
摘 要

在信息飞速发展的时代里，信息的价值渐渐体现到了各行各业中，一个好的校园在线视频学习网站在提高日常学习效率的同时，需要收集在线视频学习行为的各种信息。怎样利用现代新的科学技术建立一个在线视频学习平台，挖掘在线视频学习网站背后的信息，显得越来越重要。提高学生学习效率、教师教学质量，反馈学生学习情况，体现课程教学问题，加强教育资源的管理，实现学习资源平等，成为我们亟待解决的一个问题，本次毕业设计着力以上问题，通过设计一个在线视频学习网站，来实现提高学生学习效率、加强教育资源的管理、实现学习资源平等，通过数据挖掘和数据分析，建立数学模型尝试解决，提高教师教学质量、反馈学生学习情况、体现课程教学等问题。

在设计和开发过程中,我用了 JSP 结合 Bootstrap 框架，完成了前端的设计与制作，后端使用 Java 作为后端程序设计语言，以 Mysql 为数据库，Tomcat 为 Web 服务器进行开发。网站为 B/S 架构，开发过程中遵循了 MVC 架构模式，降低了系统的复杂性，提高了系统的可扩展性。首先我将系统分为前台学生课程页面展示与浏览、教师课程管理与上传、管理员网站更新与美化三大模块。我们利用 JavaScript 实现异步传输，用 Java 实现文件视频上传，通过利用 Bootstrap 精心设计，提高网站的观赏性，让人耳目一新，最后利用 RFM 模型、AHP 模型、以及 K-均值建立了一个数学模型用来处理和评价在线视频学习网站收集到的用户和行为信息，并利用 ECharts 框架做一个简单的数据可视化的体现，给出一个智力决策意见用作教师或管理员的决策参考。

关键词：在线学习完整，设计与实现，JSP，JAVA

目 录

第一章 绪 论.....	1
1.1 课题的背景和研究目的.....	1
1.1.1 课题的背景.....	1
1.1.2 课题的目的.....	2
1.2 选题的意义和价值.....	2
1.3 国内外研究现状.....	2
1.4 研究内容.....	3
第二章 相关技术.....	5
2.1 Java 技术.....	5
2.2 HTML 技术.....	5
2.4 jsp 技术.....	6
2.5 JavaBean.....	6
2.6 Servlet (Server Applet)	7
2.7 流媒体.....	7
2.8 MVC 模式.....	7
2.9 本章小结.....	8
第三章 需求分析.....	9
3.1 总体需求.....	9
3.2 网站前台需求分析.....	9
3.3 网站后台需求分析.....	10
3.4 教学资源的需求分析.....	11

3.5 在线视频学习行为的需求分析.....	11
3.6 开发工具和平台.....	11
3.6 本章小结.....	12
第四章 网站总体设计.....	13
4.1 网站设计原则.....	13
4.2 网站的开发方法及技术路线.....	13
4.3 网站结构图.....	13
4.4 子功能模块.....	15
4.4.1 命名规范.....	15
4.4.2 网站功能模块.....	15
4.4.3 游客用例图.....	16
4.4.4 学生用户用例图.....	17
4.4.5 教师用例图.....	17
4.4.6 管理员用例图.....	18
4.5 数据库设计.....	18
4.5.1 E-R 图.....	18
4.5.2 表关系图.....	19
4.5.3 数据库物理设计.....	20
4.7 本章小结.....	24
第五章 网站前台详细设计与实施.....	25
5.1 网站前台功能概述.....	25
5.2 网站前台系统流程图.....	25

5.3 网站前台各个功能模块详细设计与实现.....	25
5.3.1 首页设计与实现.....	25
5.3.3 登陆设计与实现.....	29
5.3.4 专业分类设计与实现.....	33
5.3.5 课程列表页面设计与实现.....	34
5.3.6 章节页面设计与实现.....	35
5.4 本章小结.....	36
第六章 数据挖掘与机器学习详细设计与实施.....	37
6.1 数据收集功能详细设计与实施.....	37
6.1.1 在线视频学习视频播放页面的功能概述.....	37
6.1.2 数据收集功能概述.....	40
6.1.2 用户行为数据收集.....	40
1、用户活跃度.....	40
2、用户黏性.....	41
3、用户产出.....	42
6.1.3 视频行为数据收集.....	43
6.3 机器学习功能详细设计与实施.....	45
6.3.1 机器学习概述.....	45
6.3.2 RFM 模型、AHP 模型以及 K-均值聚类的应用.....	46
1、模型的概述.....	46
1、模型假设.....	46
2、模型的建立.....	48

3、RFM 权重的分析.....	49
4、基于 K-均值聚类用户学习态度分类.....	49
5、学习用户类型识别分析.....	50
6、学习用户学习态度比较分析.....	52
6.3.2 算法设计与实现.....	53
1、复杂事物抽象化.....	53
2、时间量化.....	54
3、算法的实现.....	55
6.4 数据可视化功能详细设计与实施.....	56
6.5 本章小结.....	58
第七章 网站后台详细设计与实施.....	59
7.1 网站后台功能概述.....	59
7.2 网站后台系统流程图.....	59
7.3 网站后台各个功能呢个模块详细设计与实现.....	59
7.3.1 教师课程显示设计与实现.....	59
7.3.2 视频上传设计与实现.....	63
7.4 本章小结.....	64
第八章 总结与展望.....	65
致 谢.....	66
参考文献.....	67

第一章 绪论

1.1 课题的背景和研究目的

1.1.1 课题的背景

随着时代高速发展，传统化教学已经不能满足人们快节奏的学习发展需求，理想的学习模式逐渐地向现代化互联网模式的转变，并提倡终身教育。随着计算机技术高速发展，在线视频学习网站的出现很好地满足人满这一理念，而在线视频学习其自主、方便、个性的特点已然成为了新时代的需要。

运用互联网资源在线进行专题学习，这是一种现代新的学习方式。人们可以利用网络、去寻找资料、更加形象的寻找问题、分析问题、解决问题，人们还可以上网互相交流，请教网上的专家。学习的全部方式由自己做主，自觉主动的去使用互联网资源。这充分的调动了人们学习的自发性，学习方式不在时传统的学习方式，学习成本大大降低。视频化学习形式多样，较单纯的书本知识更具体，使人们理解和吸收更加高效。

相比传统学习方式，当今在线视频学习网站有以下优势：

1、学习资源整合化

专题资源某类专题化的相关资料，如今时代信息急剧增加，人们获得的信息越来越丰富，在大量的信息筛选我们需要的学习某类知识，往往是费时费力的。专题资源在线视频学习网站，整合了学习资源，这使得学习更加专题化、更加便捷。

2、学习内容可视化。

在线视频学习，是具有图像具有声音的学习资料，比起传统的书本上的知识显得更加形象，让人们更加印象深刻。把一些深奥的、难懂的通过视频讲解的方式，可以大大提高学习的效率。

3、学习更个性化。

在线视频学习更具有针对性，学习内容变为被动方，学生成为主动方，灵活设置专业与课程，提高学生个性化学习的能力，适应社会上各行各业对人才的需求。网络学习中，学员能根据自身发展需要进行选择性地学习，这样既为学员节省了很多时间，也方便了学习。

4、学习更有互动性。

在线式的学习，通过论坛，留言等许多方式，分享自己的学习经验，参考他人的学习经验，课程中困难的地方可以结成小组在网站上交流，比起一个人的学习，更有乐趣，也具有互动性，可以更加激发人们的学习的热情。

1.1.2 课题的目的

本课题的目的是进行设计在线视频学习网站的实现视频信息的挖掘，最总完成一个具有数据信息挖掘和收集的在线视频学习的“学生 MOOC 网站”的搭建。本课题是在研究国内外大学生 MOOC 知名网站的基础上，利用 Java、JSP、JavaBean、Servlet、HTML、CSS、JavaScript、流媒体等技术采用 MVC 模式开发而成。

首先对数据采集、在线视频学习网站的国内外发展进行分析，阐述相关技术知识，对课题的需求分析进行阐述，接着阐述本网站的详细设计部分，最后对具体功能进行介绍。整个项目完成了用户从浏览视频到登录；管理员从增删改查，到文件上传；数据挖掘从行为记录到行为区分等一系列功能，最终完成整个项目的实现在线视频学习的数据挖掘。

1.2 选题的意义和价值

同时互联网和云计算等技术的成熟，网络在线视频学习也得到了前所未有的发展，近年来出现了一批在线视频学习平台，学习者通过在线视频学习平台参与各种学习活动，其中，观看视频是最核心的学习环节，平台通过自动记录学习者的观看行为，将其作为评价或考核学习者的主要依据。目前，多数在线视频学习平台记录观看行为的方法仍比较简单，仅记录学习者播放视频的时间长度，该方法未完整记录下学习者的各种观看行为，其中蕴含着大量潜在价值的信息和知识。

本课题搭建了一个计算机在线视频学习网站，可以为学校学计算机专业的人们提供一些帮助；并可以为其他同学在今后研究在线视频学习视频信息挖掘网站设计，提供一些参考与借鉴；总之这次毕业设计弥补了我编程方面的很多不足，对实习时的职业技能有了很大的帮助。

1.3 国内外研究现状

据 CNNIC 发布的“第 38 次中国互联网络发展状况统计报告-网络娱乐类应用发展”显示，截止 2016 年 6 月，我国网络视频用户规模达 5.14 亿，网络视频用户使用率为 72.4%，目前，主流视频网站主要通过热门剧目的差异化编排方式来吸引用户付费，未来基于大数据对用户的深度运营和需求挖掘，能探索出更多的增值消费方式。

随着人们生活节奏加快，人们迫切需要从大量学习视频中快速、有效地发现信息资源

和知识。在大规模在线开放课程的普及中，对于学习者视频行为数据的采集很少受到关注，如何利用学习分析技术来处理这些大数据，使学习者可以高效快速的学习信息资源和知识逐渐成为研究者们研究的课题，在研究这些大数据时，仅有数据是不够的，还需要用有趣并容易理解的方式来展现数据，通过技术的手段，将枯燥的数据变得生动可爱，使受众更容易接受。随着网络传播的发达和大数据时代的到来，数据可视化已成为一项值得掌握的技能。

国内外视频学习网站选择学习资源的范围比较广，分类业很多样化，并且大多都有很强的商业性，其中美国的视频学习网站 **Coursera** 学习趣味比较高，国外公益性学习网站也多于国内。《中国大学 MOOC》是由爱课程网和网易云课堂联合推出的，不论是国内还是国外，MOOC 网站的视频来源大多都来源于高校，而所有高校的视频背后都有很多视频信息，可以反馈教学难点和重点。

1.4 研究内容

通过研究和体验大学生 MOCC 网、网易公开课、腾讯课堂、Coursera 等一系列学习网站的模式和相关技术，设计出一个针对在线视频学习视频信息挖掘网站，Mooc 信息挖掘网站的设计与实现，本论文的主要研究内容分为一下几个方面：

- 1、研究并介绍国内外在线视频学习网站的模式、收集信息的方法的发展情况和现状。
- 2、研究本论文所需的相关技术，例如 HTML、JAVA、JQuary、AJAX、JavaBean、Servlet 技术、流媒体技术。
- 3、通过研究分析 MVC 模式，本文即采用 JSP+Servlet+JavaBean 的模式进行设计与实现
- 4、设计出一个在线视频学习视频网站、使学校的视频资源可以放到我设计的网站供同学使用，并且用户能够对喜欢的视频进行一定的操作。

本论文结构如下：

第一章：绪论，阐述了“在线视频学习网站视频信息挖掘的设计与实现”这个课题的背景、选题意义、国内外研究现状。

第二章：阐述本课题中涉及到的相关专业知识与技术概念

第三章：描述本课题的需求分析、开发者工具和平台介绍

第四章：系统的总体设计、进行系统流程、功能模块、数据设计阐述

第五章：网站前台的详细设计与实现，包括所运用到的技术、关键代码以及截图

第六章：数据挖掘与机器学习详细设计与实施

第七章：网站后台的详细设计与实现，包括所运用到的技术、关键代码以及截图

第八章：总结与展望

第二章 相关技术

本章主要介绍完成 MOOC 信息挖掘网站中设计到的一些相关技术的概念及发展状况。

2.1 Java 技术^[1]

Java 是一种广泛使用的计算机编程语言，拥有跨平台、面向对象、泛型编程的特性，因为其跨平台性、安全性、高效性等特点，并且能广泛应用于大型网站、大型企业、网络通信、游戏开发以及嵌入式开发等领域，所以受到人们的喜欢。

Java 平台，由 Java 编写的软件赖以运行的平台，是 Java 软件和电脑系统的中介，Sun 公司还发布了 JDK 是在任何平台上运行 Java 编写的程序都需要用到的软件。终端用户可以以软件或者插件方式得到和使用 JRE，Sun 公司还发布了一个 JRE 的更复杂的版本，叫做 JDK，即 Java 2 开发包，里面包含了 Java 需要的编译器、参考文档和调试器等。

2.2 HTML 技术^[2]

HTML 是一种用于创建网页的标准标记语言。HTML 是一种基础技术，常与 CSS、JavaScript 一起被众多网站用于设计令人赏心悦目的网页、网页应用程序以及移动应用程序的用户界面。网页浏览器可以读取 HTML 文件，并将其渲染成可视化网页。

HTML 元素是构建网站的基石。HTML 允许嵌入图像与对象，并且可以用于创建交互式表单，它被用来结构化信息——例如标题、段落和列表等等，也可用来在一定程度上描述文档的外观和语义。HTML 的语言形式为尖括号包围的 HTML 元素（如<html>），浏览器使用 HTML 标签和脚本来诠释网页内容，但不会将它们显示在页面上。

HTML 可以嵌入如 JavaScript 的脚本语言，它们会影响 HTML 网页的行为。网页浏览器也可以引用层叠样式表（CSS）来定义文本和其它元素的外观与布局。维护 HTML 和 CSS 标准的组织万维网联盟（W3C）鼓励人们使用 CSS 替代一些用于表现的 HTML 元素。

2.3 JavaScript^[3]

一种高级编程语言，通过解释执行，是一门动态类型，面向对象（基于原型）的直译

^[1] Java-百度百科【EB/OL】

^[2] HTML-百度百科【EB/OL】

^[3] JavaScript-百度百科【EB/OL】

语言。它被世界上的绝大多数网站所使用，也被世界主流浏览器（Chrome、IE、FireFox、Safari、Opera）支持。JavaScript 是一门基于原型、函数先行的语言，是一门多范式的语言，它支持面向对象编程，命令式编程，以及函数式编程。它提供语法来操控文本，数组，日期以及正则表达式等，不支持 I/O，比如网络，存储和图形等，但这些都可以由它的宿主环境提供支持。

2.4 jsp 技术^[4]

JSP（全称 JavaServer Pages）是由 Sun Microsystems 公司倡导和许多公司参与共同创建的一种使软件开发可以响应客户端请求，而动态生成 HTML、XML 或其他格式文档的 Web 网页的技术标准。JSP 技术是以 Java 语言作为脚本语言的，JSP 网页为整个服务器端的 Java 库单元提供了一个接口来服务于 HTTP 的应用程序。

JSP 使 Java 代码和特定的预定义动作可以嵌入到静态页面中。JSP 句法增加了被称为 JSP 动作的 XML 标签，它们用来调用内建功能。另外，可以创建 JSP 标签库，然后像使用标准 HTML 或 XML 标签一样使用它们。标签库提供了一种和平台无关的扩展服务器性能的方法。

JSP 被 JSP 编译器编译成 Java Servlets。一个 JSP 编译器可以把 JSP 编译成 JAVA 代码写的 servlet 然后再由 JAVA 编译器来编译成机器码，也可以直接编译成二进制码。

2.5 JavaBean^[5]

JavaBeans 是 Java 中一种特殊的类，可以将多个对象封装到一个对象（bean）中。特点是可序列化，提供无参构造器，提供 getter 方法和 setter 方法访问对象的属性。名称中的“Bean”是用于 Java 的可重用软件组件的惯用叫法。

Javabean 的优点：

- 1、Bean 可以控制它的属性、事件和方法是否暴露给其他程序。
- 2、Bean 可以接收来自其他对象的事件，也可以产生事件给其他对象。
- 3、有软件可用来配置 Bean。
- 4、Bean 的属性可以被序列化，以供日后重用。

综上，JavaBean 是可调、可读写、可交互的。

^[4] JSP-百度百科【EB/OL】

^[5] JavaBeanK-百度百科【EB/OL】

2.6 Servlet (Server Applet)^[6]

全称 **Java Servlet**，是用 **Java** 编写的服务器端程序。其主要功能在于交互式地浏览和修改数据，生成动态 **Web** 内容。狭义的 **Servlet** 是指 **Java** 语言实现的一个接口，广义的 **Servlet** 是指任何实现了这个 **Servlet** 接口的类，一般情况下，人们将 **Servlet** 理解为后者。

Servlet 运行于支持 **Java** 的应用服务器中。从实现上讲，**Servlet** 可以响应任何类型的请求，但绝大多数情况下 **Servlet** 只用来扩展基于 **HTTP** 协议的 **Web** 服务器。

最早支持 **Servlet** 标准的是 **JavaSoft** 的 **Java Web Server**。此后，一些其它的基于 **Java** 的 **Web** 服务器开始支持标准的 **Servlet**。

2.7 流媒体^[7]

流媒体 (**Streaming media**) 是指将一连串的媒体数据压缩后，经过网络分段发送数据，在网络上即时传输影音以供观赏的一种技术与过程，此技术使得数据包得以像流水一样发送；如果不使用此技术，就必须在使用前下载整个媒体文件。流传输可发送现场影音或预存于服务器上的视频，当观看者在收看这些影音档时，影音数据在送达观赏者的电脑后立即由特定播放软件播放（如 **Windows Media Player**，**Real Player**，或 **QuickTime Player**）。

流媒体文件一般定义在 **bit** 层次结构，因此流数据包并不一定必须按照字节对齐，虽然通常的媒体文件都是按照这种字节对齐的方式打包的。

流媒体的三大操作平台是微软公司、**RealNetworks**、苹果公司提供的。

2.8 MVC 模式^[8]

MVC 模式 (**Model-view-controller**) 是软件工程中的一种软件架构模式，把软件系统分为三个基本部分：模型 (**Model**)、视图 (**View**) 和控制器 (**Controller**)。

MVC 模式的目的是实现一种动态的程序设计，使后续对程序的修改和扩展简化，并且使程序某一部分的重复利用成为可能。除此之外，此模式通过对复杂度的简化，使程序结构更加直观。软件系统通过对自身基本部分分离的同时也赋予了各个基本部分应有的功能。专业人员可以通过自身的专长分组：

^[6] ServletK-百度百科【EB/OL】

^[7] 流媒体-百度百科【EB/OL】

^[8] MVC 模式-百度百科【EB/OL】

控制器（**Controller**）- 负责转发请求，对请求进行处理。

视图（**View**）- 界面设计人员进行图形界面设计。

模型（**Model**）- 程序员编写程序应有的功能（实现算法等等）、数据库专家进行数据管理和数据库设计(可以实现具体的功能)

在最初的 JSP 网页中，像数据库查询语句(SQL query)这样的数据层代码和像 HTML 这样的表示层代码是混在一起。虽然有着经验比较丰富的开发者会将数据从表示层分离开来，但这样的良好设计通常并不是很容易做到的，实现它需要精心地计划和不断的尝试。**MVC** 可以从根本上强制性地将它们分开。尽管构造 **MVC** 应用程序需要一些额外的工作，但是它带给我们的好处是毋庸置疑的。

首先，多个 **View** 能共享一个 **Model**。在 **MVC** 设计模式中，**Model** 响应用户请求并返回响应数据，**View** 负责格式化数据并把它们呈现给用户，业务逻辑和表示层分离，同一个 **Model** 可以被不同的 **View** 重用，所以大大提高了代码的可重用性。

其次，**Controller** 是自包含（**self-contained**）指高独立内聚的对象，与 **Model** 和 **View** 保持相对独立，所以可以方便的改变应用程序的数据层和业务规则。由于 **MVC** 模式的三个模块相互独立，改变其中一个不会影响其他两个，所以依据这种设计思想能构造良好的少互扰性的构件。

此外，**Controller** 提高了应用程序的灵活性和可配置性。**Controller** 可以用来连接不同的 **Model** 和 **View** 去完成用户的需求，也可以构造应用程序提供强有力的手段。给定一些可重用的 **Model**、**View** 和 **Controller** 可以根据用户的需求选择适当的 **Model** 进行处理，然后选择适当的 **View** 将处理结果显示给用户。

综上所述 **MVC** 具有耦合性低、重用性高、生命周期成本低等特性，对开发效率、系统稳定都有很好的效果。

2.9 本章小结

本章为本课题相关的技术介绍，包括：**java** 技术、**JSP** 技术、**JavaBean** 技术、**HTML** 技术、**JavaScript** 技术、**Servlet** 技术、流媒体技术以及 **MVC** 模式。本文即采用 **JSP+Servlet+Javabean** 的模式进行学习资源在线视频网站设计，视频信息挖掘的实现。

第三章 需求分析

3.1 总体需求

本课题的主要目的是搭建一个在线视频学习平台、通过记录用户对视频流的操作行为记录，最总实现视频信息挖掘的设计与实现。

该网站方便地实现对教学视频的显示、录入、修改、查询、学习进度记录等功能。对用户来讲，用户可以浏览视频分类、课程介绍、视频信息、观看视频、留言评论、查找课程及视频等。

对于教师来讲、上传教学视频，发布教学任务，修改课题信息、修改课程信息，查看学生学习进度等。

对于管理人员来讲、可以实现分类的修改，用户的管理等。

3.2 网站前台需求分析

网站前台主要有两种身份：游客和登录用户。

游客可以浏览网站主页，查看课题分类，查看课题信息，查看课程信息，同时观看视频内容。

- 1、主页：显示课题分类信息，最新欢迎视频，宣传背景等。
 - 2、登录：输入正确的用户名和密码可登录本学习网站。
 - 3、搜索：通过输入关键字查找的相关课程
 - 5、专业分类模块：将所有专业分类展现到页面当中
 - 6、课程列表模块：将一类课题列出展现到页面当中，其中包括课题名称、课题摘要等信息。
 - 7、课程章节列表模块：此页面包含了课程各个章节预览信息。
 - 8、播放模块：用户通过此页面观看视频内容，可以选择本章节中各个小节课程进行播放。
 - 9、评论模块：用户在观看的同时，可以对课程进行评论和提问，用户互相可以答疑互动。
 - 10、视频测试题：回答试题问卷内容，反映视频内容的吸收程度。
- 登录用户在拥有游客的所有操作权限的同时，还可以查看课程进度、个人观看记录等。
- 1、课程进度模块：将观看过的课程和为观看过的课程做一区分，并显示。
 - 2、历史观看记录模块：将浏览过的信息显示出来。

3、评分系统模块：通过数据采集、数据分析，给出一个教室和用户都可以看到的一个评分，来区分学习的态度能力等，最后可以通过这个评分来给予用户学习成绩的评定。

4、教师留言模块：显示教师对自己学习进展的一个评价、建议和要求。

3.3 网站后台需求分析

网站后台主要有两种身份：教师和管理员。

教师可以进入到自己的管理页面，修改课程详细信息、修改章节详细信息、新增教学视频、上传课程、查看学生学习评分、学习进度，发送消息。

课程管理模块：可以在专业分类下自定义创建是视频课程专题，并保存，用来存放后续想上传的课程视频；在课程分类下可以进入指定课程内上传课程内容，如课程教师、课程简介、自定义创建课程章节；在章节分类下可以进入指定章节进行视频的上传，视频的替换；如果是课程、章节过多可以利用课程列表模块的查找功能，来根据视频名称查找出想删除、修改、上传的课程和章节。

学生管理模块：可以在学生列表下进入某个学生的学习记录模块，通过学生的学习评分、学习进程等信息来对学生的学习情况进行了解，可以给学生学习情况提出学习指导意见和要求，也可以调整或删除原来对于学生学习情况的指导意见和要求。

1、课程修改：修改课题的详细信息

2、章节修改：修改章节的详细信息

3、查看学生：查看观看自己视频的学生列表及学习进度

4、课程上传：课程信息的上传

5、视频上传：教学视频的上传

6、单元测试题：测试学生对视频内容的理解程度，掌握程度，以在线问卷的形式给出。

7、学习意见新增：给学生学习情况提出学习指导意见和要求。

8、学习意见修改：调整或删除原来对于学生学习情况的指导意见和要求

管理员在拥有教师的权限的同时、还可以设置网页的各个显示功能。

数据挖掘及显示模块：数据挖掘根据其目标可以分为两个具体模块，一、单用户多视频的数据挖掘，主要考虑到学生的学习情况、学习兴趣和学习力度等，需要收集到学生的活跃度、学生的活跃频率以及学生的活跃时间等等信息。二、多用户单视频的数据挖掘，主要考虑到教学内容的反馈、学习视频的质量、学习内容的兴趣浓度、课程中难易点突出等等

需要收集到视频的黏度信息、用户观看视频的行为信息以及视频行为频率及行为概率信息等等。数据显示本课题希望通过数据的收集，以及算法的分析，最终给学生教师反馈两个非结构化的信息：学生学习评分，学习视频评分。

3.4 教学资源的需求分析

本课题作为在线视频学习视频学习行为信息挖掘网址的设计与实现，不仅包括在线视频学习的相关网站数据库、网站前台、网站后台的设计，也包括学习视频的收集整理与维护。

本课题设计“IMUA MOOC”在线视频学习网站，目前主要针对的是本专业学生，课程也主要有：Java、JavaWeb、JavaScript、等专题课程学习，这些计算机学习资源主要的来源为：各个培训机构的教学视频，有达内、尚硅谷、传智、北京圣思园等。由于暂时没有学校老师的教学视频，前期只能用培训学校的视频来做在线视频教学内容，用户和学生还是可以学习到学校教学相关的理论知识，还可以学习到企业相关的实际应用，同时还能了解到其个人学习的心得体会，达到相互交流相互学习、共同提高的效果。

3.5 在线视频学习行为的需求分析

本课题的主要目标是帮助学生了解课程中的重点、难点，帮助教师及时了解学生的学习进度、学习状态等，而数据的收集再本次课题中的作用就非常关键。通过数据分析的两个目标：展现学生学习进度、学习状态、展现教学课程内容中的难点、重点，将数据收集也大致分为以下两个方面：

1、单用户多视频行为数据收集，主要目的反映学生的学习进度、学习状态；单用户的黏性数据收集：用户访问频率、用户访问间隔、平均用户访问间隔、平均用户访问频率；单用户的活跃数据收集：用户播放视频时间、平均用户播放时间

2、多用户单视频行为数据收集，主要反映教学视频质量、教学内容吸引度、视频内容重点；视频的黏性数据收集：视频播放次数、视频播放完成次数、平均视频播放次数、平均视频播放完成次数；视频行为的活跃性数据收集：视频行为次数、平均视频行为次数、用户互动次数（留言评论条数）、平均用户互动次数、播放网页停留时长、平均播放页面停留时长、平均播放速度、视频播放速度

3.6 开发工具和平台

开发平台：macOS Sierra 10.12.2

开发工具：Eclipse-jee+Textastic+Adobe Photoshop CC;

开发环境：JavaJDK 8.0+JavaSE 1.8;

数据库：Mysql 5.7.17;

网站部署环境：JRE 8.0 +Tomcat v9.0;

浏览器：Safari 浏览器，Firefox 浏览器。

3.6 本章小结

本章详细罗列了课题研究的需求分析。其中包括：对总体需求的分析、以及对网站前台与后台的需求分析、教学资源来源需求分析、数据收集需求分析、开发工具和平台的介绍等。明确了系统设计的主要目标，为后续的系统总体设计打下了基础。

第四章 网站总体设计

4.1 网站设计原则

本课题设计与实现在线视频学习网站“IMAU MOOC”，网站的数据信息量大，实现的功能复杂，对这样一个网站的设计要遵循以下几个原则：

- 1、实用性：基本的视频播放，能满足用户的基本观看需求，基本数据收集功能，满足用户在线视频学习行为信息的收集；
- 2、可维护性：程序结构合理、代码编写规范，具有充分的可读性，易于他人维护；
- 3、可扩展性：专在线视频学习网站不仅要考虑现在的使用要求，而且需要考虑以后的网站升级，以及相应的成本。

4.2 网站的开发方法及技术路线

本网站的主要遵循的模式为 MVC 的模式进行开发，即进行基于模型、视图、控制相结合的结构化方法。其中模型层，也称为 Model 层，此为业务层，采用 javabean 技术，将实际的视频用户，抽象为 bean 模型；视图层，也称为 View 层，此为交互层，采用 JSP 技术，并将 bootstrap 框架应用到其中；控制层，也称为 Control 层，此为基于模式与视图之间的连接与转换，采用 Servlet 技术。本系统就是采用 JSP + servlet + javabean 的 MVC 模式的开发方式。

本文先通过对 JSP、servlet、javabean 相关技术进行介绍，并对其技术进行学习和理解，然后从在线视频学习网站的系统分析、系统设计、系统实现和系统测试等方面开展设计工作。首先，系统分析包括可行性分析和需求分析，在需求分析中给出了总体组织结构图和数据流图。系统设计包括总体设计、详细设计；总体设计给出了总体规划外部关系图，使该系统的整体设计思路清晰；详细设计则对具体的功能模块进行了设计与实现，并完成系统测试。

4.3 网站结构图

在线视频学习网站的总体结构图，包含基于各个模块之间的隶属关系即管理与被管理的关系，如图 4-1 所示。

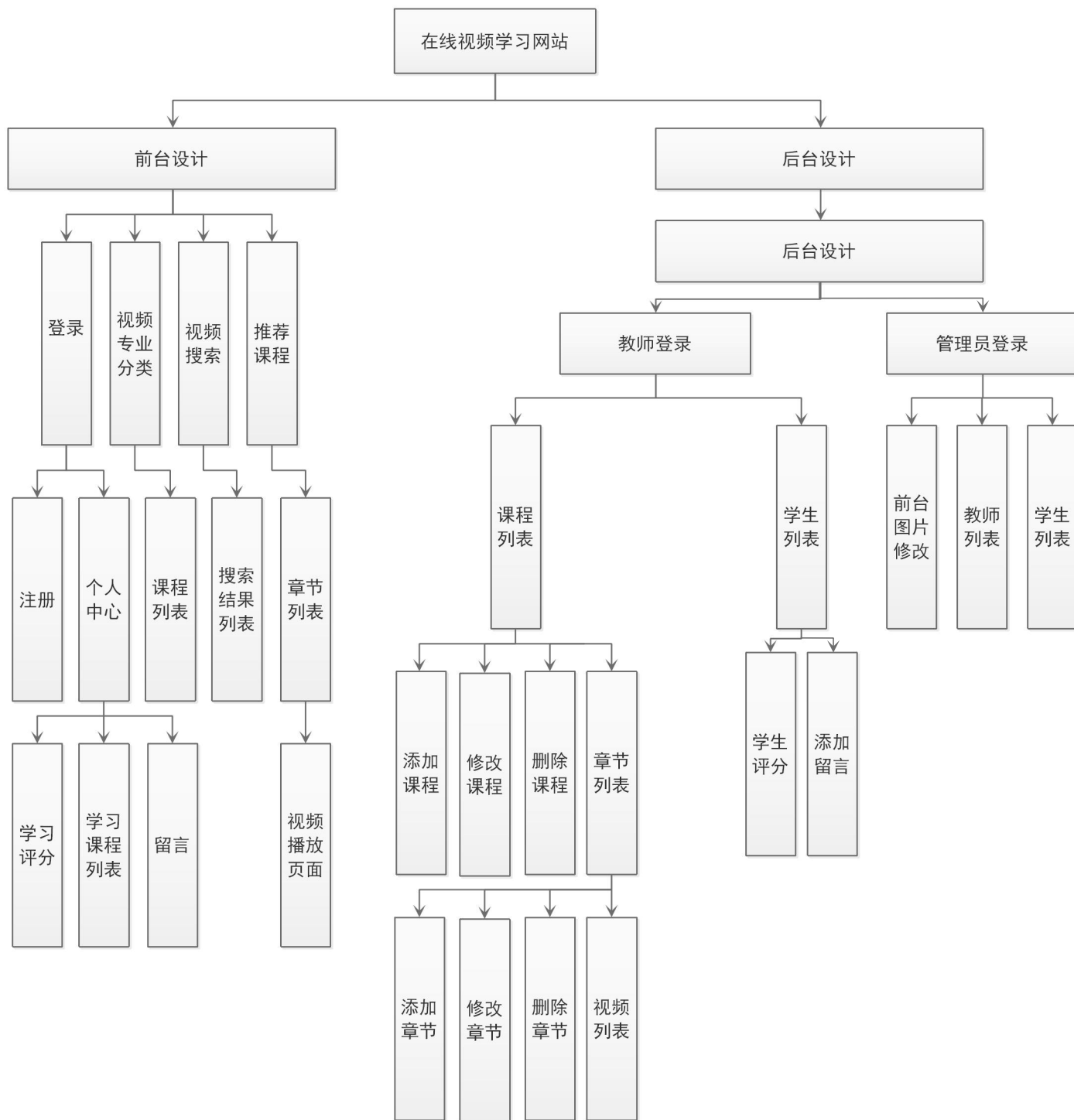


图 4-1 网站组织结构图

4.4 子功能模块

4.4.1 命名规范

定义规范的目的是为了使项目的代码样式统一，使程序有良好的可读性。一个正规的软件产品少不了要有规范的命名，一个好的软件产品一定要有个约定俗成的命名规则，本次课题研究项目中所遵守的命名原则如表 4-1 所示。

表 4-1 网站命名原则表

模块名称	命名要求	编号
项目名	所有字母小写	M1
包名	所有字母小写	M2
类名	首字母大写，如果类名由多个单词组成，每个单词的首字母都要大写	M3
变量名	除首字母小写，其它单词首字母大写	M4
常量	所有字母大写	M5
构造函数	名称及名称大小写完全一致	M6
普通方法	除首字母小写，其它单词首字母大写，单词尽量使用动词	M7

4.4.2 网站功能模块

模块命名规则确定后，需要对本系统设计到的模块及功能进行总结归纳成概要表，作为后续设计的指导，如表 4-2 所示。

表 4-2 网站功能概要设计表

网站前台首页	
模块名称	功能简述
登录、注册、个人中心	提供用户登录、注册、已组成用户进入个人首页
推荐课程列表	列出教师选出或点播次数比较高的课程
课程列表链接	可以跳转到课程详细列表
搜索栏	模糊查找后，将结果显示到课程列表页面

课程列表页面	
模块名称	功能简述
专业分类列表	提供选择专业来筛选列出的课程列表内容
课程列表显示区域	根据显示条件来显示课程列表，默认显示第类专业课程
课程详细页面	
课程详细介绍	展示课程简介，课程教师，课程学时等信息
课程章节列表	列出整个课程的各个章节并显示章节概述，点击可以观看相关视频
视频播放页面	
模块名称	功能简述
选集列表	可以选择上一节、下一节观看本章节相关视频，越界后自动跳到其他章节
播放器控件	利用 HTML5 的播放控件，进行播放学习视频
评论列表	显示学生的评论与互动，倒序显示，最后评论最先显示
视频行为记录	记录数据分析所需要的信息，AJAX 存储到数据库中，不显示到前台
管理登录页面	
模块名称	功能简述
登录验证	检测用户名密码是否正确
后台管理页面	
模块名称	功能简述
课程管理	对整个课程信息的增删改、章节的增删改、视频的增删改
用户管理	显示学生信息、学习评分、留言建议
评论管理	删除对于学习有误导的评论

4.4.3 游客用例图

网站未登录默认为游客用户，系统通过登录可以判定是学生、教师还是管理员，对游客、学生、教师、管理员的用户用例进行具体分析，以下为游客的用例图，如图所示 4-2 所示。

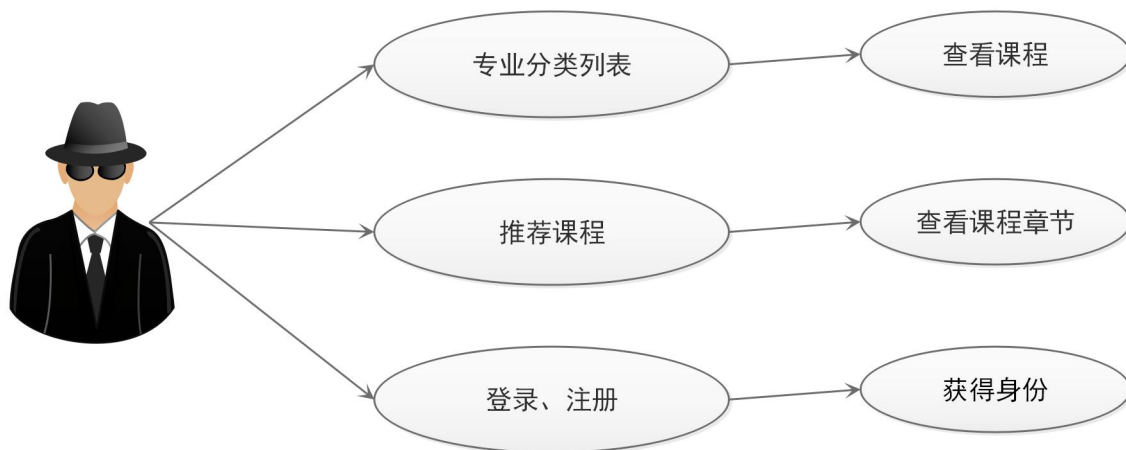


图 4-2 游客用例图

4.4.4 学生用户用例图

学生用户通过用户账户进行登录，其学生用户用例图，如图所示 4-3 所示。

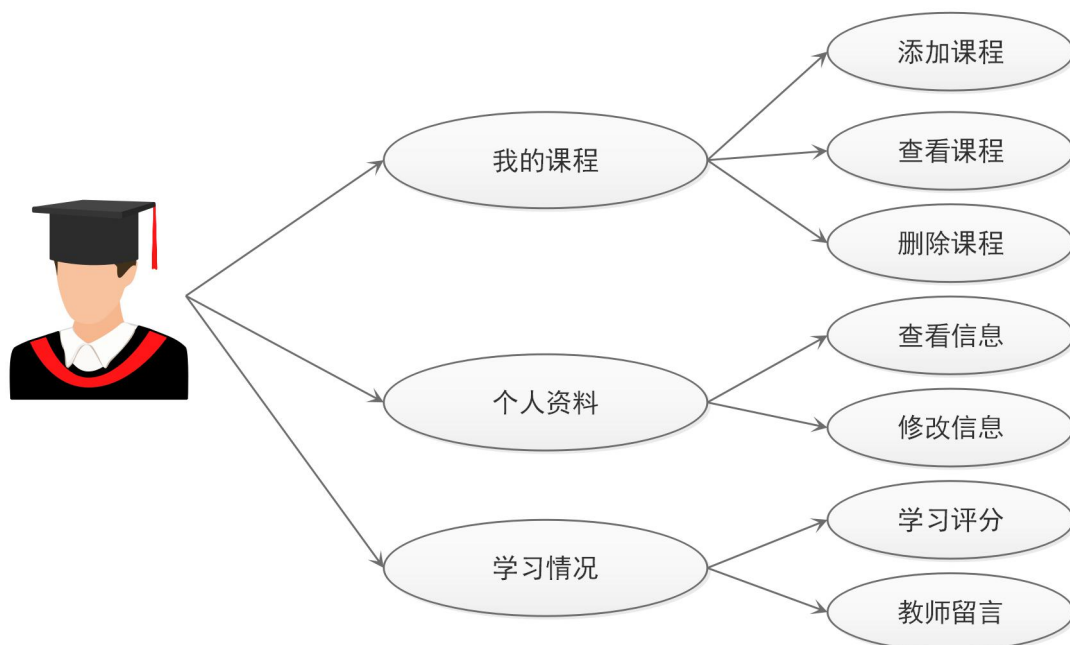


图 4-3 学生用户用例图

4.4.5 教师用例图

教师通过用户账户进行登录，其教师用例图，如图所示 4-4 所示。

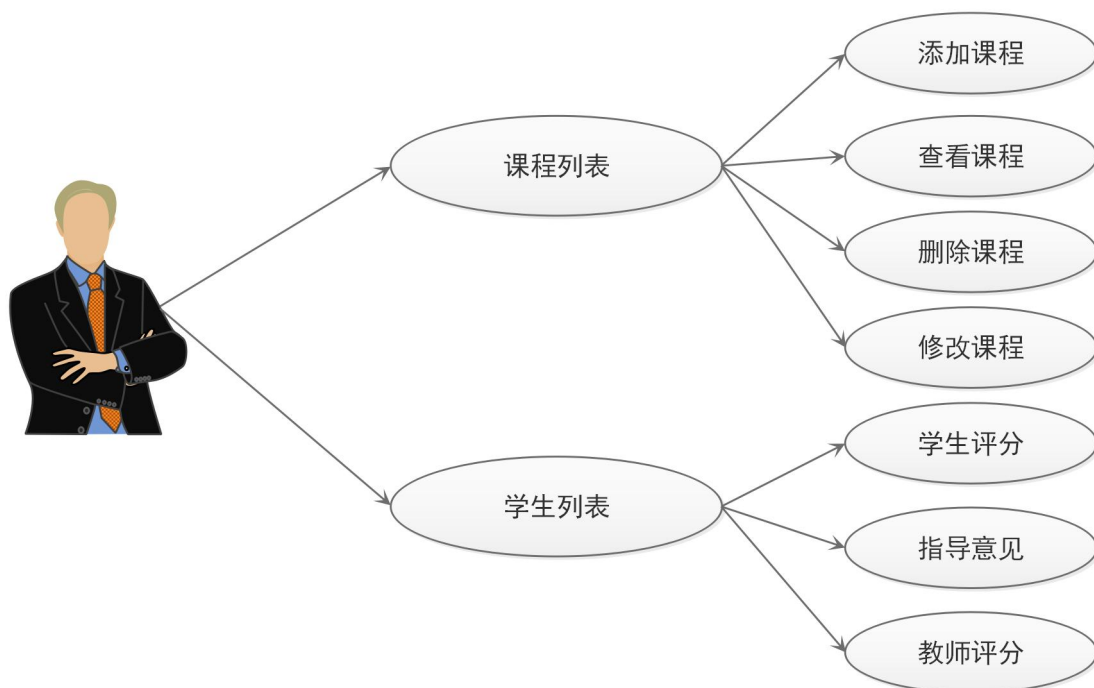


图 4-4 教师用例图

4.4.6 管理员用例图

管理员通过用户账户进行登录，其管理员用例图，如图所示 4-3 所示。

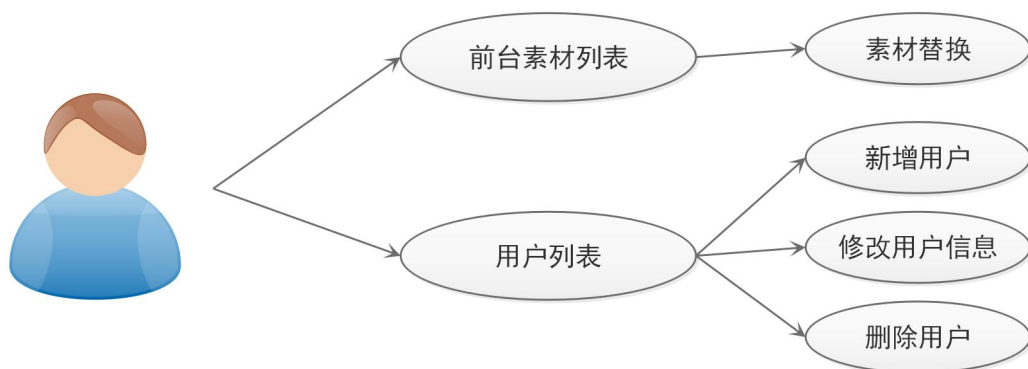


图 4-5 管理员用例图

4.5 数据库设计

数据库设计，一方面包括需要弄清楚数据库系统的 E-R 图以及表之间的关系图，这是数据库逻辑设计部分；另一方面包括所有的数据类型汇总，以及不同数据类型的具体描述，这是数据库物理设计部分。

4.5.1 E-R 图

对于数据库的设计，首先需要清楚其具体的数据的模型，其具体的实体联系方式，E-R 图如图 4-6 所示。

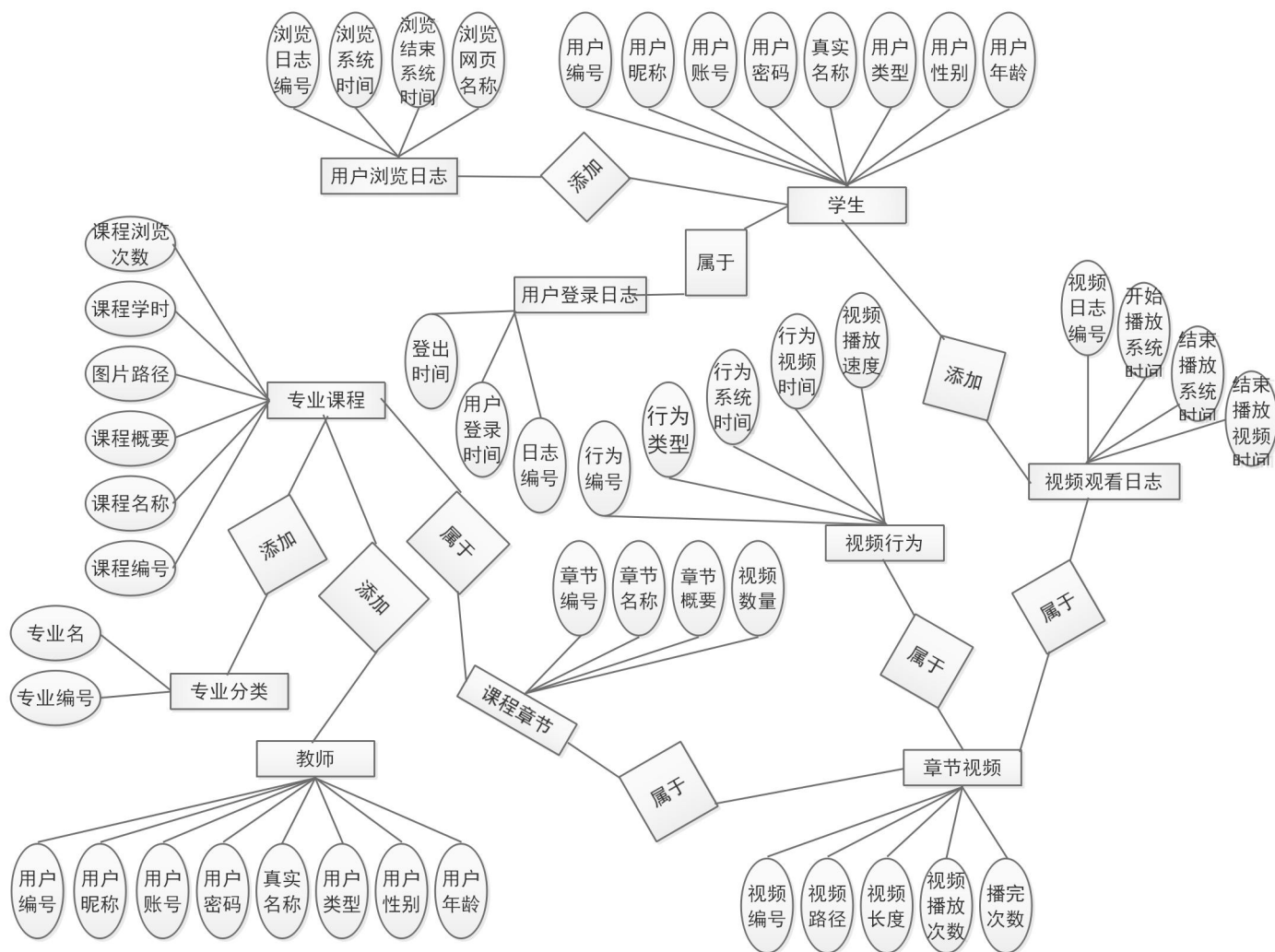


图 4-6 E-R 图

4.5.2 表关系图

数据库需要进行正规化处理，这意味着不同的数据之间的关系需要确定。首先需要确定不同的数据表中所包含的基础的元素数据，这是表关系图的基础。把这些基础的元素数据进行分类后，不同的数据表因为这些基础的元素数据进行了内在的联系本课题表之间的关系如图 4-7 所示。

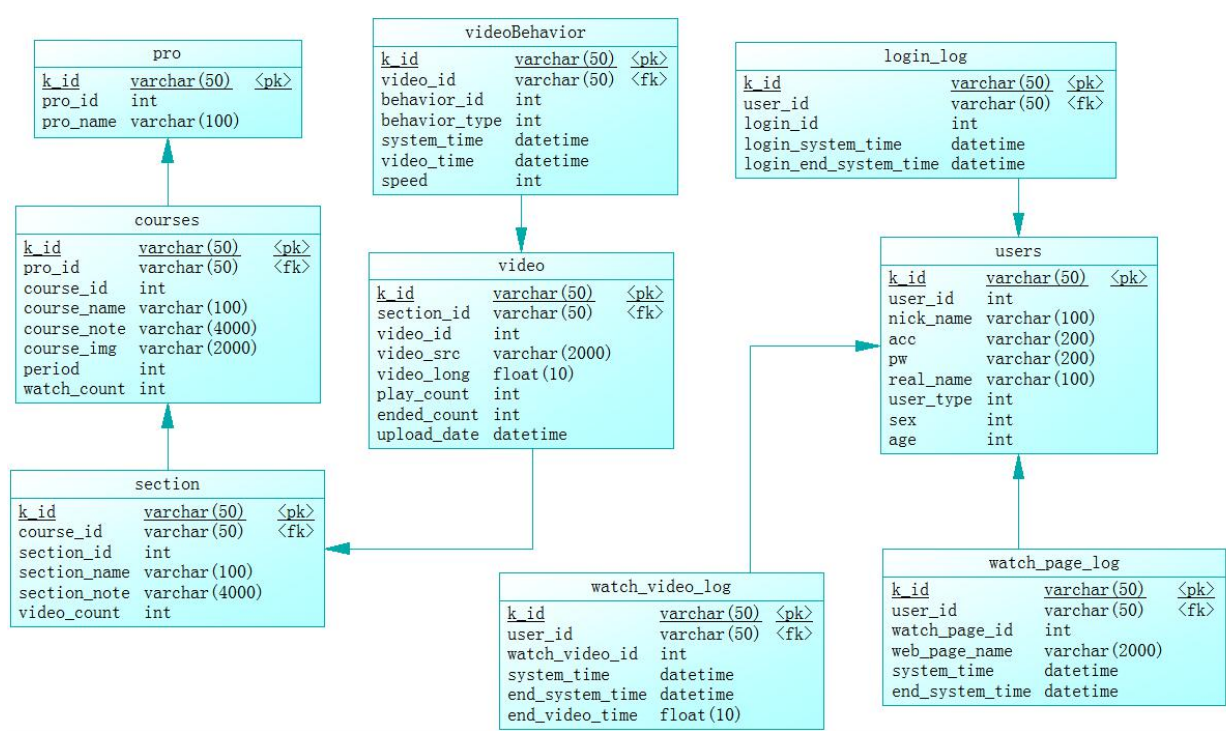


图 4-7 表关系图

4.5.3 数据库物理设计

数据库物理设计，包括所有的数据类型汇总，以及不同数据类型的具体描述。本课题数据库表汇总表如下。

表 4-3 数据库表汇总

表名	功能说明
pro	专业分类信息
course	课程详细信息
section	章节信息
video	视频详细信息
video_behavior	在线观看视频行为信息
users	用户信息
login_log	用户登录日志信息
watch_page_log	用户浏览网页日志信息

watch_video_log	用户观看视频日志信息
-----------------	------------

不同的表名代表着不同的数据结构类型对现实事物的具体描述。具体对某种数据结构类型的数据设计，见下面一系列的表。

表 4-4 pro 表

表名	Pro			
列名	数据类型	空/非空	约束条件	说明
Pro_id	Int	Not null	主键	专业编号
Pro_name	Varchar(100)	Not null	唯一	专业分类名称
补充说明	课程专业分类表			

表 4-5 course 表

表名	course			
列名	数据类型	空/非空	约束条件	说明
Course_id	Int	Not null	主键	课程编号
Pro_id	Int	Not null	唯一	专业编号
Course_name	Varchar(100)	Not null	唯一	专业分类名称
Course_note	Varchar(4000)	Null		课程内容介绍
Tea	Int	Not null	唯一	教师编号
Course_img	Varchar(2000)	Null		图片路径
period	Int	Null		学时
Watch_count	Int	Null		浏览次数
补充说明	课程表，记录课程详的细信息			

表 4-6 section 表

表名	section			
列名	数据类型	空/非空	约束条件	说明

Section_id	Int	Not null	主键	章节主键
Course_id	Int	Not null	唯一	课程编号
Section_bid	Int	Not null		章节编号
Section_name	Varchar (100)	Not null	唯一	章节名称
Section_note	Varchar (4000)	Null		章节概述
Video_count	Int	Null		章节视频数
补充说明	章节表，记录章节信息			

表 4-7 video 表

表名	video			
列名	数据类型	空/非空	约束条件	说明
Video_id	Int	Not null	主键	视频主键
course_id	Int	Not null	唯一	课程编号
section_id	Int	Not null	唯一	章节编号
Video_bid	Int	Not null		视频编号
Video_src	Varchar (100)	Not null		视频路径
Video_long	Varchar (4000)	Null		视频长度
play_count	Varchar (2000)	Null		播放次数
Ended_count	Int	Null		播放完成次数
upload_time	datetime	Not null		上传时间
补充说明	视频表，记录视频信息收集视频观看记录			

表 4-8 videoBehavior 表

表名	videoBehavior			
列名	数据类型	空/非空	约束条件	说明
Behavior_id	Int	Not null	主键	行为编号
video_id	Int	Not null	唯一	视频编号

Behavior_type	Int	Not null		行为类型
System_time	datetime	Not null		发生时系统时间
Video_time	Varchar(100)	Not null		发生时视频时间
speed	Int	Not null		当前播放速度
补充说明	行为记录表，记录行为发生时的详细信息			

表 4-9 users 表

表名	users			
列名	数据类型	空/非空	约束条件	说明
user_id	Int	Not null	主键	用户编号
Nick_name	Varchar(100)	Null	唯一	用户昵称
acc	Varchar(200)	Not null	唯一	用户账户
pw	Varchar(200)	Not null		账户密码
Real_name	Varchar(100)	Null		真实姓名
User_type	Int	Null		用户类型
sex	Int	Null		性别
age	Int	Null		年龄
Register_time	Datetime	Not null		注册时间
补充说明	用户表，注册时记录非空字段，其他字段后续添加			

表 4-10 Login_log 表

表名	Login_log			
列名	数据类型	空/非空	约束条件	说明
Login_id	Int	Not null	主键	登录记录编号
user_id	Int	Not null	唯一	用户编号
system_time	Datetime	Not null		登录系统时间
type	Int	Not null		登出系统时间

补充说明	登录日志表，记录课程详的细信息
------	-----------------

表 4-11 Watch_page_log 表

表名	Watch_page_log			
列名	数据类型	空/非空	约束条件	说明
Watch_page_id	Int	Not null	主键	日志记录编号
user_id	Int	Not null	唯一	用户编号
page_name	Varchar (2000)	Not null		浏览页面名称
System_time	Datetime	Not null		系统时间
type	Int	Not null		行为类型
补充说明	课程表，记录课程详的细信息			

表 4-12 Watch_video_log 表

表名	Watch_video_log			
列名	数据类型	空/非空	约束条件	说明
Watch_video_id	Int	Not null	主键	日志记录编号
user_id	Int	Not null	唯一	用户编号
video_id	Int	Not null	唯一	日志记录编号
System_time	Datetime	Not null		系统时间
video_time	Datetime	Not null		视频时间
Type	Int	Not null		行为类型
补充说明	课程表，记录课程详的细信息			

4.7 本章小结

本章为网站的总体设计，其中有网站的总体设计原则、以及本课题的开发方法及技术路线、网站功能设计、用户用例、数据库设计、开发界面几方面的内容。

第五章 网站前台详细设计与实施

5.1 网站前台功能概述

本课题设计与实现在线视频学习网站“IMAU MOOC”，网站前台是用户通过前台网页来学习和浏览“IMAU MOOC”网站，前台分为首页、注册、登录、视频、搜索、视频分类、课程详情、视频播放、视频评论等。

5.2 网站前台系统流程图

前台系统流程图如图 5-1 所示。

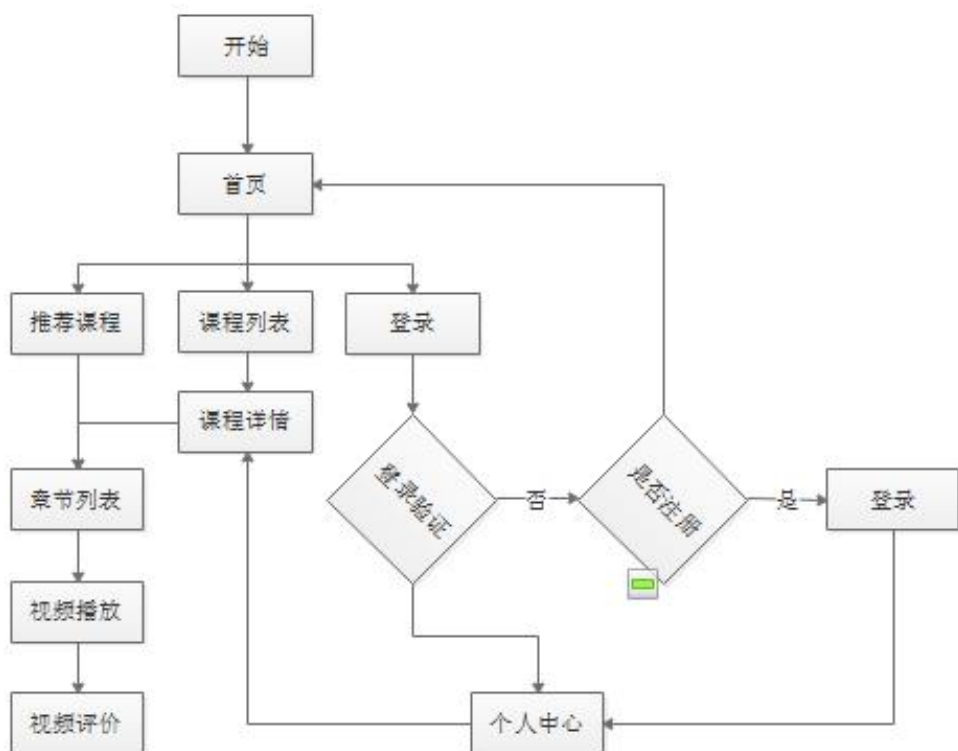


图 5-1 前台网站系统流程图

5.3 网站前台各个功能模块详细设计与实现

5.3.1 首页设计与实现

1、首页设计概述

首页设计概述包括设计的原理、流程。

用户访问首页时加载的是动态页面，此页面中的最新推荐视频模块、用户登录模块、进入课程按钮，首页是通过 first.jsp 通过重定向到 Servlet 中，通过 Servlet 调用业

务层中获取推荐视频的视频 id 列表，又调用 dao 方法 CourseDao 访问数据库中的课程实体类，匹配推荐视频 id 来获取课程列表，将返回的值分别存放到 List 集合中，采用重定向（request.getRequestDispatcher）的方式将所有 List 集合重定向到 jsp 并进行解析，最后追加到 JSP 页面显示给用户。用户登陆模块通过 AJAX 异步 Post 请求到 Servlet 中通过 HttpServlet 中的 GetParamter() 方法获取用户名和密码。如果登陆成功后将 User 对象存到 Session 中并显示到前台。

2、前台的实现

用户在输入网站地址后，首先看到的是网站前台首页。首页的主要功能有，登录、视频推荐、进入学习页面等等。前台视频界面设计主要是设计用于用户自己的视频页面，其设计的目的是用于用户以及好友交互浏览信息，让界面尽量简单，易懂，易操作，便于用户操作。

Servlet 部分代码：

功能解释：通过 Get 请求，调用到无参数的 Index.do，通过 RecommendDao 来获取数据库中推荐列表相同的课程列表并放到 Request 容器中，通过重定向到 Index.jsp。

```
RecommendDao dao = new RecommendDao();
List<Course> recommend = dao.getAllList();
request.setAttribute("recommendList", recommend);
request.getRequestDispatcher("index.jsp").forward(request, response);
```

JSP 部分关键代码：

功能解释：将 Servlet 放在 Request 容器中的推荐课程列表（recommendList）取出，通过 JSTL 将其显示出来。

```
<div class="row">
    <c:forEach items="${recommendList}" var="i">
        <div class="col-md-4">
            <h2>${i.course_name}</h2>
            <p>${i.course_note}</p>
            <p>
                <a class="btn btn-default" href="section.do?course_id=
                ${i.course_id}" role="button">
```



```
        进入课程 &raquo;;  
    </a>  
  
</p>  
  
</div>  
  
</c:forEach>  
  
</div>
```

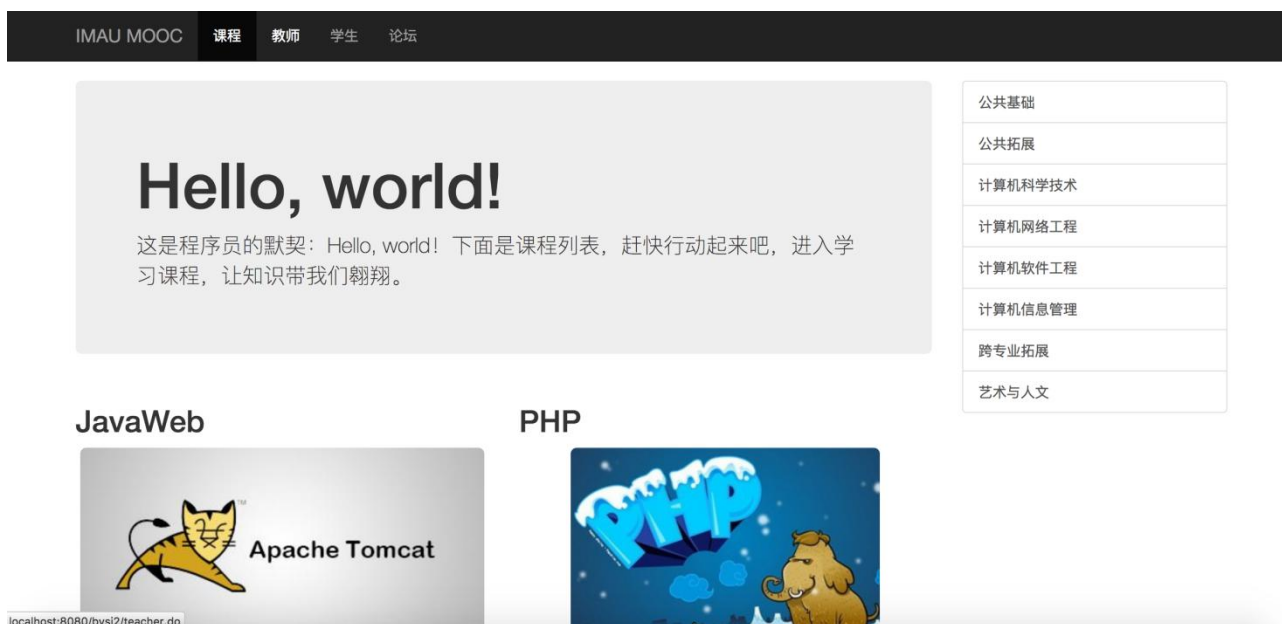


图 5-2 网站首页

5.3.2 注册设计与实现

1、注册设计概述

注册设计概述包括注册设计的原理、流程。

当用户填写完用户名后会对其做一次验证，当用户填写完所有资料选择注册按钮，将此用户名传到 Servlet 中，然后调用业务层中的查询方法，业务层再调用数据访问层，对数据进行操作并返回相应的值，若当前用户名在数据库中已存在则提示用户此用户名已被注册，若不存在则执行插入语句，注册成功后自动跳转到首页。

2、注册实现

如果客户不是本网站的用户，在注册界面上，需要填写邮箱地址、密码地址重新注册，如果没有注册才能注册。当用户把所有信息正确填写完成后，点击“提交注册”按钮，才能注册。注册的流程图如图 5-3 所示。

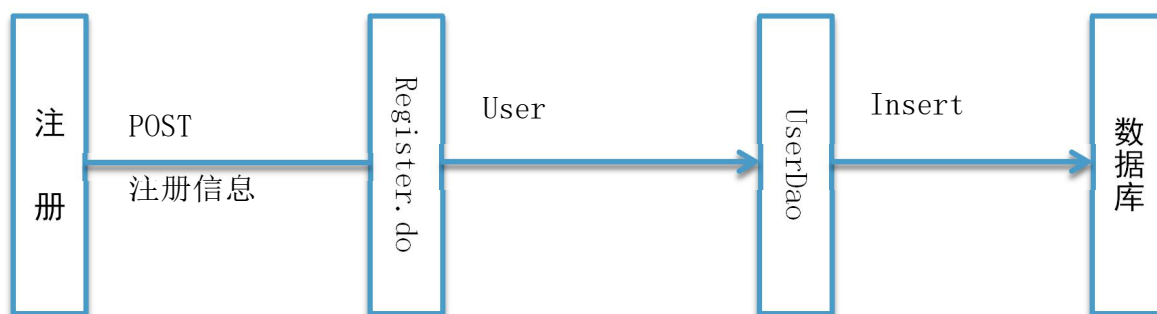


图 5-3 注册流程图

JSP 部分关键代码如下：

功能解释：通过 Form 表单将用户的注册信息以 POST 方法发送到 Register.do，并规定用户名必须为邮箱。

```

<form class="form-signin" action="register.do" method="post">
    <h2 class="form-signin-heading">用户注册</h2>
    <input type="text" name="name" class="form-control" placeholder="N
    ick name" required autofocus>
    <input type="email" name="mail" class="form-control" placeholder="
    Email address" required autofocus>
    <input type="password" name="password" class="form-control" placeh
    older="Password" required>
    <button class="btn btn-lg btn-primary btn-block" type="submit">注
    册</button>
</form>
  
```

Servlet 关键代码如下：

功能解释：将 Form 提交过来的信息从 Request 容器中取出，并 new 一个 User 对象通过 Set 方法改变它的属性，最后将整个 User 对象传到 UserDao 中，调用 insertBean 方法将注册的信息插入到数据库中。

```
public void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException , IOException{

    request.setCharacterEncoding("utf-8");
    response.setCharacterEncoding("utf-8");
    String mail = request.getParameter("mail");
    String name = request.getParameter("name");
    String password = request.getParameter("password");
    User user = new User(name, mail, password);
    System.out.println(user.toString());
    UserDao dao = new UserDao();
    dao.insertBean(user);
}
```

The image shows a user registration form titled "用户注册" (User Registration). It contains three input fields: "Nick name", "Email address", and "Password". Below these fields is a blue button labeled "注册" (Register). The form is centered on a light gray background.

图 5-4 注册页面

5.3.3 登陆设计与实现

1、登陆设计概述

登陆设计概述包括登陆设计的原理、流程。

当用户输入完用户名、密码和验证码后点击登录，ajax 使用 DOM 获取账号密码并以 post 方式提交到 Servlet，Servlet 调用业务层中的登录方法，业务层再调用数据访问层，然后在数据库中匹配是否存在此条数据，返回相应的结果到 Servlet 中，若用户名和密码存在则将获取到的用户存入 Session 中并传回到首页，若不存在此用户名和密码则未登录成功。

2、登陆实现

本网站学习评分系统是只对注册的学生用户服务，如果已经是本网站的用户，直接登录即可。登录在首页界面的右上方，在填写入正确的用户名、密码以及验证码以后方能登录。在制作用户登录部分，首先需要通过 request 来获取到登录界面中所输入的用户名，即文本框“acc”中的值，同理也需要取得用户输入的密码和验证码。如果登陆不成功，则返回登录页。

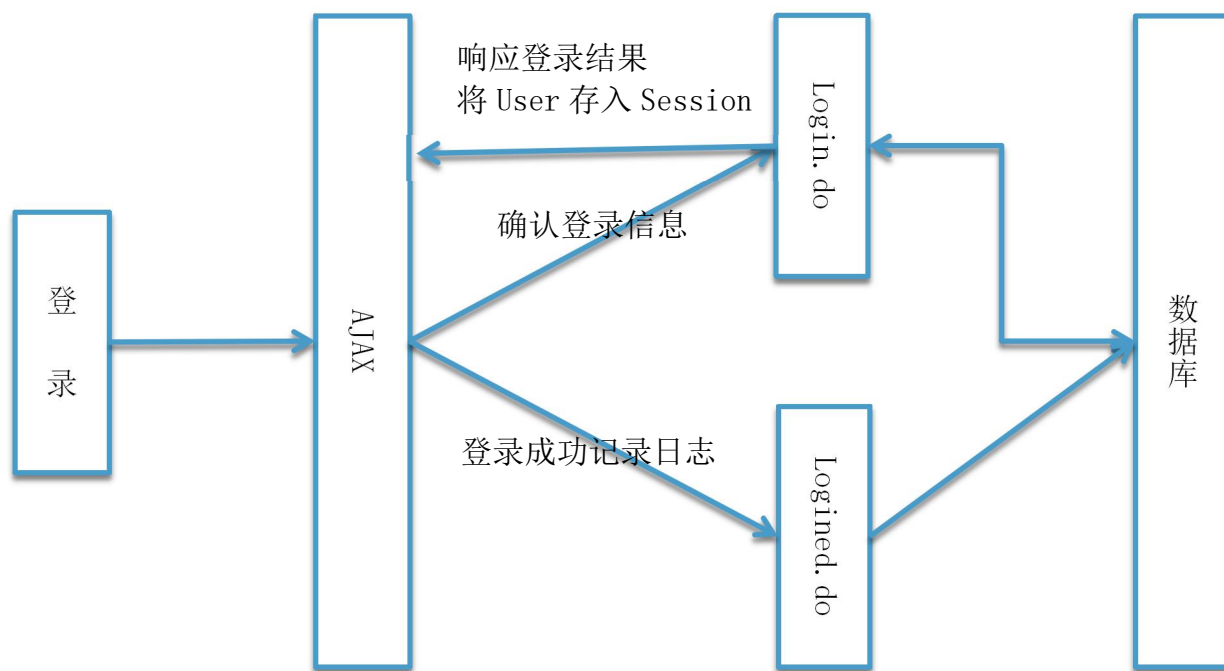


图 5-5 登录功能程序流程图

Jsp 部分代码：

功能解释：提供输入用户名密码的文本框以及登录按钮，位于首页的右上角。

```
<div id="loginform" class="navbar-form navbar-right">
```

```
    <div class="form-group">
```

```

        <input id="mail" name="mail" type="text" placeholder="Email" class
        ="form-control">
    </div>
    <div class="form-group">
        <input id="password" name="password" type="password" placeholder="
        Password" class="form-control">
    </div>
    <button id="loginSubmitButton" type="button" class="btn btn-succes
    s">登录</button>
</div>

```

AJAX 关键代码:

功能解释: 通过 DOM 中的方法, 来获取登录按钮的单击事件, 用 AJAX 来提交用户的登录信息、以及记录登录时的产生的日志信息。如果登录成功隐藏登录框显示用户昵称, 密码或账户错误提示登录失败。

```

$(document).ready(function() {
    $("#loginSubmitButton").click(function() {
        login();
    });
    log();
});
function login() {
    $.ajax({
        type:"post",
        url:"login.do",
        data:{
            name:$("#mail").val(),
            password:$("#password").val()
        },
        success:function (user) {

```

```

        if (user=="n") {
            alert("登录失败，请重新登录");
            window.location.href="login.jsp";
        }
        else{
            $("#loginform").hide();
            $("#loginSucceed").html("<p class='"+'" glyphicon glyphi
con-piggy-bank' "+'">"+"&nbsp;${user.nick_name    } "+'"</p>");
        }
    },
    error:function(aa) {
        alert("网络出错");
    }
});
}

```

Servlet 代码:

功能解释: 验证用户登录信息, 通过 response 响应给前台登录结果

```

String mail = request.getParameter("name");

String pw = request.getParameter("password");

UserDao dao = new UserDao();

User user = dao.loginDo(mail, pw);

HttpSession session = request.getSession();

if (user==null) {

    System.out.println("登录失败");

    response.getWriter().write("n");

}

else{

```

```
LoginLog log = new LoginLog(user.getUser_id(), user.getUser_
type());

LoginLogDao logdao = new LoginLogDao();
logdao.insertRecord(log);
session.setAttribute("user", user);
session.setAttribute("userid", user.getUser_id());
System.out.println("登录成功");
}
```

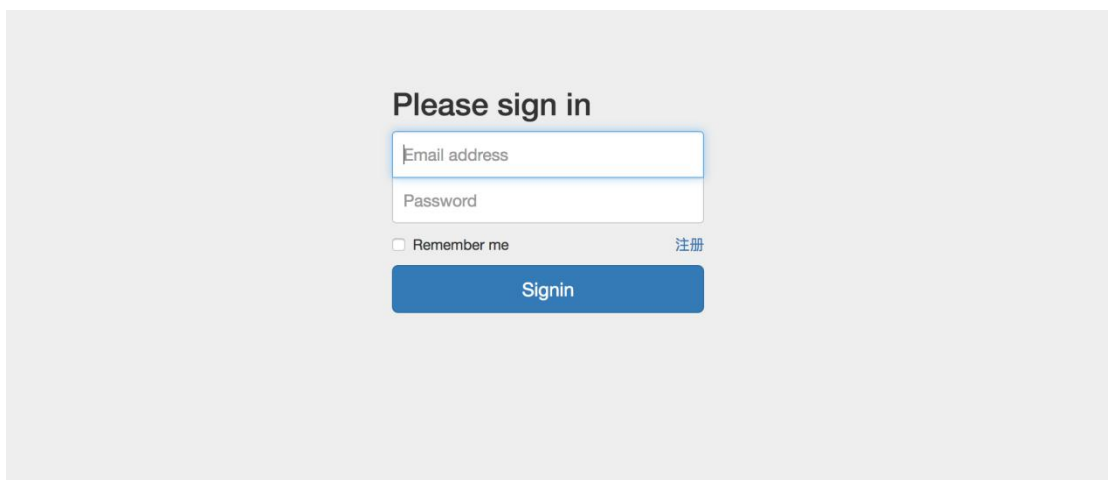


图 6-6 登录页面

5.3.4 专业分类设计与实现

1、专业分类设计概述

用户在首页点击导航栏中的分类链接会发出一个 get 请求，并传递一个参数 proId 到 Servlet，在 Servlet 中通过 HttpServlet 中的 getParameter() 方法获取到 Pro_id 的值，调用业务层中的查询主类方法，业务层再调用数据访问层，在数据访问层中执行数据库操作，再通过子类的 pro 查询出此子类下的所有视频，然后将其存放到 List 集合，再转发到分类页面的 jsp，通过 JSTL 和 EL 表达式调用两次 forEach 循环获取值并显示在页面中。

2、专业分类实现

获取分类列表的部分代码：

Servlet 部分关键代码：

功能解释：通过调用 ProDao 来获取整个专业分类列表

```
ProDao prodao = new ProDao();
List<Pro> pro = prodao.getAllList();
request.setAttribute("proList", pro);
```

ProDao 部分关键代码：

```
public List<Pro> getAllList() {
    String sql = "SELECT * FROM pro";
    return queryBeanList(sql, Pro.class);
}
```

5.3.5 课程列表页面设计与实现

1、课程列表页面设计概述

用户在首页点击进入学习页面会跳转到课程列表、专业列表页面并传递一个默认参数 proId，在此 jsp 页面中通过 EL 表达式获取链接中的 ProId，并在详细页面加载时发出一个 get 请求，并将获取到的 proid 传递到 Servlet，在 Servlet 中通过 HttpServlet 中的 getParameter() 方法获取到 ProId 的值，调用业务层中的查询课程列表信息方法，业务层再调用数据访问层，在数据访问层中执行数据库操作，将查询到课程列表信息存放到 List 集合中，采用重定向的方式传递到 jsp 中，依次获取出课程名称、课程介绍、专业分类等列表进行显示。

Servlet 部分关键代码：

功能解释：通过专业编号 (ProId) 来获取相对应的课程列表 (CourseList)

```
int pid = Integer.parseInt(request.getParameter("pro_id"));
ProDao prodao = new ProDao();
List<Pro> pro = prodao.getAllList();
request.setAttribute("proList", pro);
CourseDao coursedao = new CourseDao();
List<Course> course = coursedao.getBeanList(pid);
request.setAttribute("courseList", course);
request.setAttribute("pagename", prodao.getName(pid));
request.getRequestDispatcher("course.jsp").forward(request, response);
```


5.3.6 章节页面设计与实现

1、章节页面设计概述

用户在首页或分类页面点击具体视频会跳转到详细介绍页面并传递一个参数 `courseid`，在此 `jsp` 页面中通过 `EL` 表达式获取链接中的 `courseid`，并在详细页面加载时发出一个 `get` 请求，并将获取到的 `courseid` 传递到 `Servlet`，在 `Servlet` 中通过 `HttpServlet` 中的 `getParameter()` 方法获取到 `courseid` 的值，调用业务层中的查询视频详细信息方法，业务层再调用数据访问层，在数据访问层中执行数据库操作，将查询到详细信息存放到 `List` 集合中，采用请求转发的方式传递到 `jsp` 中，依次获取出章节名称、章节概要。

2、详细页面实现

获取页面详细信息部分代码：

`Servlet` 部分关键代码：

功能解释：通过课程编号（`CourseID`）来获取相对应的的章节列表（`SectionList`）

```
protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
    request.setCharacterEncoding("utf-8");
    response.setCharacterEncoding("utf-8");

    int pid = Integer.parseInt(request.getParameter("pro_id"));
    ProDao prodao = new ProDao();
    List<Pro> pro = prodao.getAllList();
    request.setAttribute("proList", pro);
    CourseDao coursedao = new CourseDao();
    List<Course> course = coursedao.getBeanList(pid);
    request.setAttribute("courseList", course);
    request.setAttribute("pagename", prodao.getName(pid));
    request.getRequestDispatcher("course.jsp").forward(request, response);
}
```



图 5-7 课程详细页面

5.4 本章小结

本章对网站前台的详细设计与实现进行了阐述，主要包括网站前台功能介绍，网站前台的系统流程图，以及对于网站各个模块的设计与实现的代码展示。

第六章 数据挖掘与机器学习详细设计与实施

6.1 数据收集功能详细设计与实施

6.1.1 在线视频学习视频播放页面的功能概述

视频播放页面提供给用户在线视频学习，通过课程详情页点击章节名称跳转到 Video.do 并将章节编号 ID 和视频编号通过 Get 方法传递参数，视频编号默认值为 1 表示播放章节中第一个视频，通过 VideoDao 来获取章节中的所有视频列表，将视频列表显示到视频播放页中，视频播放页中的上下章节按钮点击后会将章节编号 ID 增加或减少后再跳转到 Video.do 视频编号的值还为默认，视频列表的各个视频按钮点击后会保留原章节编号 ID 将视频编号 ID 增加或减少后跳转到 Video.do。其中播放器使用 H5 原生播放组件，只提供暂停播放功能，可以在进度条上拖动或跳跃视频进度。

学生视频播放页面与游客视频播放页面为同一个页面，视频播放页面需要显示视频章节名称、视频列表、上一章下一章按钮、一个播放控件。通过 VideoDao 获取视频列表，通过获取前一页面传来的章节名称都放入 request 容器，通过 Video.do 请求转发到视频播放页面进行视频播放与视频列表显示。视频播放页面如图 6-1 所示。

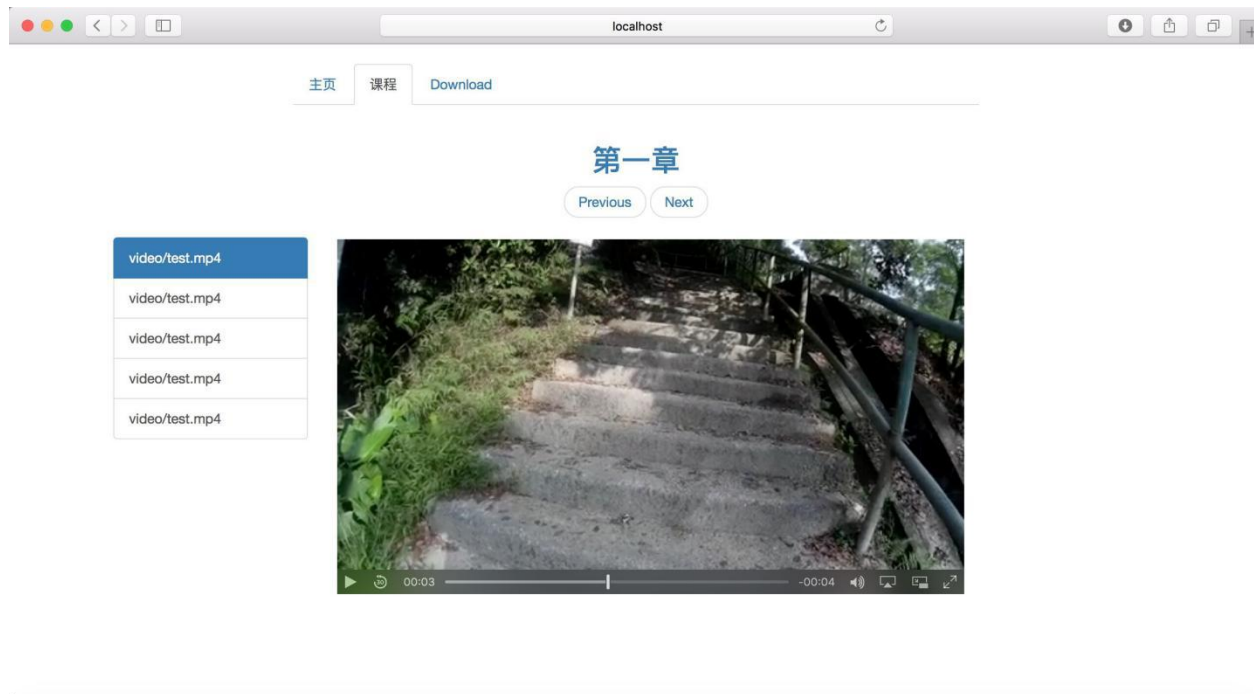


图 6-1 视频播放页面

Servlet 关键代码如下：

```
SectionDao sectiondao = new SectionDao();
```

```

int bid = Integer.parseInt(request.getParameter("section_id"));
int temp = bid - Integer.parseInt(request.getParameter("temp"));
int bidMax = sectiondao.getBidMax(bid);
if(temp==0) {
    temp=1;
}else if(temp == bidMax) {
    temp= bidMax;
}
System.out.println(temp);
Section section = sectiondao.getBean(temp);
request.setAttribute("section", section);
VideoDao videodao = new VideoDao();
List<Video> video = videodao.getBeanList(temp);
request.setAttribute("videoList", video);
request.setAttribute("video", videodao.getBean(Integer.parseInt(request.getParameter("video_id"))));
request.setAttribute("video_id", request.getParameter("video_id"));
request.getRequestDispatcher("video.jsp").forward(request, response);

```

JSP 关键代码如下:

```

<div class="container-fluid">
    <nav aria-label="...">
        <ul class="pager">
            <h2>
                <a href="video.do?section_id=${section.section_id}&temp=1&video_id=1">
                    ${section.section_name}
                </a>
            </h2>

```

```

        <li><a href="video.do?section_id=${section.section_id}&temp
=1&video_id=1">Previous</a></li>

        <li><a href="video.do?section_id=${section.section_id}&temp=
-1&video_id=1">Next</a></li>

    </ul>

</nav>

<div class="row">

    <div class="col-md-2 col-md-offset-1 list-group">

        <c:forEach items="${videoList}" var="i">

            <c:if test="${i.video_id==video.video_id}">

                <a href="video.do?section_id=${section.section_i
d }&temp=0&video_id=${i.video_id}" class="list-group-item active">

                    ${i.video_src }

                </a>

            </c:if>

            <c:if test="${i.video_id!=video.video_id}">

                <a href="video.do?section_id=${section.section_i
d }&temp=0&video_id=${i.video_id}" class="list-group-item">

                    ${i.video_src }

                </a>

            </c:if>

        </c:forEach>

    </div >

    <div class="col-md-7">

        <div style="background-color: @gray-darker">

            <video id="player" preload="none" controls width="640"
height="364" posterNew="img/1.jpg">

                <source id="myVideo" src="${video.video_src}" t
ype=' video/mp4' />

```

```

        </video>
    </div>
</div>
</div>
</div>

```

6.1.2 数据收集功能概述

数据收集在本次毕业设计中总体分为两大部分，其中有用户行为数据收集以及视频行为数据收集，通过数据收集收集结构化信息，建立合适的数学模型，将结构化的信息转化为半结构化或非结构化的信息提供给学习用户以及管理用户，最终达到可以帮助教师完成教学工作，提醒和促进学生的积极性。

用户的浏览的页面（不包括视频播放页），这些页面的数据收集主要考虑的是用户活跃度 R 、用户粘性 F 、用户的产出 M ，其中 R 主要体现在了用户的访问间隔， F 为用访问次数、本文认为学习者的产出等同于学习消耗的时长所以 M 体现在用户的访问时长，其中时长因素包括：页面停留时间、视频播放时间、挂机类无效时间。视频行为数据收集页面通过嵌套在视频播放页面中来挖掘用户行为、视频行为，用户通过视频播放进行在线学习，整个视频播放页面通过 DOM 监听窗口打开关闭、视频播放暂停、视频播放结束等等。

总体的目标就是收集一些平时大量产出且平时没有很关注的结构化信息用作产生半结构化或非结构化的信息的关键依据，最终给出一些智能给出一些决策建议。

6.1.2 用户行为数据收集

1、用户活跃度

活跃用户，是相对于“流失用户”的一个概念，是指那些会时不时地光顾下网站，并为网站带来一些价值的用户。流失用户，是指那些曾经访问过网站或注册过的用户，但由于对网站渐渐失去兴趣后逐渐远离网站，进而彻底脱离网站的那批用户。活跃用户用于衡量网站的运营现状，而流失用户则用于分析网站是否存在被淘汰的风险，以及网站是否有能力留住新用户。本次论文中对于用户的活跃度的记录主要体现到用户登录距查看日期的时长即访问间隔。用户登录记录关键代码如下：

JavaBean 关键代码如下：

```

public LoginLog( int user_id, int type) {
    super();
}

```

```

        this.system_time = new Date();

        this.login_id = 0;

        this.user_id = user_id;

        this.type = type;

    }

```

JavaDao 关键代码如下：

```

public Date getMaxDate(int id) {

    String sql = "SELECT max(system_time) FROM login_log WHERE user_id=?";

    return (Date) queryMap(sql, id).get("max(system_time)");

}

```

Servlet 关键代码如下：

```

LoginLog log = new LoginLog(user.getUser_id(), user.getUser_type());

LoginLogDao logdao = new LoginLogDao();

logdao.insertRecord(log);

```

2、用户黏性

用户粘性在本次课题中，用户的粘性主要体现为用户的访问频率，对于访问频率的收集主要针对用户对一个页面一段时间内的访问次数，每次访问网站都会记录用户的浏览记录。记录用户访问频率的关键代码如下：

AJAX 关键代码如下：

```

$.ajax({

    type:"post",

        url:"watched.do",

        data:{

            session_id:"<%=session.getAttribute("userid")%>",

            page_name:"首页",

            type:"1"

        },

    });

```

Servlet 关键代码如下：

```
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException{
    int id = Integer.parseInt(request.getParameter("session_id"));
    String pname = request.getParameter("page_name");
    int ty = Integer.parseInt(request.getParameter("type"));
    System.out.println(id);
    WatchPageLog watchpagelog = new WatchPageLog(id, pname, ty);
    WatchPageLogDao watchpagelogdao = new WatchPageLogDao();
    watchpagelogdao.insertBean(watchpagelog);
}
```

3、用户产出

在线学习的产出本文认为就是学习的成功，学会了课程内容便认为是用户的产出，用户的产出越高，他的的学习成果也就越高，同时用户的学习代价就越大，消耗的时间也就越多，这里也就得出了模型中第三个关键影响因素 M 视频观看时长。记录观看视频时长的关键代码如下：

AJAX 关键关键：

```
var time = video.currentTime;
$.ajax({
    type : "post",
    url : "watchvideo.do",
    data : {
        video_time:time,
        user_id:<%=session.getAttribute("userid")%>,
        video_id:"<%=request.getAttribute("video_id")%>",
        type:"1"
    });
```

Servlet 关键代码


```
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException , IOException{

    int vid = Integer.parseInt(request.getParameter("video_id"));
    int uid = Integer.parseInt(request.getParameter("user_id"));
    float vtime = Float.parseFloat(request.getParameter("video_time"));
    int ty = Integer.parseInt(request.getParameter("type"));
    WatchVideoLog watchvideolog = new WatchVideoLog(uid, vid, vtime, ty);
    WatchVideoLogDao watchvideologdao = new WatchVideoLogDao();
    watchvideologdao.insertBean(watchvideolog);

}
```

6.1.3 视频行为数据收集

视频播放和暂停通过 JavaScript 监听播放器的两个行为事件，其中播放开始时间为 0 和播放完成结束时间为视频总长度，这里两个时间产生的事件是单独存储的，播放次数会在播放视频开始后 3 分钟后记录一次，播放完成次数会在播放到总长度 95%以上且停留播放页面时间超过视频长度 80%以上记录一次。快进和后退两种事件产生也会产生暂停和播放时间，则将快进后退收集方法记录为暂停和播放的时间差。关键代码如下：

AJAX 关键代码如下

```
var video = document.getElementById('player');
video.addEventListener("play",function() {

    var time = video.currentTime;

    if(video.currentTime==0) {

        $.ajax({

            type:"post",
            url:"videocount.do",
            data:{

                type:1,
                video_id:"<%=request.getAttribute("video_id")%>"

            },

        });

    }
```

```
    }else{
        $.ajax({
            type:"post",
            url:"videobehavior.do",
            data:{
                video_id:"<%=request.getAttribute("video_id")%>",
                type : 1,
                video_time : time,
                speed : 1
            },
        });
    }
});

video.addEventListener("ended",function() {
    var time = video.currentTime;
    $.ajax({
        type:"post",
        url:"videobehavior.do",
        data:{
            video_id:"<%=request.getAttribute("video_id")%>",
            type : 3,
            video_time : time,
            speed : 1
        },
    });

    $.ajax({
        type:"post",
        url:"videocount.do",
```

```

        data:{
            type:1,
            video_id:"<%=request.getAttribute("video_id")%>"
        },
    }
});
});

```

Servlet 关键代码:

```

protected void doPost(HttpServletRequest request ,HttpServletResponse response)
throws ServletException ,IOException{
    int vid = Integer.parseInt(request.getParameter("video_id"));
    int bty = Integer.parseInt(request.getParameter("type"));
    float bvtime = Float.parseFloat(request.getParameter("video_time"));
    VideoBehavior videobehavior = new VideoBehavior(vid, bty, bvtime);
    VideoBehaviorDao behaviordao = new VideoBehaviorDao();
    behaviordao.insertBean(videobehavior);
}

VideoDao videodao = new VideoDao();
int ty = Integer.parseInt(request.getParameter("type"));
if(ty == 1){
    videodao.upPlayCount(Integer.parseInt(request.getParameter("video_id")));
}else{
    videodao.upEndedCount(Integer.parseInt(request.getParameter("video_id")));
}

```

6.3 机器学习功能详细设计与实施

6.3.1 机器学习概述

机器学习是一门多领域交叉学科，涉及概率论、统计学、逼近论、凸分析、算法复杂度理论等多门学科。专门研究计算机怎样模拟或实现人类的学习行为，以获取新的知识或技能，重新组织已有的知识结构使之不断改善自身的性能。^[9]

在线视频学习收集到了用户在线观看时的行为信息以及视频播放时视频行为信息，这些信息大多都为结构化信息，通过设计算法让计算机自己学习收集用户行为信息，视频行为信息最终将结构化信息转化为半结构化信息或非结构化的信息，实现数据挖掘。

6.3.2 RFM 模型、AHP 模型以及 K-均值聚类的应用

1、模型的概述

根据美国数据库营销研究所 Arthur Hughes 的研究，客户数据库中有三个神奇的要素，这三个要素构成了数据分析最好的指标：最近一次消费 (Recency)、消费频率 (Frequency)、消费金额 (Monetary)。在网站的计算模型中 R 因素为：最近一次登录，F 因素为：播放浏览频率，M 因素为：学习时长。理论上，上一次登录时间越近的用户应该比较好的学生，对提供即时的在线学习任务或是课题项目也最有可能会有反应。学习频率是用户在限定的期间内所在线视频学习的次数。我们可以说最常学习的用户，也是满意度最高的用户。如果相信课程、教师忠诚度的话，最长学习的用户，忠诚度也就最高。^[10]

层次分析法 (AHP)，在 20 世纪 70 年代中期由美国运筹学家托马斯·塞蒂 (T. L. Saaty) 正式提出。它是一种定性和定量相结合的、系统化、层次化的分析方法。由于它在处理复杂的决策问题上的实用性和有效性，很快在世界范围得到重视。它的应用已遍及经济计划和管理、能源政策和分配、行为科学、军事指挥、运输、农业、教育、人才、医疗和环境等领域。^[11]

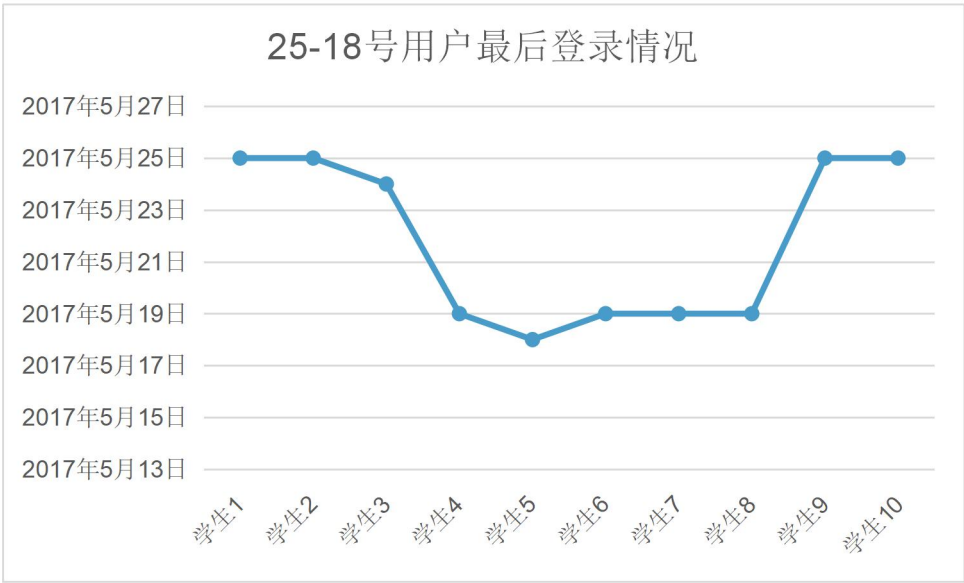
1、模型假设

网站冷启动阶段没有参考依据需要通过人工大量的模拟学习过程为网站添加数据，在此之前建立 10 个样本数据设想预期结果，将样本的特征扩大用作突出学习行为结果信息，样本分别为学生 1、学生 2..... 学生 10，其中样本数据为自拟数据。详细样本如下表图 6-1、图表 6-2、图表 6-3。

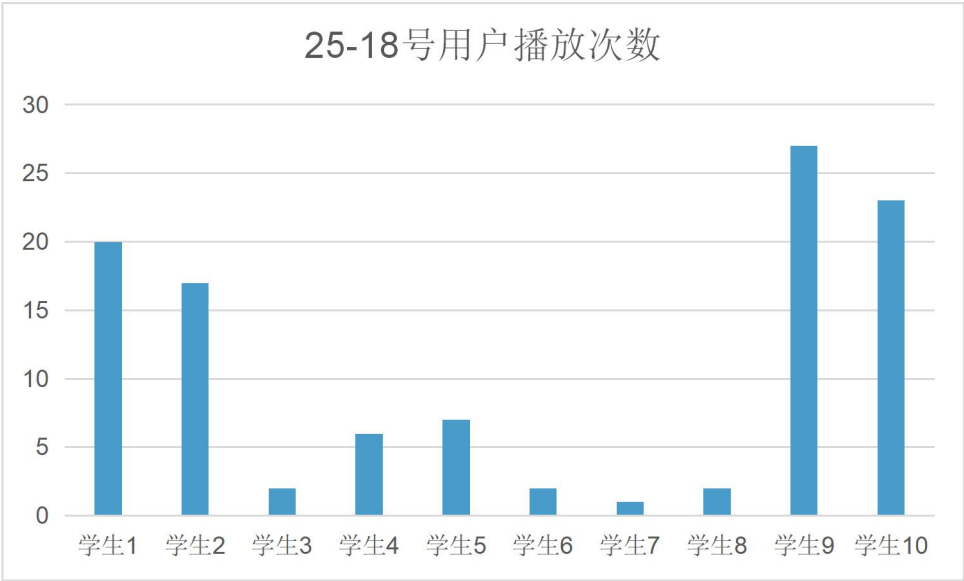
^[9] 机器学习-百度百科【EB/OL】

^[10] RFM 模型 - MBA 智库百科【EB/OL】

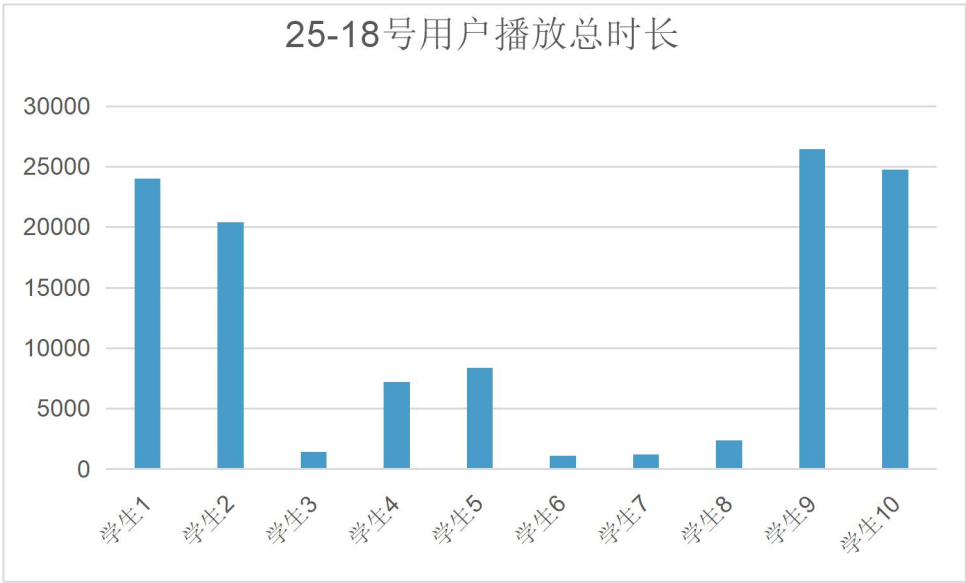
^[11] 层次分析法 - MBA 智库百科【EB/OL】



图表 6-1 最近登录情况



图表 6-2 用户播放次数



图表 6-3 用户播放总时长

2、模型的建立

针对样本提出一种基于改进 RFM 模型的客户分类方法。应用层次分析法来确定 RFM 模型中每个变量的权重,在此基础上,应用 K 均值聚类法来对客户进行分类,之后分析每一类客户的行为特征和价值,并且对不同的顾客类别采取不同的策略。

客户分类方法主要有基于顾客利润率的分类和基于指标组合的客户分类方法。RFM 模型经常使用的三个指标是近度 (Recency)、频度 (Frequency)、值度 (Monetary)。^[12]从用户学习成果角度来考虑在线视频学习用户的 RFM 模型,改进后的 RFM 指标与传统的 RFM 以及参考电信用户 RFM 模型指标含义比较如下表 6-4 所示

表 6-1 传统的 RFM 模型、电信业 RFM 模型与自身的各指标含义比较

模型	R (近度)	F (频度)	M (值度)
传统的 RFM 模型	客户最近一购买距离分析点的时间	客户一定时期内购买该企业产品的次数	客户一定时期内购买该企业产品的总金额
电信业 RFM 模型	客户最后一次交费距离分析点的时间	客户一定时期内交费次数	客户一定时期内的交费总金额
在线视频学习网站 RFM 模型	用户最后一次访问距离分析点的时间	用户一定时期内播放浏览次数	用户一定时期内总浏览和播放的时长

^[12] Stone,Bob.Successful Direct Marketing Methods【M】.Linco Inwood:NTC Bussiness Books,1994

在线视频学习网站的影响因素与传统 RFM 模型、电信用户 RFM 模型影响因素并不相同，本文以用户登录和浏览最后时间、用户一定时期播放次数、用户播放总时间长度来替代客户消费的近度、频度和值度。

3、RFM 权重的分析

对 RFM 各变量的指标权重问题, Hughes, Arthur 认为 RFM 在衡量一个问题上的权重是一致的, 因而并没有给予不同的划分。而 Stone, Bob 通过对信用卡实证分析, 认为各个指标的权重并不相同, 应该给予频度最高, 近度次之, 值度最低的权重^[13], 认为针对不同的行业甚至不同的公司, 频度、近度、值度的权重均存在一定差异, 因此需要采用科学的方法进行分析。对此, 以层次分析法为支撑, 结合专家咨询方式来解决指标权重的确定问题。

在线视频学习网站冷启动, 没有同电信公司对用户的数据的收集, 为了权重的可靠性在样本设计时, 以及按照现实情况进行设计, 对于样本的反映的结果以及有了答案, 通过权重的测试来匹配一个合适的权重用作冷启动时的权重, 网站经过用户浏览一段时间后, 达到热启动可以再次进行修改权重并同相关教师或学生咨询来解决指标权重的确定问题。

根据智库百科对 RFM 模型的解释看到对电信用户权重做出的一个评价矩阵模型后, 尝试对在线学习网站初期的样本计算出一个类型的评价矩阵如表 6-5 电信用户评价矩阵与如表 6-6 在线网站用户评价矩阵。

表 6-2 电信用户评价矩阵

	R	F	M
R	1	0.71	0.46
F	1.41	1	0.085
M	2.18	1.18	1

表 6-3 IMAUM00C 用户评价矩阵

	R	F	M
R	1	0.85	0.61
F	1.17	1	0.71
M	1.64	1.40	1

$R < 0.1$, 表明该判断矩阵的一致性在接受范围内。由上表得出 RFM 各指标相对权重为 $[WR, WF, WM] = [0.221, 0.341, 0.439]$ 。其中 M 的权重最大, 即认为学习总时长的高低是影响学习成果高低的最主要因素。

4、基于 K-均值聚类用户学习态度分类

^[13] 连惟谦. 应用资料分析技术进行顾客流失与顾客价值之研究【M】. 台湾: 中原大学. 2004

K 均值聚类算法是先随机选取 K 个对象作为初始的聚类中心。然后计算每个对象与各个种子聚类中心之间的距离，把每个对象分配给距离它最近的聚类中心。聚类中心以及分配给它们的对象就代表一个聚类。一旦全部对象都被分配了，每个聚类的聚类中心会根据聚类中现有的对象被重新计算。这个过程将不断重复直到满足某个终止条件。终止条件可以是没有（或最小数目）对象被重新分配给不同的聚类，没有（或最小数目）聚类中心再发生变化，误差平方和局部最小。^[14]

基于 K-均值聚类法的用户分类过程应用 K-均值聚类法^[15]，用加权后的 RFM 三项指标，将具有相近的用户学习态度得分进行分类，基本思路如下：

(1) 将 RFM 三项指标标准化，降低分布上差异过大。

(2) 应用层次分析 AHP 确定 RFM 各个指标的所占权重，并将各个指标加权。

(3) 确定 K-均值聚类的中心点数量。

(4) 通过加权后的对指标进行聚类，得到 x 类学习用户。

(5) 将每类学习用户的 RFM 各项平均值和总 RFM 各项平均值作比较，每次对比会产生有两个结果：大于平均值和小于（等于）平均值，通过比较得出每类学习用户 RFM 的变动情况。

(6) 根据每种学习用户的 RFM 各项的变动情况分析该学习用户的性质，如该学习用户是倾向于热情的还是倾向于消极的，最终在此基础得出一个半结构化或非结构化的学习类型信息。

(7) 对每个学习用户 RFM 三项指标标准化后，将平均值加权求和，得到每个学习用户的学习态度总得分，分析各个学习用户的学习行为。

5、学习用户类型识别分析

自拟 10 个样本数据于 5 月 18 号至 5 月 25 日的用户行为记录进行分析，样本的描述性统计见下表 6-7：

表 6-4 学习用户 RFM 各项数值描述性统计

指标	最小值	最大值	平均值	标准差
近度	0	7	3.5	3.324
频度	1	27	10.7	9.487

^[14] K 均值聚类-百度百科【EB/OL】

^[15] 张文彤，董伟. SPSS 统计分析高级教程【M】. 北京：高等教育出版社，2004. 247-248

值度	1100	26460	11734.5	10302.61
----	------	-------	---------	----------

由于 RFM 各项数据的量纲各不相同，各项数据的取值差异也很大。为了消除分布上差异较大和量纲度量不同的影响,在对 RFM 三项指标进行加权之前,需要进行数据标准化处理。其中 FM 对学习结果存在正相关的影响，因此标准化调整公式为：

$x' = (x - x^s) / (x^l - x^s)$ ，因素 R 对学习结果存在负相关的影响，因此标准化调整公式考虑为： $x' = (x^l - x) / (x^l - x^s)$ ，其中 x 为原数据值， x' 为标准化后的数据值， x^l 为原数据中最大值， x^s 为原数据中最小值。

在使用 K-均值聚类法时,首先需要判断其聚类的中心点个数 x 。在模型中用户类别通过每个学习用户的 RFM 各项平均值与总 RFM 各项平均值相比较来产生的,而单个指标的比较只能有两种情况:大于或小于（等于）平均值,因此可能有 $2^x(x=3)$ 种类别。

标准化和确定聚类中心点个数后,进行聚类分析,得到 8 种学习用户分类。将 8 种学习用户分类的 RFM 各项平均值与总 RFM 各项均值比较。其中高于平均水平用“↑”，低于平均或等于平均水平用“↓”表示。如图表 6-8 所示：

表 6-5 聚类后产生的 8 中学习用户分类

客户类别	客户数量	近度 / d	频度	值度 / 元	比较结果	客户级别
1	4	0	21.75	23906	$R \uparrow F \uparrow M \uparrow$	重点了解,与学习成绩做比较
2	0	无	无	无	$R \uparrow F \uparrow M \downarrow$	一般重点了解，普通学生
3	0	无	无	无	$R \uparrow F \downarrow M \uparrow$	重点发展，与学生课程做比较
4	1	1	2	1420	$R \uparrow F \downarrow M \downarrow$	一般了解，学习兴趣不高
5	0	无	无	无	$R \downarrow F \uparrow M \uparrow$	重点保持，与学生课程做比较
6	0	无	无	无	$R \downarrow F \uparrow M \downarrow$	一般了解，兴趣较低
7	0	无	无	无	$R \downarrow F \downarrow M \uparrow$	重点了解，与学习能力比较
8	5	6.8	3.6	4060	$R \downarrow F \downarrow M \downarrow$	无价值记录，了解优先级最低
总均值		11.05	12.72	1565		

通过 RFM 分析将学习用户群体划分成重点了解、一般重点了解、重点发展、重点保持、一般了解、无价值记录六个级别,各学习用户簇的用户级别如表 6-8 所示。学习用户分类不仅揭示了学习用户在类别上的差异,而且反映了学习用户在学习行为上的特性和变化

倾向。通过 RFM 各项指标分析可将现有学习用户划分为不同的学习用户等级,针对不同等级的学习用户,采取不同的教育和指导方式。这些用户学习行为等级信息以及属于半结构化或非结构化的信息了。但是,这种类别只是确定了学习用户学习的等级,却没有各类学习用户之间的一个量化的价值比较,因而对各类学习用户做相应的学习态度分析是非常有必要的。

6、学习用户学习态度比较分析

表 6-8 将学习用户簇 1 和簇 7 同分为重点了解用户,将学习用户簇 4 和簇 6 同分为一般了解用户,这样难以对这两组学习用户簇进行细分.此外,学习用户分类后,并不知道每一类学习用户的价值差别有多大,相对学习成果,怎样利用 AHP 层次分析得到的 RFM 各项指标权重,结合每类学习用户的 RFM 各项指标,根据每一类学习用户的学习态度得分来进行排序.将数据标准化处理后的各个指标平均值如表 6-9 的 C_R^j, C_F^j, C_M^j 其中 $j=1..m$ 表示聚类后的聚类中心的类别。 C_R^j, C_F^j, C_M^j 第 j 类学习用户的 R, F, M 各个指标标准化后的平均值, C_I^j 是第 j 类学习用户的 R, F, M 各项指标加权后的总评分,其运算公式为:

$$C_I^j = W_R C_R^j + W_F C_F^j + W_M C_M^j$$

其中, W_R, W_F, W_M 分别为由 AHP 层次分析得来的 R, F, M 指标的权重,最后根据总学习态度得分的大小来对各个学习用户来进行排序(见表 4).排名靠前的学习用户相对排名靠后的学习用户对视频学习的学习态度和课程内容的忠诚度更高,对于学习成果。表 6-9 显示,学习用户簇 9 总得分最高,因此簇 9 的客户是学习态度最好学习成果相对最高,而簇 7 学习用户总得分最低,因此可以认为簇 7 学习用户的学习态度最差学习成果相对最后。

表 6-9 学习态度评分

学习用户	C_R^j	C_F^j	C_M^j	C_I^j	排序
学生 1	0.221	0.249192308	0.424808789	0.895001097	2
学生 2	0.221	0.209846154	0.358026622	0.788872775	3
学生 3	0.189428571	0.013115385	0.005936193	0.208480149	6
学生 4	0	0.065576923	0.113158673	0.178735596	7
学生 5	0.031571429	0.078692308	0.135419396	0.245683132	5

学生 6	0	0.013115385	0	0.013115385	9
学生 7	0	0	0.00185506	0.00185506	10
学生 8	0	0.013115385	0.024115783	0.037231167	8
学生 9	0.221	0.341	0.470443271	1.032443271	1
学生 10	0.221	0.288538462	0.031443271	0.540981732	4

注: $C_I^j = W_R C_R^j + W_F C_F^j + W_M C_M^j (W_R = 0.221, W_F = 0.341, W_M = 0.439)$

在进行学习用户分类后再对学习用户进行学习行为态度排序,使得教师能够量化各学生学习行为态度的差别,弥补了的学习用户分类方法的不足。这有助于教师制定更为可行的教学方案。

6.3.2 算法设计与实现

1、复杂事物抽象化

整个算法的设计思路将 RFM 模型抽象一个实例,通过设置成员属性 R、F、M 以及公共的属性均值 R、均值 F、均值 M 来处理 RFM 模型中的各个指标值。关键代码如下:

```
public class StudentCji {
    private int id;
    private String name;
    private double cjR;
    private double cjF;
    private double cjM;
    private double cJI;
    private static double meanCJR;
    private static double meanCJF;
    private static double meanCJM;
    private static double meanCJI;
    private static double sumCJR;
    private static double sumCJF;
    private static double sumCJM;
    private static double sumCJI;
    public StudentCji(int id, double cjR, double cjF, double cjM) {
        super();
        UserDao dao = new UserDao();
        this.name=dao.getStudent(id).getNick_name();
        this.id = id;
        this.cjR = cjR;
        this.cjF = cjF;
        this.cjM = cjM;
        this.cJI = this.cjR*0.221+this.cjF*0.341+this.cjM*0.439;
    }
}
```

2、时间量化

在通过前端收集到的用户行为、视频属性、视频行为中日期和时间的量化是一个关键处理部分，通过日期的加减，强制类型转换与数据库做存取也是一个难点。关键代码如下：

```

:
public class CountDateTime {
    public Map<String, Object> count(int id){
        float timecount=0;
        float timecount2=0;
        float timemax=0;
        float time = 0;
        float count = 0;
        WatchPageLogDao pagedao = new WatchPageLogDao();
        List<Map<String, Object>> ld = pagedao.getBeanDate(id);
        for (int i = 0; i < ld.size()-1; i++) {
            if((int)(ld.get(i).get("type"))==1){
                if((int)(ld.get(i+1).get("type"))==0){
                    float a
=((Date)(ld.get(i+1).get("system_time"))).getTime();
                    float b
=((Date)(ld.get(i).get("system_time"))).getTime();
                    if(timecount+(a-b)/1000<600&&timecount+(a-
b)/1000>0){去奇异
                        timecount = timecount+(a-b)/1000;
                        i++;
                    }
                }
            }
        }
        WatchVideoLogDao watchvideoLogdao = new WatchVideoLogDao();
        List<Map<String, Object>> ld2 = watchvideoLogdao.getBeanListMap(id);
        for (int i = 0; i < ld2.size()-1; i++) {
            float temp = ((Date)ld2.get(i).get("system_time")).getTime();
            //查询观看次数间隔 5 秒有效
            if(i==1){
                time = (long) temp;
                count=1;
            }else {
                if((temp-time)/(1000*5)>1){
                    time = (long) temp;
                    count++;
                }
            }
            if(temp/1000>timemax){
                timemax = (long) (temp/1000);
            }
            if((int)ld2.get(i).get("type")==1){

```

```

        if((int)ld2.get(i+1).get("type")==0){
            System.out.println(i+"当前是几");
            float temp2 =
((Date)ld2.get(i+1).get("system_time")).getTime();
            System.out.println(temp2+"temp2");
            float temp3 = (temp2-temp)/1000;

            System.out.println(((float)ld2.get(i+1).get("video_time"))+"temp3");
            if
((temp3>3)||((float)ld2.get(i+1).get("video_time"))>3) {
                if
((float)ld2.get(i+1).get("video_time")<temp3) {

                    timecount2 = timecount2 + temp3;
                }else{
                    timecount2 = timecount2 +
(float)ld2.get(i+1).get("video_time");
                }
                i++;
            }else{
                System.out.println("未超过 3 秒");
            }
            continue;
        }
    }
}

Map<String, Object> map = new HashMap();
map.put("timecount", timecount+timecount2);
map.put("count", count);
map.put("timemax", timemax);
return map;
}
}

```

3、算法的实现

算法实现在构造方法、静态成员变量的 Set 方法中通过生成实例时就为各个指标进行加权处理，通过 Set 方法来计算各个指标的均值。关键代码如下：

```

public static void addSumCJR(double sumCJR) {
    StudentCji.sumCJR += sumCJR;
}

public static void addSumCJF(double sumCJF) {
    StudentCji.sumCJF += sumCJF;
}

public static void addSumCJM(double sumCJM) {
    StudentCji.sumCJM += sumCJM;
}

public static void addSumCJI(double sumCJM) {

```

```

        StudentCji.sumCJI += sumCJI;
    }
    public static void addMean(int i) {
        StudentCji.meanCJR = StudentCji.sumCJR/i;
        StudentCji.meanCJF = StudentCji.sumCJF/i;
        StudentCji.meanCJM = StudentCji.sumCJM/i;
        StudentCji.meanCJI = StudentCji.sumCJI/i;
    }
}

```

6.4 数据可视化功能详细设计与实施

数据可视化通过利用 Echarts 框架在 JSP 中实现了实时的一个动态图表，来显示 RFM 模型的结果。显示结果如图

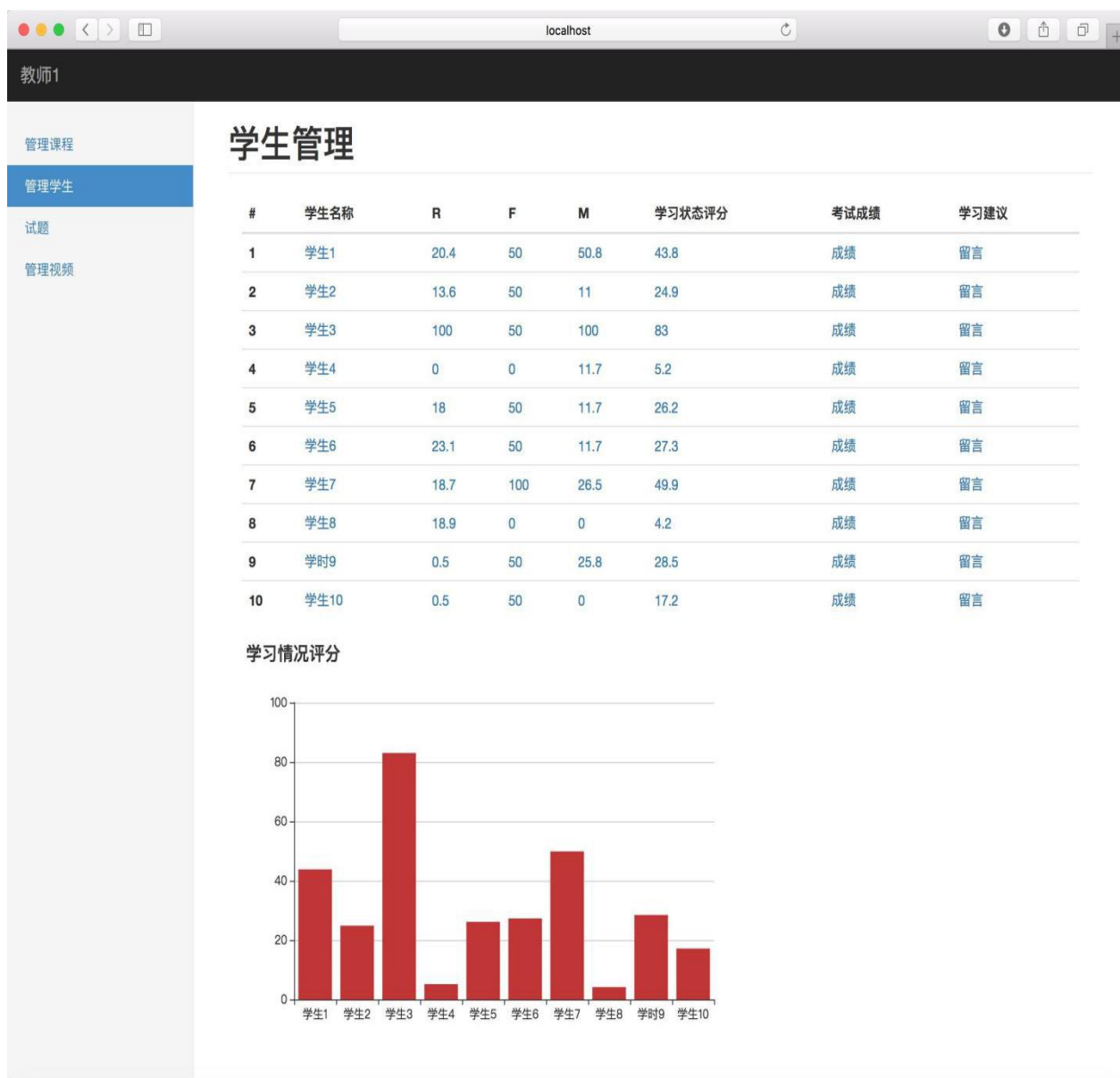


图 6-2 数据可视化

关键代码如下：

```
<script type="text/javascript">
    // 基于准备好的 dom，初始化 echarts 实例
    var myChart = echarts.init(document.getElementById('main'));
    // 指定图表的配置项和数据
    var option = {
        title: {
            text: '学习情况评分'
        },
        tooltip: {},
        legend: {
            data: ['学习情况']
        },
        xAxis: { // "学生 1", "学生 7", "学生 3", "学生 5", "学生 2", "学时 9", "学生
8", "学生 10", "学生 6", "学生"
            data: [{names} ]
        },
        yAxis: {},
        series: [{
            name: '情况',
            type: 'bar',
            data: [{scores}] // 78.9, 76.3, 59.2, 48.2, 42.2, 39.9, 22.1,
17.7, 10.2, 10.1]
        }]
    };
    myChart.setOption(option);
</script>
```

6.5 本章小结

本章介绍了数据挖掘与机器学习详细设计思路，从数据收集将一些结构化的信息收集到数据库中，通过编写程序让计算机自己学习收集数据，通过 RFM、AHP、K-均值聚类等算法将结构化的数据信息转化为管理层所需要的半结构化或非机构化信息，最终实现数据挖掘与机器学习。

第七章 网站后台详细设计与实施

7.1 网站后台功能概述

本课题设计与实现在线视频学习网站“IMAU MOOC”，网站后台是教师管理“IMAU MOOC”网站课程、浏览学生学习情况，后台分为登录、课程浏览、课程章节删除、视频上传、搜学生学习情况浏览等。

7.2 网站后台系统流程图

后台系统流程图如图 7-1 所示。

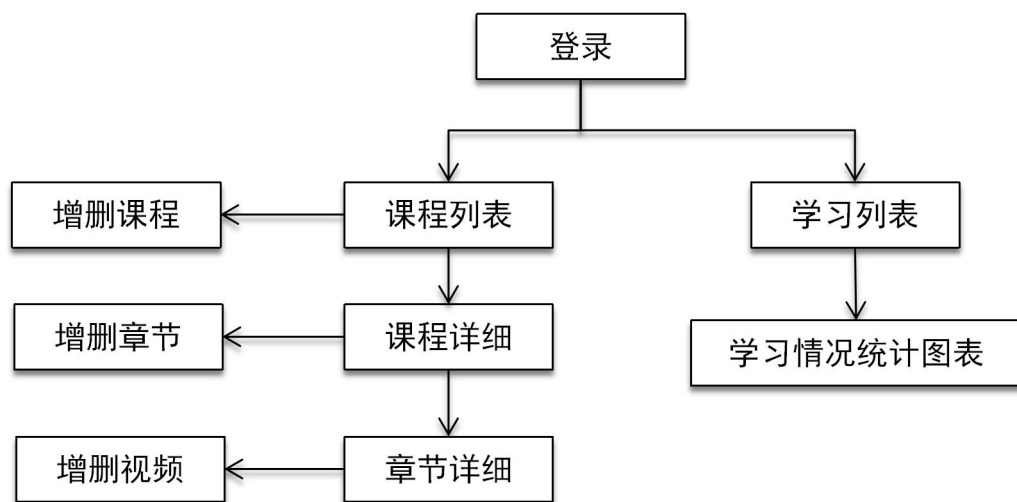


图 7-1 后台系统流程图

7.3 网站后台各个功能呢个模块详细设计与实现

7.3.1 教师课程显示设计与实现

1、课程显示概述

课程显示设计概述包括设计的原理、流程。

教师登录显示的首页为课程列表显示页面，此页面中显示教师的各个课程，点击课程名称会跳转到章节列表页面显示章节列表，点击章节名称会跳到章节详细页面显示视频列表页面，每个详情页面都可以修改对应的详细信息，视频详细页面可以上传视频。

2、课程显示实现

课程列表页面：显示教师课程列表信息，可以增加删除课程。

功能解释：通过 section 中的 User 取出教师 Id 并通过 Get 请求来获取 Request 容器中的教师课程列表，通过 TeacherServlet 来调用 TeacherDao 来获取相对应的教师课程列表，并放到 Request 容器中，通过请求转发到教师首页使用 JSTL 来逐条显示。

Servlet 关键代码如下：

```

int c_id = Integer.parseInt(request.getParameter("course_id"));

SectionDao sectiondao = new SectionDao();

List<Section> section = sectiondao.getBeanList(c_id);

CourseDao coursedao = new CourseDao();

Course course = coursedao.getBean(c_id);

ProDao prodao = new ProDao();

String pname = prodao.getName(course.getPro_id());

request.setAttribute("pro_name", pname);

request.setAttribute("course", course);

request.setAttribute("sectionList", section);

request.getRequestDispatcher("teachercourse.jsp").forward(request,
response);

```

JSP 部分关键代码如下:

```

<div class="container-fluid">
    <c:forEach items="${courseList}" var="i">
        <div class="jumbotron">
            <h1>
                <a href="teachercourse.do?course_id=${i.course_id}">
                    ${i.course_name }
                </a>
            </h1>
            <p>${i.course_note }</p>
        </div>
    </c:forEach>
</div>

```

教师课程列表信息页面如图 7-2 所示:



图 7-2 教师课程列表

3、课程修改实现

JSP 部分关键代码如下：

```
<div class="page-header">

    <div class="alert alert-success" role="alert">

        <form action="coursechange.do">

            <div class="row">

                <div class="col-lg-12">

                    <div class="input-group">

                        <span class="input-group-btn">

                            <button class="btn btn-default" type="button">
                                修改</button>

                        </span>

                        <input name="course_name" id="course_name" type="text"
                            class="form-control"

value="\${course.course_name}"
```

```

placeholder="课程名">

</div>

</div>

<div class="col-lg-12">
    <div class="input-group">
        <span class="input-group-btn">
            <button class="btn btn-default" type="button">
                修改</button>
            </span>
            <input name="pro_name" type="text" class="form-control"
                value="{pro_name}"
placeholder="专业分类">
        </div>
    </div>
</form>
</div>
</div>

```

Servlet 关键代码如下:

```

String cname = request.getParameter("course_name");
String pname = request.getParameter("pro_name");
String cnote = request.getParameter("course_note");
String cimg = request.getParameter("course_img");
int period = Integer.parseInt(request.getParameter("period"));
int id = Integer.parseInt(request.getParameter("course_id"));
CourseDao coursedao = new CourseDao();
ProDao prodao = new ProDao();
long x = coursedao.gx(cname, prodao.getId(pname), cnote, cimg, period, id);
request.setAttribute("x", x);
RequestResult requestresult = new RequestResult();
requestresult.updateResult(request, response);

```

修改课程详细与章节列表页面如图 7-3 所示:



图 7-3 课程信息修改

7.3.2 视频上传设计与实现

1、视频上传概述

“IMAU MOOC”主要目的在于在线视频学习，教师通过上传视频来让学生在线视频学习，文件上传通过将视频文件转化为 2 进制文件，再 2 进制转为视频文件。关键代码如下：

JSP 关键代码：

```
<form action="${pageContext.request.contextPath}/servlet/UploadHandleServlet"
  enctype="multipart/form-data" method="post">
    <input type="text" name="username"><br/>
    <input type="file" name="file1"><br/>
    <input type="submit" value="提交">
</form>
```

Servlet 关键代码如下：

```
String basePath = this.getServletContext().getRealPath("/video");
DiskFileItemFactory fact = new DiskFileItemFactory();
ServletFileUpload sf = new ServletFileUpload(fact);
sf.setSizeMax(1024*1024*100);
```

```
try {  
    List<FileItem> list=sf.parseRequest(request);  
    for (FileItem fileItem : list) {  
        String fileName = fileItem.getName();  
        File file=new File(basePath + "/" + fileName);  
        fileItem.write(file);  
    }  
} catch (Exception e) {  
    e.printStackTrace();  
}
```

上传视频页面如图 7-4:

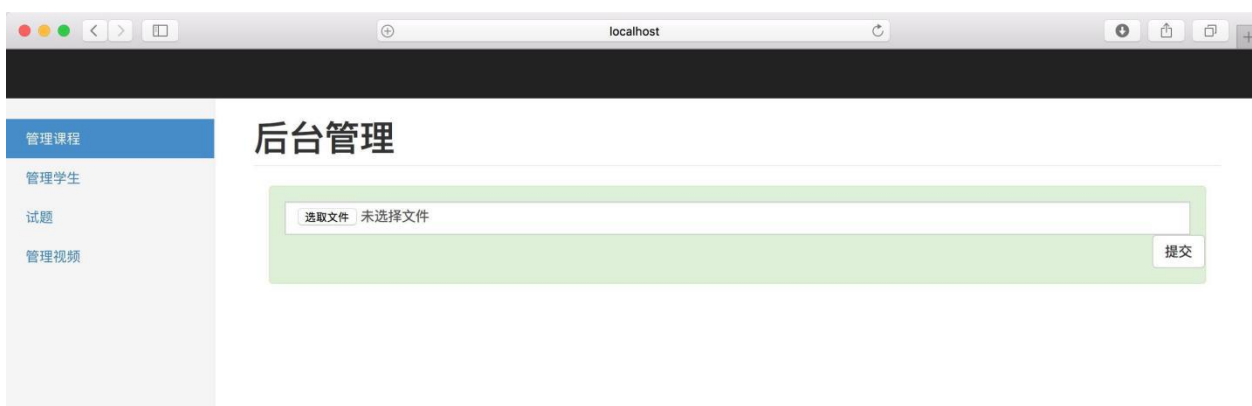


图 7-4 视频上传页面

7.4 本章小结

本章介绍了网站后台详细设计与实现，详细介绍了课程列表、课程修改、课程添加、以及视频上传的详细设计与实现。对其中关键页面的代码进行了展示。

第八章 总结与展望

本课题为在线视频学习数据挖掘网站，在老师同学的帮助下，我终于设计出了一个针对在线视频学习在线用户行为的数据挖掘网站，充分了用了我在学校学习到的专业知识，也将数学建模利用到其中，在线视频学习网站叫做“IMAU MOOC”。在做毕业设计的期间李，主要的工作内容为：

- 1、研究国内外在线视频学习网站的现状
- 2、研究国内外数据挖掘的方法与应用
- 3、研究和学习毕业设计相关技术，如 JAVA、HTML、JSP、JavaScript、JavaBean、SQL 等等技术，这些都是我以前从未学习过的。
- 4、通过学习 MVC 架构，将网站的整个开发都利用 MVC 的思想进行的开发
- 5、通过对网站的总体设计、详细设计、数据库设计、实施、测试，对整个网站开发有了一个深刻的了解，并从中发现了自己喜欢的方向。

本次毕业设计课程主要成果如下：

- 1、完成了毕业设计说明书《在线视频学习数据挖掘》的撰写；
- 2、设计了针对在线用户的学习行为的数据收集、数据整理、数学建模、完成了简单的机器学习，最终实现了在线视频学习用的数据挖掘；

本次毕业设计让我学习到了很多新的知识，并且为学习数学建模与编程的同学提供了一些学习的参考与借鉴。

但是由于本人的知识水平，学习时间的限制以及开发经验的限制，《IMAU MOOC》做得并不是十分的完善，在很多细节处理的地方并没有考虑清楚，页面美化也并不是很理想，最终的数据可视化与文件上传还存在一些问题。希望通过以后的学习可以将这些不足一一弥补。

致 谢

首先感谢我的指导老师：(匿名)老师，他的严谨细致、一丝不苟的作风一直是我学习、工作的榜样；他自始至终严厉的要求我认真与坚持，是我这次毕业设计顺利完成的很大原因，他循循善诱的教导思路给予非常大我的启发和鼓励，从一开始 HTML 看不懂、JavaWeb 看不懂、SQL 看不懂到最后可以简单上手完成最后的毕业设计，这个巨大的进步自己都有些骄傲和满意，当然这在以后的职业生涯中还是远远不够的，但是老师的对我工作热情的激发一定是终身受益的。

在大学期间，导师为我创造了优越的学习环境，是我在计算机信息管理学科中自由翱翔，同时在思想上、人生态度上、意志品质方面给予了淳淳教诲，这些教益必定在我今后的道路上激励我奋勇前行。

感谢大学里的同学们，在我遇到困难时帮我解决与分析，使我学习速度、学习方法有了很大的提升。

感谢内蒙古农业大学职业技术学院的所有教师和领导，在学校生活、学习条件的关系和支持。

感谢我的家人，你们是我在学习和工作的最大动力，谢谢你们的无私付出，让我安心学习和工作，

最后感谢母校内蒙古农业大学，在这个祖国北疆的这个学风浓郁校园里，我挥洒汗水，收获知识和喜悦，今后我将用实际工作来向母校汇报！

参考文献

- [1] Java-百度百科【EB/OL】
- [2] HTML-百度百科【EB/OL】
- [3] JavaScript-百度百科【EB/OL】
- [4] JSP-百度百科【EB/OL】
- [5] JavaBeanK-百度百科【EB/OL】
- [6] ServletK-百度百科【EB/OL】
- [7] 流媒体-百度百科【EB/OL】
- [8] MVC 模式-百度百科【EB/OL】
- [9] 机器学习-百度百科【EB/OL】
- [10] RFM 模型 - MBA 智库百科【EB/OL】
- [11] 层次分析法 - MBA 智库百科【EB/OL】
- [12] Stone, Bob. Successful Direct Marketing Methods【M】.Linco lnwood:NTC Bussiness Books, 1994
- [13] 连惟谦. 应用资料分析技术进行顾客流失与顾客价值之研究【M】. 台湾:中原大学 . 2004
- [13] 蒋玲. 专题学习网站管理系统的设计与实现【D】. 武汉: 华中师范大学, 2004, 1-12
- [14] 刘惠娟. 基于 WEB 的视频课件系统的设计与实现【D】. 广州: 华南理工大学, 2012, 2-15
- [15] 常锋峰, 谢运佳. 专题学习网站的现状和发展趋势【J】. 广东技术师范学院学报, 2008(10): 24-27
- [16] 冯军, 刘福敏. 建设专题学习网站促进学校网络资源的发展【J】. 教育教学论坛, 2010(7): 14-16
- [17] 段海新, 扬家海, 吴建平. 基于 Web 和数据库的网络管理系统的设计与实现【J】. 软件学报, 2012(6): 7-9

- [18] 范生万, 陈薇. 基于 Web2.0 的 B-Learning 学习系统开发技术研究【J】. 电脑知识与技术,
2009(3): 23-29
- [19] 徐磊. 网页制作与网站建设技术大全【M】. 北京: 清华大学出版社, 2008, 42-46
- [20] 柴晓路, 梁宇奇. Web Service 技术、架构和应用【M】. 北京: 电子工业出版社.
2003, 5-16
- [21] 王宇静, 胡文发. 基于 Web Service 的建设项目集成化管理信息系统研究【J】.
电子学报,
2005(6): 25-41
- [22] 柴晓路. Web 服务架构与开放互操作技术【M】. 北京: 清华大学出版社, 2002, 3-14
- [23] 徐美仙. 专题学习网站的设计与案例分析【M】. 广东: 广东教育出版社, 2004, 7-16
- [24] 李刚. Java 疯狂讲义【M】. 北京: 电子工业出版社, 2012, 27-35
- [25] 梁斌, 李谕吕. 基于 Wiki 的专题学习网站的设计与实现【M】. 北京: 中国远程教育
, 2007,
12-13
- [26] 席蓓蓓, 刘葭. Wiki 与专题学习网站相整合的策略【J】. 软件导刊·教育技术,
2008(5): 8-14
- [27] 飞思科技产品研发中心. JSP 应用开发详解 (第二版)【M】. 北京: 电子工业出版社
, 2004,
5-26
- [28] 孙卫琴, 李洪成. Tomcat 与 JSP Web 开发技术详解【M】. 北京: 电子工业出版社,
2003, 35-57
- [29] BruceEckel. JSP 编程思想【M】. 北京: 机械工业出版社, 2003, 75-83



学校代码： 10129
系 别： 计算机技术与信息管理系
年 份： 2017 年 6 月

内蒙古农业大学
Inner Mongolia Agricultural University

毕业设计说明书

题 目： 在线视频学习数据挖掘

小组成员： 塔宇

班 级： 14 信息管理

指导教师： 包永红

2017 年 6 月

