parameterType="list"
resultType="cn.xml.jwxt.bean.trainScheme.Coursecapacityrequireinfo">
    SELECT * FROM coursecapacityrequireinfo WHERE trainCourseID IN
    <foreach collection="list" item="id" open="(" close=")" separator=",">
        #{id}
    </foreach>
</select>
```

6.6 本章小结

本章以系统中各个不同的功能模块为例,挑选了各个模块中的不同的子功能进行了系统的编码实现的介绍。由于篇幅有限,本章只对各子功能的核心代码进行了介绍。

第七章 系统测试

本章主要对系统进行黑盒测试。下面将对系统中比较核心的功能进行测试。

7.1 课程新增测试

课程新增测试界面如下图 7.1 所示，当输入完课程信息后点击确认添加，界面提示添加成功。

图 7.1 课程新增测试界面

7.2 导出课程测试

导出课程时先点击右上角导出课程按钮，页面弹出提示框“确认下载课程信息？”，点击确定可得到导出成功的提示信息。

图 7.2 导出课程测试

7.3 上传课程结构图测试

上传课程结构图测试如下图 7.3 所示。点击上传课程关系结构图后选择自己想要上传的图片，接着系统会提示上传成功的信息。

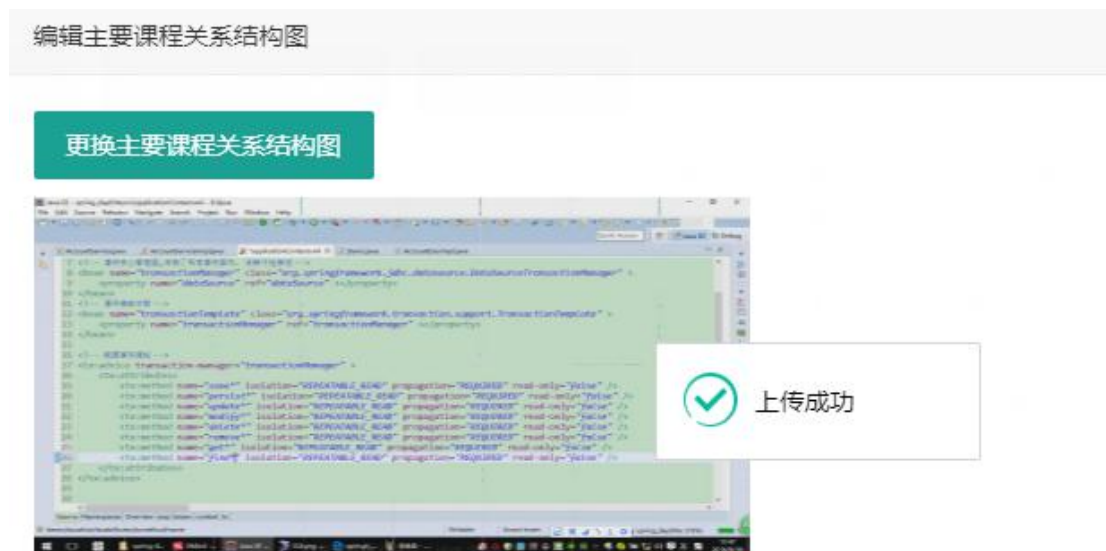


图 7.3 上传课程结构图测试

7.4 课程类别树测试

选择一种课程类别后，点击加号按钮，系统会提示“确认在当前选中的课程类别下添加新的类别”，点击确定后将会进入添加课程类别的界面。如下图 7.4 所示。



图 7.4 添加课程类别测试

当对课程类别树的根节点进行排课时，如果该节点已经有排好的课程，系统会提示“该节点已经排有课程，不允许添加课程类别”信息。



图 7.5 根节点排课测试

7.5 删除培养方案课程测试

删除培养方案时系统界面会提示“确认删除”的提示信息如下图 7.6 所示，点击确定按钮后界面提示删除成功的信息，如下图 7.7 所示。



图 7.6 删除培养方案课程信息



图 7.7 课程删除成功提示

7.6 本章小结

本章对系统中各个不同的模块进行了测试。完成了本系统开发中的重要的一步。

结束语

伴随着本次毕业设计的结束，我四年的大学生涯也接近了尾声。四年的青春有精彩也有遗憾。过去的已经过去他再也不会回来。重点是要抓住眼前的大好时光，过好当前的每一天。

此次毕业设计从烟雨三月开始，结束于纷纷扰扰的六月。三个月以来有过痛苦也有收获。痛苦是短暂的，收获是持久的。这期间所有的经历都会是我今后人生道路上最宝贵的财富。

在我敬爱的谢老师的指导之下，我们一步一个脚印，踏实的过着每一天的生活，尽力的完成每一个任务。还有那些每天一起奋斗一起陪伴的小伙伴们，正是由于我们每天的奋斗才有今天的论文成果。

在此次毕业设计中，我完成了课程信息的管理以及培养方案的管理。功能相对比较完善，还有一些细节问题需要进行下一步的优化。在整个毕业论文的设计工作中，自己也学到了许多有用的知识，对于软件项目的开发有了更加深刻的理解。通过系统的开发与论文的编写工作，自己的专业技能有了进一步的提升，为我进入社会工作积累下了丰富的社会经验。

通过本次设计工作我也明白的“纸上得来终觉浅”的道理。我们应该坚持实践是检验真理的唯一标准，注重实践的价值。

致 谢

毕业即将来临，四年的大学时光伴随着毕业设计的完成步入了尾声，即将面临告别，有欢声笑语也有依依不舍。在太原科技大学的四年学习中，从一开始的懵懂的入学新生，到现在对未来充满憧憬的毕业生，这期间得到身边的很多老师、同学、朋友的帮助以及父母在背后的默默支持，在这里，我真心想向他们致以最诚挚的谢意。

首先，我要对站在我背后的父母说一声谢谢。在整个学习生涯中，是他们的支持与期望，让我有了那份责任感，督促着我不断地学习进步。也正是想不辜负他们的期望，我才能站在这里与来自全国各地的学子共同学习进步。

然后，我要对我的指导老师谢斌红说一声谢谢。在本次毕业设计的系统实现及论文撰写的过程中，他给我提出了很多实用而且有价值的意见，每当我遇到问题和困难时，他总会悉心的对我指导和讲解，对我完成此次设计提供了至关重要的帮助。

最后，我要对我大学四年遇到的所有老师、同学以及朋友说一声谢谢。课堂上，老师的教诲是我一生难忘的，不仅仅是知识，更多的是做人的道理。课堂下，是同学和朋友陪我度过了这酸甜苦辣的大学时光，困难时他们给予我最温暖的安慰，快乐时他们与我共享最美好的时刻，正是因为他们，让我在大学中拥有了最不舍得的东西-友情。

参考文献

- [1] 李兴华. Java 开发实战经典[M]. 北京: 清华大学出版社, 2009, 1-846
- [2] 孙卫琴. Tomcat 与 Java Web 开发技术详解 (第二版) [M]. 北京: 电子工业出版社, 2009, 1-734
- [3] Bruce Eckel. Java 编程思想 (第四版) (陈昊鹏译) [M]. 北京: 机械工业出版社, 2007, 1-880
- [4] 杨开振. 深入浅出 MyBatis 技术原理与实战[M]. 北京: 电子工业出版社, 2016, 1-268
- [5] Budi Kurniawan, PaulDeck. Servlet、JSP 和 Spring MVC 初学指南(林仪明, 俞黎敏译) [M]. 北京: 人民邮电出版社, 2016, 1-375
- [6] 朱崇来. 基于 MySQL 组复制技术数据备份策略实现[J]. 电子世界, 2018(06):193+195.
- [7] 黄俊勇. 基于 Spring 框架的研究与设计[J]. 电脑知识与技术, 2018, 14(03):116-117+120.
- [8] 吴沧舟, 兰逸正, 张辉. 基于 MySQL 数据库的优化[J]. 电子科技, 2013, 26(09):182-184.
- [9] 张伟丽, 江春华, 魏劲超. MySQL 复制技术的研究及应用[J]. 计算机科学, 2012, 39(S3):168-170.
- [10] 白尚旺, 党伟超. 软件分析建模与 PowerDesigner 实现. 北京: 清华大学出版社, 2010.
- [11] 张海藩. 软件工程导论. 北京: 清华大学出版社, 2008.
- [12] 王晓敏, 邝孔武. 信息系统分析与设计. 北京: 清华大学出版社, 1998.
- [13] 秦航, 杨强. 软件质量保证与测试. 北京: 清华大学出版社, 2014.