第九届中国大学生计算机设计大赛

作品综合设计报告

题　　目： 基于 Hadoop 的学习资源推荐系统

队　　名： MagicCloud

参赛队员： 唐士杰、张清恒、郑燏辉

指导老师： 张雪洁

参赛学校： 河海大学

目录

一、 绪论

1.1 背景

在当今的时代背景下，学习已经从工作需要或者生活需要越来越演变成了人们的一种自发的行为。如今，学习已经逐渐成为一种主流、时尚的生活方式，是人们生活中不可或缺的重要组成部分。而自学又在其中占据了极大的比例。从适应人们的需求，顺应时代潮流出发，众多的在线学习网站如慕课网、百度传课网以及腾讯课堂网、网易云课堂等如雨后春笋一般涌现出来，为人们的自学带来了极大的方便，也大大降低了学习的成本。

然而，在线学习网站的大量出现，除了可以带给我们方便，降低学习成本之外，也给人们带来了一定程度的困扰，那就是，如何在浩如烟海的网络资源中找到最适合自己，最符合自己学习实际的学习资源呢？显然，作为资源的展示和交流的平台，当下现有的学习网站并没有提供学习资源推荐的服务，这样就要求人们必须自主地去寻找最适合自己的资源，从而导致人们将大量的精力浪费在寻找资源上而非应用于学习中。这不得不说是一种遗憾，也影响了人们在线学习的方便性。

基于这样的背景，我们团队设计开发一款学习资源推荐系统，力求将存于各平台数据库中的资源准确、高效、完备的推荐给用户。

1.2 项目内容

该项目的主要目标是设计实现一个提供各种类型学习资源推荐的系统，系统可以方便用户快速查找所需的学习资源。同时该系统将支持用户的在线学习以及学习资源推荐，为用户带来方便；该系统将爬虫技术结合云存储技术，利用爬虫广泛收集在线学习网站的信息，并且利用云存储技术存储关键信息；该系统在数据安全的基础上为用户提供准确、全面的学习资源推荐信息；该系统专注学习资源的推荐，为用户能相对方便快捷地找到自己需要的学习资源提供强有力的保障。

1.3 特色

1.3.1 功能特色

（1）个性化推荐

本系统根据用户个人留下的浏览、学习、评分以及收藏等记录得到原始数据，进而根据这些数据，利用设计好的算法进行计算，给出针对特定用户进行个性化的推荐，这样使得推荐结果满足特定用户的需要，用户体验极佳。

（2）多样化推荐

本系统为了实现推荐的准确性和广泛性，将采用多种推荐方式：1、针对用户没有产生浏览记录时（即冷启动问题），系统推荐当前最热门的资源；/2 当用户产生一定数量的访问记录后，则可以根据当前用户的访问记录形成针对当前用户的推荐；3、最后，当用户数量达到一定规模时，本系统可以利用协同过滤算法，根据一些用户的浏览记录向其他用户进行推荐。

（3）资源来源广泛

本系统中集合了当前网络上比较知名的各大在线学习网站的资源，力求为用户提供最丰富的资源和多样化的选择。

（4）系统高效易用

满足系统高效访问需求；爬取、存储、抽取等环节高效完成；满足不同用户、不同功能之间并发的需求。

系统性能稳定、可靠运行，有较好的检错能力，对于单点故障，能够迅速有效解决，并且保证不丢失重要数据。

系统的用户图形界面友好，人性化，方便用户快速获取想要的旅游信息；系统管理界面能直观反映系统当前运行情况。

该业务系统具有良好的扩充能力，提供今后扩充系统功能、规模的接口，且系统功能扩充时不影响原系统的功能。

二、系统介绍

2.1 系统用户特点

本系统主要面向希望获得准确学习资源推荐信息的用户。用户使用该系统获取学习资源的推荐信息，从而帮助自己提高学习效率，使学习更具目标性，因此对信息的准确性要求较高，同时信息及时更新也至关重要。考虑到可能会有很大的用户量，系统须具备较快的运行速度。

2.2 系统功能

2.2.1 在线学习网站信息爬取

从给定的各个在线学习网站爬取信息。在爬取过程中，爬虫将会不断发掘新的 URL，经过页面下载、URL 发现、URL 去重、目标页面信息存储几个步骤，完成一个完整的爬取过程。

2.2.2 信息存储

存储的信息主要包括：爬取获得的目标网页的全部信息、经过分析和提取之后得到的有价值信息、用户与管理员信息等。

该部分采用关系型数据库（MySQL）与非关系型数据库（HBase）相结合的方式，优势互补，提高系统运行效率。

2.2.3 关键信息抽取

从已爬取的网页内容中，提取出有价值的旅游信息，已爬取的网页以 json 格式存储在 HBase 数据库中，需要先从数据库中读取数据，然后由 json 格式转换成 html 形式的“page”，再利用 jsoup、Xpath、正则表达式等解析工具进行页面解析和提取工作。

2.2.4 课程推荐

本系统的课程推荐主要有三种方式：1、公共推荐，系统的主页将会按照资源的类别提供一系列推荐，将当前最热门的资源展示在主页当中；2、基于当前浏览记录的推荐，当当前用户产生浏览记录的时候，系统就可以根据这些浏览记录进行计算，并给出针对当前用户的推荐‘3、基于用户的个性化推荐，系统采用协同过滤算法，根据一些用户的信息给出对特定用户的推荐。

2.2.5 个人中心管理

本系统的个人中心主要提供如下功能：1、查看和修改个人信息，用户在注册时填写的个人信息均会展示在个人中心中，而用户可以在这里进行查看以及修改；2、查看或删除收藏信息和浏览记录，用户在使用本系统的时候自然会产生一系列浏览记录，这些记录都会在这里展示，用户可以执行查看或删除操作，而对于收藏信息，也同样可以进行查看或删除。

三、系统设计

3.1 总体架构

本系统采用基于 SOA 的分布式应用框架和 B/S 结构，系统从服务器到客户端分为分布式爬取层、数据存储层、业务逻辑层、前端展示层（浏览器）。技术架构如图 3-1 所示:

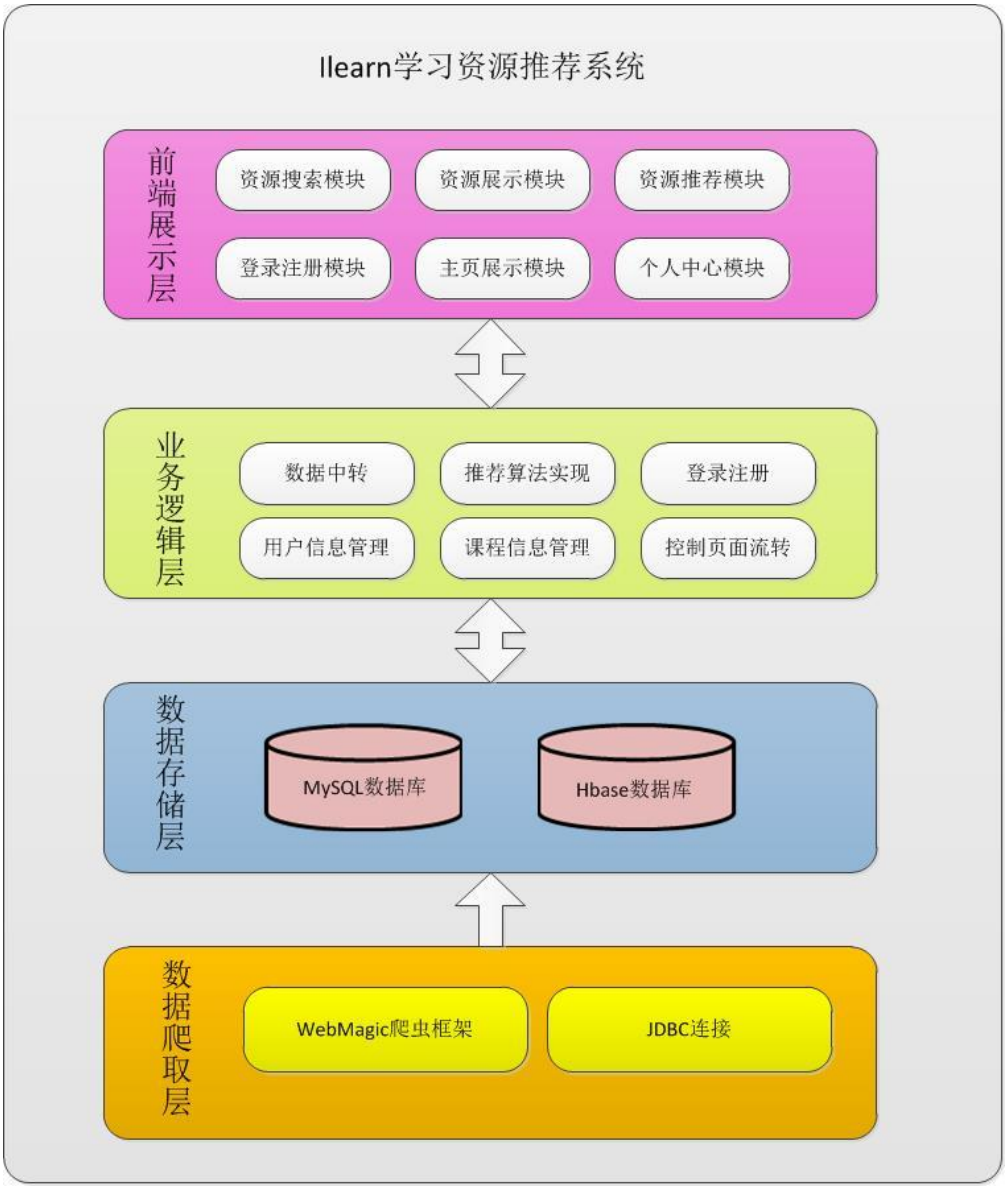

图 3-1 系统架构图

为了提高学习资源推荐系统的灵活性、可重用性、高可靠性及使用的方便性，整个系统分层实现，分布式爬取层和网站业务逻辑层的开发是整个系统的核心。其中业务逻辑层是实现数据操作及事务管理的，主要技术特征有：

（1）基于 SOA 的理念和技术，采用了 REST 架构风格发布服务，便于接入到用户体验平台和系统管理平台中。

（2）符合实际业务需求，通过处理用户不同的浏览记录，系统进行用户定制或推荐路线的查找、比价、决策、收藏等操作，同时管理员也可以通过管理平台进行相关业务处理，方便快捷。

（3）减少业务处理层和服务提供层的耦合度，提高系统安全性，维护方便，具备数据容灾功能。

3.2 模块划分

3.2.1 分布式爬取模块

该模块由系统设计的高效爬虫实现。爬虫在启动后爬取指定在线学习网站的页面，从中抽取出与学习资源相关的关键信息，并将其存储到系统的数据库中以备使用。

3.2.2 数据存储模块

数据存储模块由基于 Hadoop 的非关系型数据库 HBase 以及关系型数据库 MySQL 共同组成，两种不同类型的数据库相辅相成，扬长避短，共同完成数据存储的任务。

3.2.3 业务逻辑模块

该模块主要包括课程推荐、个人中心管理、登录注册的逻辑实现以及资源查询等部分。在业务逻辑模块中实现了前述的推荐算法，真正让推荐开始发挥作用，同时帮助用户管理个人中心，实现用户中心信息的抽取与组织，实现了登录注册的逻辑，并且可以实现资源的抽取与组织。

3.2.4 网页展示模块

该模块主要由各前端页面组成，包括主页、资源展示和个人中心等一系列页面，同时为了更好地展示数据，给用户带来非常好的使用体验，在这一模块中还

用到了 bootrap、jQuery、ajax 等框架与技术。

3.3 关键技术描述

（1）基于协同过滤的推荐算法

本系统涉及的推荐算法主要为基于协同过滤的推荐算法，其中包括基于用户的协同过滤算法和基于 Item（即课程）的协同过滤算法，该推荐算法由三部分构成：针对用户没有产生浏览记录时（即冷启动问题），系统推荐当前最热门的资源；当用户产生一定数量的访问记录后，则可以根据当前用户的访问记录形成针对当前用户的推荐；最后，当用户数量达到一定规模时，本系统可以利用协同过滤算法，根据一些用户的浏览记录向其他用户进行推荐。

（2）高效的爬虫设计

本系统利用网络上一个成熟的优秀爬虫框架——WebMagic 来定制自己的爬虫，并利用该定制的爬虫进行资源的爬取，保证整个系统可以有丰富的资源可供调用。同时结合系统自身特点，对 WebMagci 进行特定方向的技术优化，使得系统的爬虫具有高效性和针对性，可以在较短的时间内完成系统的任务需求。

（3）基于 Struts2+SpringMVC+Hibernate 框架的业务开发模型

本系统使用基于 Struts2 + SpringMVC + Hibernate 的 JavaWeb 开发框架，高效的构建系统的业务开发模型，借用三大框架的整合优势，提高系统的开发效率，同时也保证系统的稳定运行。

四、详细设计

4.1 数据爬取层

4.1.1 功能说明

本系统的数据爬取层使用网络中比较成熟的优秀爬虫框架 WebMagic 来进行定制，WebMagic 的架构图如图 4-1 所示：

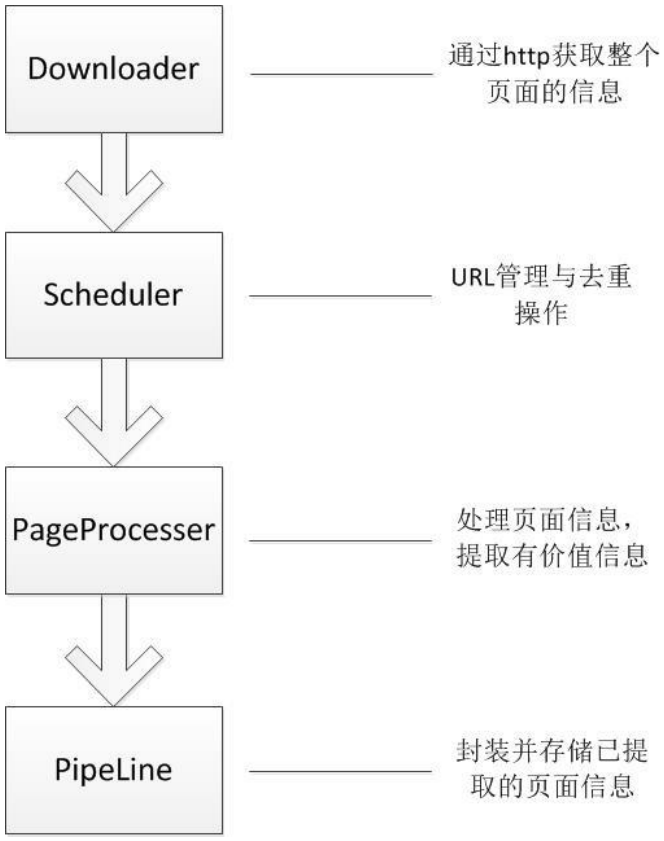

图 4-1 WebMagic 架构图

爬虫通过 Downloader 从网络上爬取得到页面信息，通过 Scheduler 进行去重操作，随后将页面信息传送给 PageProcessor，在 PageProcessor 中可以抽取出页面中所需的信息，然后根据设计好的逻辑对数据进行处理，一般来说，PageProcessor 是用户自定义的部分，而本系统也就是通过实现 PageProcessor 而实现对爬虫的定制。

4.1.2 处理流程

爬虫的启动以及处理流程如图 4-2 所示：

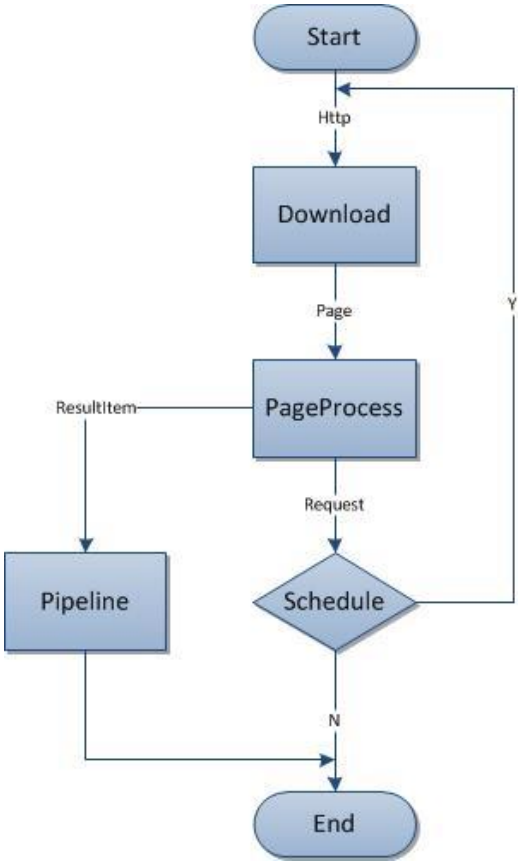

图 4-2 爬取流程图

1.Spider 启动后，爬取任务开始；

2.Spider 调用 Download，把某个 page 的全部内容获取并下载；

3.将 page 传入 PageProcess，PageProcess 将处理两个过程：将当前页面的内容封装成 ResultItem，交给 Pipeline 处理；将当前页面中的所有新的 URL 请求交给 Schedule 处理；

4.Schedule 获取到新的 URL 请求后，进行去重工作，将尚未爬取的 URL 请求提交给 Spider，进行新一轮爬取；

5.Pipeline 对 ResultItem 进行格式处理，并存入指定数据库。

4.1.3 关键实现技术描述

Spider 类(核心调度)

Spider 是爬虫的入口类，Spider 的接口调用采用了链式的 API 设计，其他功能全部通过接口注入 Spider 实现。

PageProcessor(页面分析及链接抽取)

页面分析是垂直爬虫中需要定制的部分。在 webmagic-core 里，通过实现 PageProcessor 接口来实现定制爬虫。PageProcessor 有两个核心方法：public void process(Page page)和 public Site getSite()。

Downloader(页面下载)

Downloader 是 webmagic 中下载页面的接口，主要方法：

public Page download(Request request, Task task)

Request 对象封装了待抓取的 URL 及其他信息，而 Page 则包含了页面下载后的 Html 及其他信息。Task 是一个包装了任务对应的 Site 信息的抽象接口。

public void setThread(int thread)

因为 Downloader 一般会涉及连接池等功能，而这些功能与多线程密切相关，所以定义了此方法。

public void process(Page page)

通过对 Page 对象的操作，实现爬虫逻辑。Page 对象包括两个最重要的方法：addTargetRequests()可以添加 URL 到待抓取队列，put()可以将结果保存供后续处理。 Page 的数据可以通过 Page.getHtml()和 Page.getUrl()获取。

public Site getSite()

Site 对象定义了爬虫的域名、起始地址、抓取间隔、编码等信息。

Scheduler(URL 管理)

Scheduler 是 webmagic 的管理模块，通过实现 Scheduler 可以定制自己的 URL

管理器。Scheduler 包括两个主要方法：

public void push(Request request,Task task)

将待抓取 URL 加入 Scheduler。Request 对象是对 URL 的一个封装，还包括优先级、以及一个供存储数据的 Map。Task 仍然用于区分不同任务，在多个任务公用一个 Scheduler 时可以此进行区分。

public Request poll(Task task)

从 Scheduler 里取出一条请求，并进行后续执行。

Pipeline(后续处理和持久化)

Pipeline 是最终抽取结果进行输出和持久化的接口。它只包括一个方法：

public void process(ResultItems resultItems,Task task)

ResultItems 是集成了抽取结果的对象。通过 ResultItems.get(key)可以获取抽取结果。Task 同样是用于区分不同任务的对象。

4.2 数据存储层

4.2.1 功能说明

数据存储层主要负责存储数据爬取层从网络上爬取到的信息，并将它们按照合理的顺序组织起来，以方便业务逻辑层的抽取。可以说数据爬取层与其他两层（业务逻辑层和前端展示层）就是通过数据存储层进行耦合的，因此数据存储层至关重要。

4.2.2 处理流程

数据存储层的顺序图如图 4-3：

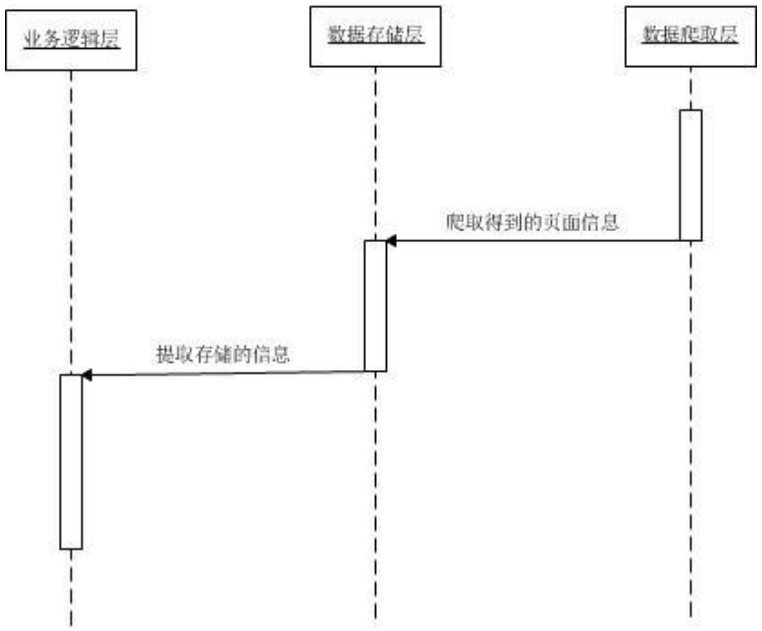

图 4-3 数据存储层顺序图

数据存储层接受数据爬取层的输入，得到系统所需的各种关键信息，然后按照一定的组织方式有机地将其存储到数据库中，之后业务逻辑层从数据存储层中查询出与业务逻辑相关的数据，并按照设计好的算法进行计算和处理。

4.2.3 关键实现技术

（1）MySQL

MySQL 是一个关系型数据库管理系统，由瑞典 MySQL AB 公司开发，目前属于 Oracle 旗下公司。MySQL 最流行的关系型数据库管理系统，在 WEB 应用方面 MySQL 是最好的 RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件之一。

MySQL 是一种关联数据库管理系统，关联数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。

（2）HBase

HBase HBase – Hadoop Database，是一个分布式的、面向列的开源数据库，也是一个高可靠性、高性能、面向列、可伸缩的分布式存储系统，利用 HBase 技术可在廉价 PC Server 上搭建起大规模结构化存储集群。不同于一般的关系数

据库，HBase 是一个适合于非结构化数据存储的数据库。

4.3 业务逻辑层

4.3.1 关键信息抽取

（1）功能说明

若在爬取页面信息的同时进行信息抽取工作，则整个系统的效率会完全受限于爬虫的运行效率，因此系统设计分为两个过程，首先将整个页面爬取得到，然后进行第二步，从中抽取信息，本功能即实现第二步，抽取信息。

（2）处理流程

处理流程图如图 4-4：

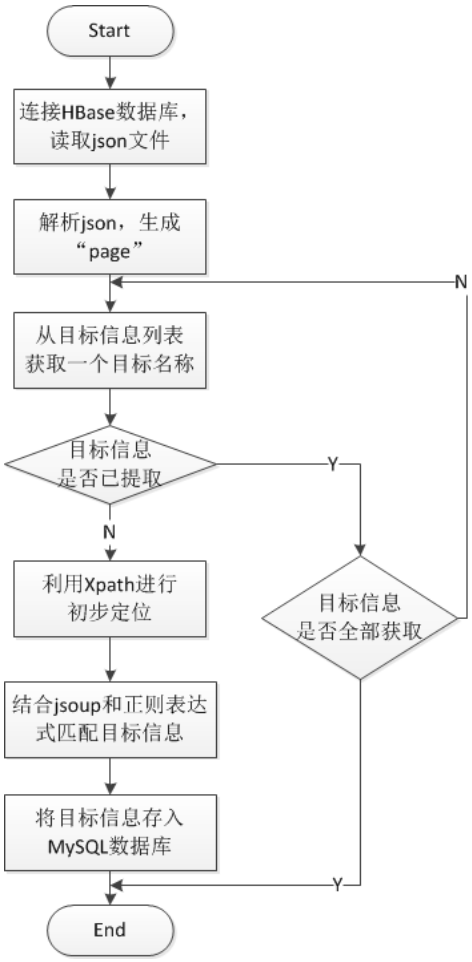

图 4-4 抽取流程图

1.连接 HBase 数据库，并从中读取之前已经存为 json 格式的页面数据，此处读取时，以出发地点和目标地点为读取依据；

2.从已设定的目标信息欲提取列表中，获取一个目标信息的名称；

3.检查该目标信息是否已被提取；

4.若该目标信息未被获取，先利用 Xpath 进行标签定位；

5.结合 Jsoup 和正则表达式，具体匹配该目标信息；

6.若该目标信息已经被获取，检查是否所有目标信息都已被获取；

7.将最终提取的具体目标信息，存入 MySQL 数据库。

（3）关键实现技术描述

1、json：是一种轻量级的数据交换格式，可以将 对象中表示的一组数据转换为字符串，然后就可以在函数之间轻松地传递这个字符串，或者在异步应用程序中将字符串从 Web 客户机传递给服务器端程序，本项目中主要用作中间形态的数据格式，与 Hbase 数据库交互。

2、Xpath：即为 XML 路径语言，是一种用来确定 XML（标准通用标记语言的子集）文档中某部分位置的语言，XPath 基于 XML 的树状结构，提供在数据结构树中找寻节点的能力，项目利用 Xpath 进行初步定位，为后续操作做准备。

3、Jsoup：一款 Java 的 HTML 解析器，可直接解析某个 URL 地址、HTML 文本内容，提供了一套非常省力的 API，可通过 DOM、CSS 以及类似于 jQuery 的操作方法来取出和操作数据，项目中主要用来做解析工作。

4、正则表达式：使用单个字符串来描述、匹配一系列符合某个句法规则的字符串，在 Xpath 初步定位的情况下，使用正则表达式来完成具体的匹配和筛选。

4.3.2 推荐信息生成

（1）功能说明

本系统为学习资源推荐系统，因此推荐部分是系统的核心。该部分实现从数据存储层中提取全部所需信息，然后根据提前设计好的推荐算法进行计算和处理，最终得到令人满意的推荐信息，然后将这些推荐信息发送到前端展示层，通过前

端页面展示给用户，从而实现学习资源的推荐。

（2）处理流程

1、使用数据操作部分从数据库中查询所需的关键信息（本系统中使用 ORMapping 框架：hibernate）；

2、遍历得到从数据库中抽取出的信息（必要时进行类型转换）；

3、对数据进行处理，依据推荐算法计算出相应的结果；

4、根据计算出的结果从数据库中取出对应数据；

5、将取出的数据发送至前端页面。

（3）关键实现技术

1、针对冷启动的问题，本系统的推荐策略为查找特定类别中已有用户评分最高的前 6 位资源进行推荐，在主页中可以展示多个类别的当前热门资源；

2、在用户产生一定量的浏览记录后，使用余弦相似度算法，计算与用户当前浏览过的资源相似度最高的资源，认为用户很有可能会感兴趣，将这些资源推荐给用户，余弦相似度算法解释如下：

使用余弦相似度算法，提取每个学习资源的一系列特征组成一个 N 维的向量，（类型，收藏人数，评分，学习人数，满意度），可以认为这是一个空间中的 5 维向量，每个学习资源都具有这样一个向量，此时根据余弦定理，计算出这两个向量之间的夹角，若为 0°，则认为两个向量重合，若为 90°，则认为没有关系，若为 180°则认为刚好相反，因此根据余弦定理的公式：

$$\cos\theta = \frac{\sum_{i=1}^{n}(A_i \times B_i)}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}} = \frac{A \cdot B}{|A| \times |B|}$$

图 4-5 余弦定理公式

得到的值介于 0 到 1 之间，越接近 1 则认为相似度越高。当用户浏览过本网站，产生浏览记录之后，则可以根据余弦向量法计算用户浏览过的产品和数据库

中产品的相似度，将符合要求的产品（认为余弦相似度大于等于 0.95 符合要求）定向推荐给特定用户。

3、当有一定数量的用户之后，采用基于用户的协同过滤算法以及基于 Item 的协同过滤算法进行推荐，协同过滤算法的思想简单描述就是：

首先有多个用户，每个用户都有一系列的浏览记录，然后计算不同用户之间的相似度，若两个用户之间相似度较高，则认为其中一个用户感兴趣的资源另外一个用户也有可能会感兴趣，因此将该用户浏览过而相似用户没有浏览顾的资源推荐给相似用户。

4.4 前端展示层

4.4.1 功能说明

前端展示层是直接与用户交互的一层，它沟通了用户和整个系统，系统通过爬虫爬取信息，通过数据存储层存储数据，通过业务逻辑层抽取并处理数据，最终将计算结果发送到前端展示层展示给用户，而用户通过前端页面这个借口向系统传递自己的请求，之后系统根据用户的请求针对性地从系统中抽取数据，组织并展示数据。

4.4.2 处理流程

前端展示层的顺序图如图 4-6：

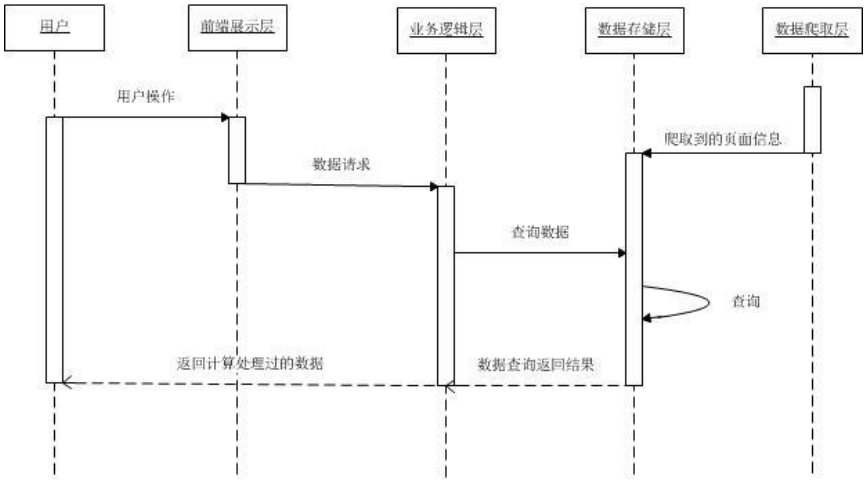

图 4-6 前端展示层顺序图

1、 数据爬取层首先进行数据的爬取，为系统准备好数据；

2、 用户在前端页面进行操作；

3、 前端页面将用户操作的请求发送给业务逻辑层；

4、 业务逻辑层根据用户请求，针对性地对数据库中的数据进行增、删、改、查操作；

5、 数据存储层向业务逻辑层返回操作结果；

6、 业务逻辑层根据预定算法对返回的数据进行计算和处理；

7、 业务逻辑层将数据返回到前端展示层，展示给用户。

4.4.3 关键实现技术

1、MVC 设计模式：MVC 全名是 Model View Controller，是模型(model)－视图(view)－控制器(controller)的缩写，M 是指业务模型，V 是指用户界面，C 则是控制器，使用 MVC 的目的是将 M 和 V 的实现代码分离，从而使同一个程序可以使用不同的表现形式。它是一种软件设计典范，用一种业务逻辑、数据、界面 显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。

2、Bootstrap：前端采用的是 Bootstrap，来自 Twitter，是目前很受欢迎的前端框架。Bootstrap 是基于 HTML、CSS、JAVASCRIPT 的，它简洁灵活，使得 Web 开发更加快捷。Bootstrap 提供了优雅的 HTML 和 CSS 规范，它即是由动态 CSS 语言 Less 写成。国内一些移动开发者较为熟悉的框架，如 WeX5 前端开源框架等，也是基于 Bootstrap 源码进行性能优化而来。其具有相当简洁的 css 样式，组件与插件封装。利用该框架可以十分方便的实现一个风格独特的网页，另外利用 HTML5 与 CSS3 的新特性可以呈现一个良好的动画过程，赋予了网页新的活力。

3、Spring MVC：Spring MVC 属于 SpringFrameWork 的后续产品，已经融合在 Spring Web Flow 里面。Spring 框架提供了构建 Web 应用程序的全功能 MVC 模块。使用 Spring 可插入的 MVC 架构，从而在使用 Spring 进行 WEB 开发时，可以选择使用 Spring 的 SpringMVC 框架或集成其他 MVC 开发框架，如 Struts1，Struts2 等。

五、数据库设计

5.1 概述

本数据库服务于 ILearn 团队的 ILarn 学习资源推荐系统。

本数据库命名均以 ilearn_ 开头，再加上表内容的英文单词或英文短语缩写，这样可以统一命名，并且见名知意。

ILearn 系统使用的数据库软件为：Hbase 0.98 和 MySQL 5.0，数据库运行环境为：Ubuntu14.04。

5.2 概念结构设计

5.3 逻辑结构设计

5.3.1 数据库表设计

本系统中用到的数据库表如表 5-1 所示：

表 5-1 数据库表

表名	中文表名	数据库类型	功能描述
ilearn_user	用户信息表	MySQL	存储用户信息
ilearn_resources	课程资源表	MySQL	存储课程信息
ilearn_categoey	课程目录表	MySQL	存储目录信息
ilearn_recommend	推荐课程表	MySQL	存储定期更新的推荐课程信息
ilearn_collection	用户收藏信息表	MySQL	存储收藏信息
ilearn_log	用户浏览记录表	MySQL	存储浏览记录
ilearn_crawler	爬取信息表	MySQL	存储爬取的相关信息
ilearn_resource_page	学习资源表	HBase	存储爬取的目标页面的信息

每张表详细设计情况分析如下：

（1）用户信息表 ilearn_user 设计

用户信息表用于存储用户的注册信息和统计信息，其中，对于用户登录密码要进行 MD5 加密存储。

表 5-2 用户信息表设计

字段名	字段类型	字段解释	示例
uid	int	主键；用户 ID	101
user_name	varchar	用户名	ilearn
password	varchar	用户密码	BFE94C76028861FD
telephone	varchar	联系方式	13845678901
email	varchar	电子邮件	ilearn@hhu.edu.cn
image_url	varchar	用户头像路径	/usr/local/ilearn/image/user
arr_collection_id	varchar	收藏信息数组	543，2688，10394
arr_log_id	varchar	浏览记录数组	23，54，98
direction	varchar	学习方向	云计算，后台开发

（2）课程资源表 ilearn_resources 设计

该表用于存储系统处理后的课程学习资源，其原数据来自于 ilearn_resource_page，其中 picture 字段用于存储图片路径。

表 5-3 课程资源表设计

字段名	字段类型	字段解释	示例
rid	int	主键；课程 ID	2337
title	varchar	课程标题	网页布局基础
url	varchar	课程路径	www.chuanke.com/1828532-106318.html
imgurl	varchar	课程图片路径	http://web.img.chuanke.com/cou

			rse/2015-02/06/.....jpg
category_1	varchar	一级目录名称	IT/互联网
category_2	varchar	二级目录名称	编程语言
category_3	varchar	三级目录名称	脚本语言
category_1_id	int	一级目录 ID	838
category_2_id	int	二级目录 ID	844
category_3_id	int	三级目录 ID	2
collection	int	收藏数量	143
remark	int	评论数量	77
grade	double	评分	0
satisfaction	double	满意度	0
join_number	int	学习人数	0
source_web	varchar	来源网站	百度传课网
rkey	varchar	关键字	网.....础
recommend_grade	double	推荐评分	10.206

（3） 课程目录表 ilearn_categoey 设计

该表用于存储课程目录信息，其中包括一级目录、二级目录、三级目录。一级目录的 category_1、category_1_id、category_2、category_2_id 四个字段均为 NULL；二级目录的 category_2、category_2_id 四个字段均为 NULL。

表 5-4 课程目录表设计

字段名	字段类型	字段解释	示例
id	int	主键；课程目录 ID	27
cate_name	varchar	目录名称	Linux
category_1	varchar	一级目录名称	IT/互联网
category_1_id	int	一级目录 ID	838
category_2	varchar	二级目录名称	操作系统

category_2_id	int	二级目录 ID	848

（4）基于用户的推荐课程表 ilearn_recommend_user 设计

该表用于存储基于用户推荐的课程资源信息，系统会定期更新当前推荐，该表与 ilearn_resources 表关联。

表 5-5 基于用户的推荐课程表设计

字段名	字段类型	字段解释	示例
rid	int	主键；推荐资源 ID	56
resource_id	int	详细资源 ID	2093
user_id	int	用户 ID	9
grade	double	推荐评分	16.4576
time	datetime	时间	NULL

（5）用户收藏信息表 ilearn_collection 设计

该表用于存储用户收藏信息，与 ilearn_user 表关联。

字段名	字段类型	字段解释	示例
coid	int	主键；收藏信息 ID	010
title	varchar	课程标题	网页布局基础
url	varchar	课程路径	www.chuanke.com/1828532-106318.html
imgurl	varchar	课程图片路径	http://web.img.chuanke.com/course/2015-02/06/.....jpg
uid	int	用户 id	101
rid	int	资源 id	56

（6）用户浏览记录表 ilearn_log 设计

该表用于存储用户浏览记录信息，与 ilearn_user 表关联。

字段名	字段类型	字段解释	示例
lid	int	主键；浏览记录 ID	015
title	varchar	课程标题	网页布局基础
url	varchar	课程路径	www.chuanke.com/1828532-106318.html
imgurl	varchar	课程图片路径	http://web.img.chuanke.com/course/2015-02/06/.....jpg
uid	int	用户 id	101
rid	int	资源 id	56

（7）爬取信息表 ilearn_crawler 设计

该表用于存储爬虫运行记录。

字段名	字段类型	字段解释	示例
crid	int	主键；运行记录 ID	015
duration	varchar	爬虫运行时长	34（单位为毫秒）
source_web	varchar	爬取的目标网址	www.chuanke.com
time	varchar	运行时刻的时间戳	2345453453342

（8）学习资源表 ilearn_resource_page 设计

字段名	字段类型	字段解释	示例
pid	int	主键：学习资源 ID	015
url	varchar	该资源的 url	www.chuanke.com/1828532-106318.html
page	varchar	资源页面信息	全部 HTML 代码

六、系统测试

6.1 测试环境

6.1.1 集群运行结点

配置	Master 性能参数	Worker 性能参数
处理器 CPU	1 核	1 核
主存	1024 MB	1024 MB
操作系统	Ubuntu 14.04 64 位	Ubuntu 14.04 64 位
带宽	1Mbps（峰值）	1Mbps（峰值）
云环境	Hadoop 2.4.0	Hadoop 2.4.0
数据库	Hbase 0.98.16.1 + MySQL 5.5.14	Hbase 0.98.16.1 + MySQL 5.5.14

6.1.2 服务器端

配置	性能参数
处理器 CPU	1 核
主存	1024 MB
操作系统	Ubuntu 14.04 64 位
带宽	1Mbps（峰值）
服务器环境	Nginx + Tomcat + Apache
云环境	Hadoop 2.4.0
数据库	Hbase 0.98.16.1 + MySQL 5.5.14

6.1.3 客户端

客户端只需要一个浏览器即可。为了更好的用户体验，在测试时使用了 Firefox 浏览器。

6.2 主要功能测试

6.2.1 单元测试

（1）注册单元

功能	输入	预期输出	实际输出
注册	账号/密码： null/null wer/123	①用户名或密码不能为空！②注册成功	一致

（2）登录单元

功能	输入	预期输出	实际输出
登录	账号/密码： wer/123 null/null 123/123	①登录成功②用户名或密码不能为空！③密码错误	一致

（3）目录单元

功能	输入	预期输出	实际输出
展示多级目录	鼠标悬浮于一级目录/二级目录	①展示对应全部二级目录②展示对应全部三级目录	一致

（4）首页推荐单元

功能	输入	预期输出	实际输出
动态展示当前最热资源	数据库数据更改	更新首页展示的推荐资源	一致

（5）资源展示单元

功能	输入	预期输出	实际输出
按照类别的筛选展示资源	鼠标点选一/二/三级目录	展示对应类别的资源	一致

（6）搜索单元

功能	输入	预期输出	实际输出
展示搜索得到的资源	资源的关键字	含有该关键字的相关资源	一致

（7）用户个性化推荐单元

功能	输入	预期输出	实际输出
对特定用户进行个性化推荐	某用户浏览记录	用户可能感兴趣的资源（与用户浏览过的资源相似度最高的资源）	一致

（8）协同过滤推荐单元

功能	输入	预期输出	实际输出
根据协同过滤算法进行推荐	一定量用户的浏览记录	向 A 用户推荐与该用户相似度最高的用户 B 浏览过而 A 用户未浏览过的资源	一致

（9）查看个人信息单元

功能	输入	预期输出	实际输出
向用户展示个人信息	用户鼠标点击相应按钮	当前用户的个人信息展示	一致

（10）修改个人信息单元

功能	输入	预期输出	实际输出
修改当前用户的个人信息	用户输入信息	用户个人信息更新	一致

（11）修改密码单元

功能	输入	预期输出	实际输出
修改当前用户密码	用户输入新密码	用户密码更改	一致

（12）收藏课程单元

功能	输入	预期输出	实际输出
收藏课程	用户鼠标点击	当前用户的收藏信息增加一条记录	一致

（13）查看收藏课程单元

功能	输入	预期输出	实际输出
展示当前用户收藏的课程	用户鼠标点击	当前用户收藏的课程	一致

（14）删除收藏单元

功能	输入	预期输出	实际输出
删除/批量删除收藏信息	用户鼠标点击	当前用户收藏信息更新	一致

（15）查看浏览记录单元

功能	输入	预期输出	实际输出
展示当前用户浏览记录	用户鼠标点击	当前用户浏览记录	一致

（16）删除记录单元

功能	输入	预期输出	实际输出
删除用户浏览记录	用户鼠标点击	用户浏览记录信息更改	一致

6.2.2 集成测试

（1）用户管理模块

功能	输入	输出	实际输出
注册	账号/密码： null/null wer/123	①用户名或密码不能为空！②注册成功	一致
登录	账号/密码： wer/123 null/null 123/123	①登录成功②用户名或密码不能为空！③密码错误	一致

（2）个人信息管理模块

功能	输入	预期输出	实际输出
向用户展示个人信息	用户鼠标点击相应按钮	当前用户的个人信息展示	一致
修改当前用户的个人信息	用户输入信息	用户个人信息更新	一致
修改当前用户密码	用户输入新密码	用户密码更改	一致

（3）课程收藏模块

功能	输入	预期输出	实际输出
收藏课程	用户鼠标点击	当前用户的收藏信息增加一条记录	一致
展示当前用户收	用户鼠标点击	当前用户收藏的	一致

藏的课程		课程	
删除/批量删除收藏信息	用户鼠标点击	当前用户收藏信息更新	一致

（4）浏览记录模块

功能	输入	预期输出	实际输出
展示当前用户浏览记录	用户鼠标点击	当前用户浏览记录	一致
删除用户浏览记录	用户鼠标点击	用户浏览记录信息更改	一致

（5）展示模块

功能	输入	预期输出	实际输出
展示多级目录	鼠标悬浮于一级目录/二级目录	①展示对应全部二级目录②展示对应全部三级目录	一致
按照类别的筛选展示资源	鼠标点选一/二/三级目录	展示对应类别的资源	一致
展示搜索得到的资源	资源的关键字	含有该关键字的相关资源	一致

（6）推荐模块

功能	输入	预期输出	实际输出
动态展示当前最热资源	数据库数据更改	更新首页展示的推荐资源	一致
对特定用户进行	某用户浏览记录	用户可能感兴趣	一致

个性化推荐		的资源（与用户浏览过的资源相似度最高的资源）	
根据协同过滤算法进行推荐	一定量用户的浏览记录	向 A 用户推荐与该用户相似度最高的用户 B 浏览过而 A 用户未浏览过的资源	一致

6.2.3 确认测试

（1）登录与注册

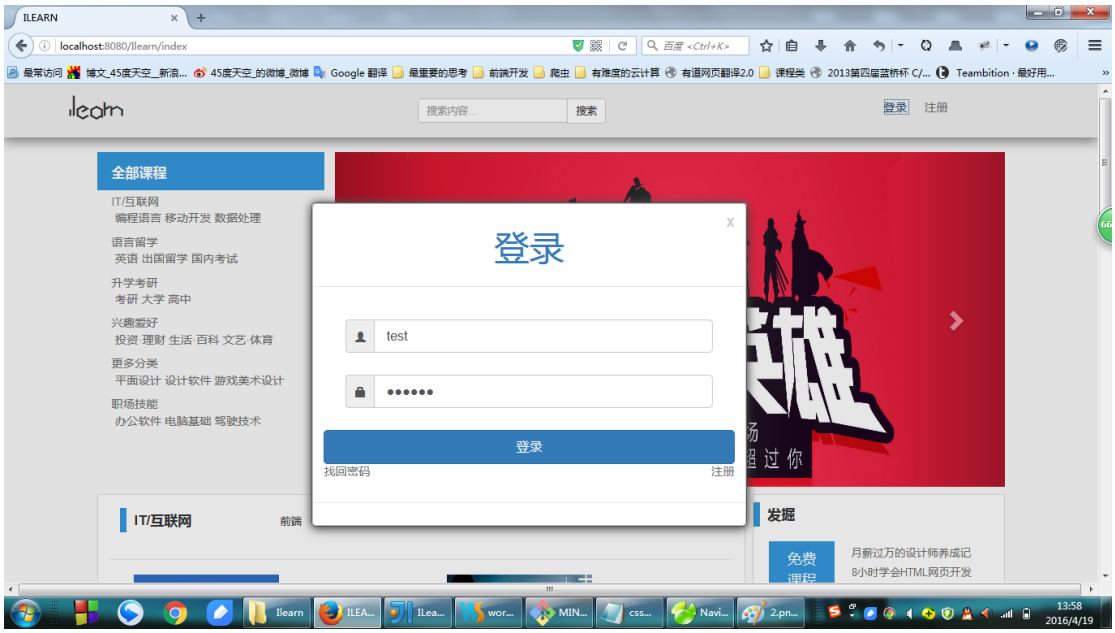

图 6-1 登录界面

图 6-2 注册界面

（2）首页

图 6-3 首页 1

图 6-4 首页 2

图 6-5 首页 3

（3）课程页面

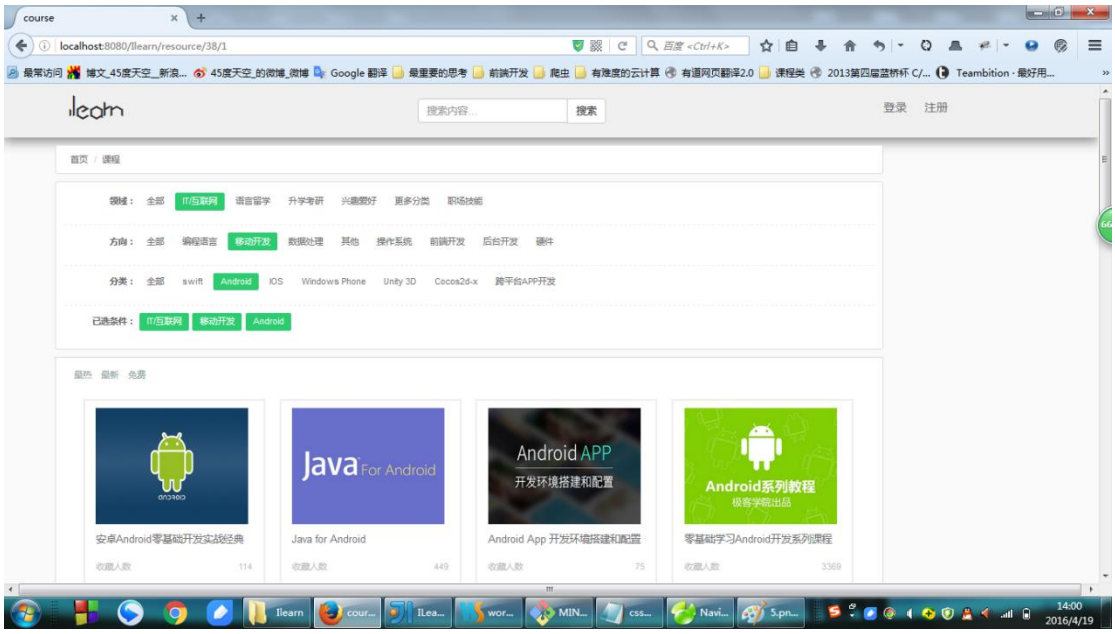

图 6-6 课程页面 1

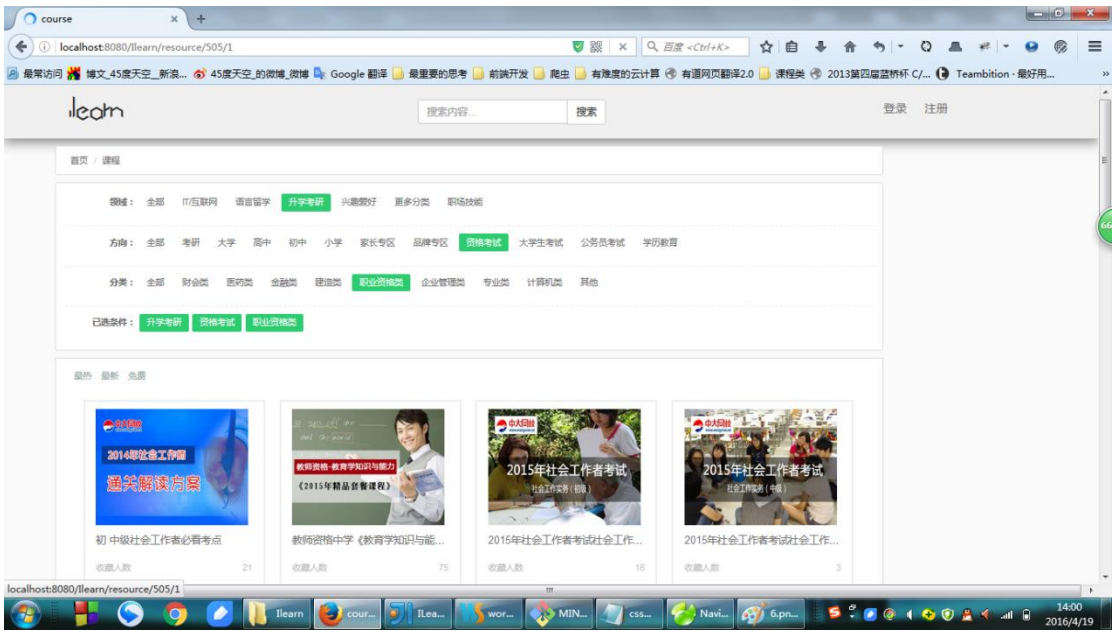

图 6-7 课程页面 2

（4）搜索页面

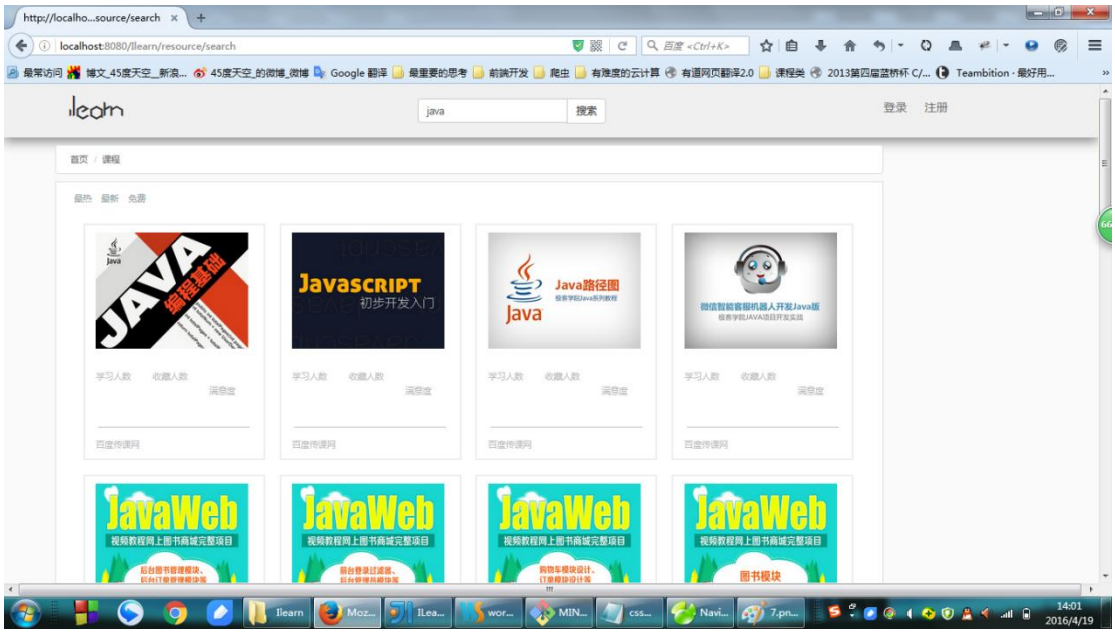

图 6-7 搜索页面 1

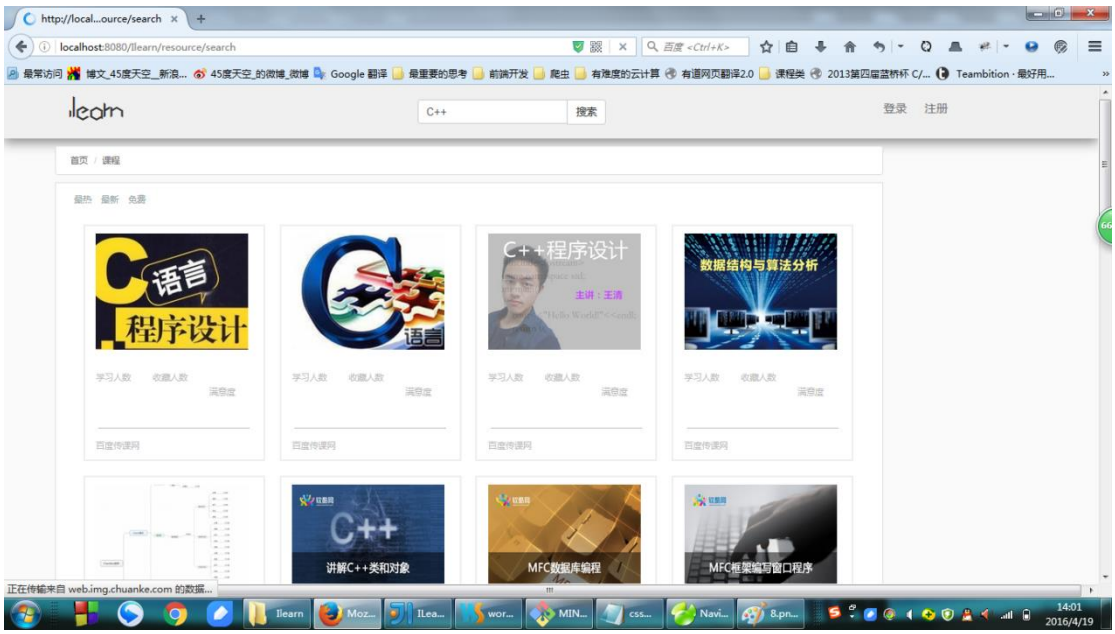

图 6-8 搜索页面 2

（3）个人中心页面

图 6-9 个人中心页面

七、总结

本平台完成了预定的功能性需求和非功能性需求，开发了一个完善的学习资源推荐系统，成功完成了所有任务。

在平台的设计和实现过程中，我们充分尊重用户的需求以及设身处地地为用户着想，为了简化用户的操作和降低使用难度，为用户提供了人性化的操作界面，便捷的操作接口。同时在系统的性能上做了较深的研究，充分保证了系统性能优秀。

如果继续做下一步的研究，可以考虑在现有系统的基础上探寻更多提高性能的方法，追寻系统的进一步优化，同时思考更多、更好、更适合的推荐方式，为广大用户提供一个更加个性化、更加友好并且更加智能化的学习资源推荐系统，将该系统持续发展下去，利用当今发展得如火如荼的信息化的技术为用户们提供舒适、贴心、便捷的服务。