

# 合肥工业大学 工程实训报告



专业班级： 物联网工程 17-2 班

姓 名： 文华

学 号： 2017218007

实训时间： 2020-2021 学年第一学期 1-6 周

合肥工业大学  
2020 年 10 月

实训时间	2020-2021 学年 第一学期 1-6 周	实训地点	计算机中心	指导教师	周波
实训小组 成员	姓名			学号	
	黄阳			2017219009	
	文华			2017218007	
	刘振			2017218178	
	刘城浩			2017217995	
	陈赞			2017217979	
	蔡合东			2017217977	
	王剑			2017217980	
实训计划及 完成情况	训练计划： 结合 Java Spring 架构和数据库等进行前端和后端的设计，运用所学软件和技术，实现一个培训机构课程管理系统。具体可以实现用户的登录和注册，个性化推荐，视频的分类查询，搜索和收藏。并且用户能够管理个人信息并绑定邮箱、管理课程信息以及进行评价。				
	完成情况： 达到预期目标，按照预期在项目结束时应该完成一个完整的课程培训网站。				
	个人负责工作： 在本次工程实训过程中，我主要负责借助 Spring 框架、使用 Java 语言开发系统的后端逻辑功能。				
	个人展望： 老师，給个良吧。				

<p>实训报告内容：包括绪论、系统设计、实现、测试以及总结体会等。</p>	<p>见后续正文。</p> <p>学生签名：</p>
---------------------------------------	----------------------------

# 目 录

1	绪论 .....	1
1.1	项目背景 .....	1
1.2	项目范围 .....	1
1.3	负责工作 .....	1
2	总体描述 .....	2
2.1	产品前景 .....	2
2.2	系统设计 .....	2
2.3	运行环境 .....	3
2.4	设计和实现上的约束 .....	3
2.5	假设和依赖 .....	3
3	概要设计 .....	5
3.1	系统功能框架 .....	5
3.2	功能模块说明 .....	5
3.2.1	用户： .....	5
3.2.2	管理员： .....	6
3.2.3	培训班： .....	6
4	详细设计 .....	7
4.1	前端设计 .....	7
4.1.1	设计理念 .....	7
4.1.2	实现方法 .....	7
4.1.3	关键代码： .....	8
4.1.4	效果图示： .....	9

4.2	用户登录.....	12
4.2.1	数据设计 .....	12
4.2.2	相关 jsp 实现.....	13
4.2.3	流程图 .....	14
4.3	个性化推荐.....	15
4.3.1	数据设计 .....	15
4.3.2	代码描述 .....	16
4.3.3	流程图 .....	17
4.4	培训机构课程管理.....	18
4.4.1	数据设计 .....	18
4.4.2	相关代码 .....	19
4.5	搜索中心.....	20
4.5.1	数据设计 .....	20
4.5.2	算法描述 .....	21
4.5.3	流程图 .....	22
4.6	管理员禁用普通用户或管理员.....	23
4.6.1	数据设计 .....	23
4.6.2	相关代码 .....	24
4.7	数据库设计.....	24
4.7.1	数据设计 .....	24
4.7.2	物理设计表汇总 .....	27
5	系统功能 .....	36
5.1	用户角色.....	37
5.2	功能列表与优先级.....	37

5.3	功能性需求.....	38
5.3.1	普通用户功能 .....	38
5.3.2	培训机构功能 .....	43
6	外部接口展示 .....	47
6.1	用户界面布局.....	47
6.1.1	主页 .....	47
6.1.2	课程列表页 .....	47
6.1.3	登录注册页面 .....	48
6.1.4	完善个人信息页面 .....	48
6.1.5	课程详情页面 .....	49
6.1.6	培训机构详情页面 .....	50
6.1.7	个人主页 .....	51
6.1.8	我的收藏页面 .....	52
6.1.9	培训机构登录页面 .....	53
6.1.10	培训机构注册页面 .....	53
6.1.11	培训机构管理简介页面.....	54
6.1.12	培训机构管理分类页面 .....	55
6.1.13	培训机构管理课程页面 .....	56
6.1.14	培训机构添加课程页面 .....	57
6.1.15	管理员登录页面 .....	58
6.1.16	管理员管理用户页面 .....	59
6.1.17	管理员管理培训机构页面 .....	59
7	系统架构与模块实现 .....	61
7.1	项目架构.....	61

7.2	有关模块.....	64
7.3	开发环境.....	66
7.4	测试环境.....	66
8	项目成就 .....	67
8.1	项目性能.....	67
8.2	项目安全性.....	67
8.3	个人总结.....	67
8.4	个人展望.....	68
附录.....		69
测试用例.....		69
普通用户.....		69
培训机构.....		69
管理员.....		69

# 1 绪论

## 1.1 项目背景

目前大多数人群周末更倾向宅在家中或辗转于各大商场，周末无所事事，精神生活空缺。其次，各个培训机构质量良莠不齐，想要寻找的培训班或培训机构的信息来源匮乏。另外，据了解，目前网络上此类网站鲜少，是一个很好地切入点。

利用周末或其他闲暇时间来进行学习，不仅可以学习专业技能提升自己的综合竞争力，也可以学习兴趣课程，如：茶艺、插花等，以陶冶情操，丰富精神生活，提升自己的生活质量和精神高度。

## 1.2 项目范围

“甘蔗课程推荐网站”允许培训机构在网站发布信息以及用户的查找信息，具体功能详见第二章总体描述，这一部分列出了网站 1.0 版本中实现的部分功能。

## 1.3 负责工作

在本次工程实训过程中，我主要负责借助 Spring 框键、使用 Java 语言开发系统的后端逻辑功能。



## 2 总体描述

### 2.1 产品前景

“甘蔗培训课推荐网站”是一个新网站，主要针对上班族人群，目的是为了填补上班族精神生活的空白。使他们的节假日不再宅在家或浪费在商场街头，而是有东西可学，有事可做。它改变了以往寻找培训机构或培训课程的繁琐过程，为用户打造了一个便捷的查询平台和信息提供平台。也为各个培训机构提供了一个更好地宣传自己、扩大影响力的方式。

### 2.2 系统设计

用户类	用户功能
普通用户	<ol style="list-style-type: none"><li>1. 普通用户可以完成登录、注册、完善个人信息等基本功能</li><li>2. 用户可以看到系统根据用户的信息所完成的精准推荐，如：根据课程喜好、所在职业职务、学历等进行推荐。</li><li>3. 用户可以在首页进行关键字搜索或者分类检索，得到培训机构列表或者培训课程列表。</li><li>4. 用户可以点击某一个培训机构或者培训课程查看详情。</li><li>5. 用户可以在查看详情页面选择是否进行收藏操作</li><li>6. 用户可以看到详情页面推荐的该课程所属培训机构的类似课程</li><li>7. 用户可以在详细了解之后进行评价。</li><li>8. 用户可以修改个人信息，更新自己的兴趣爱好等。</li></ol>
培训机构	<ol style="list-style-type: none"><li>1. 培训机构注册登录需要进行上传执照证件等。</li><li>2. 培训机构可以对自己网站的内容进行添加和修改</li><li>3. 培训机构可以对课程分类进行增删改</li><li>4. 培训机构可以在课程进行增删改</li><li>5. 培训机构可以随时预览管理操作后的效果</li></ol>
管理员	<ol style="list-style-type: none"><li>1. 管理员可以进行用户的禁用以及培训机构的禁用</li><li>2. 维护网站的页面效果，如进行一些美工操作。</li></ol>

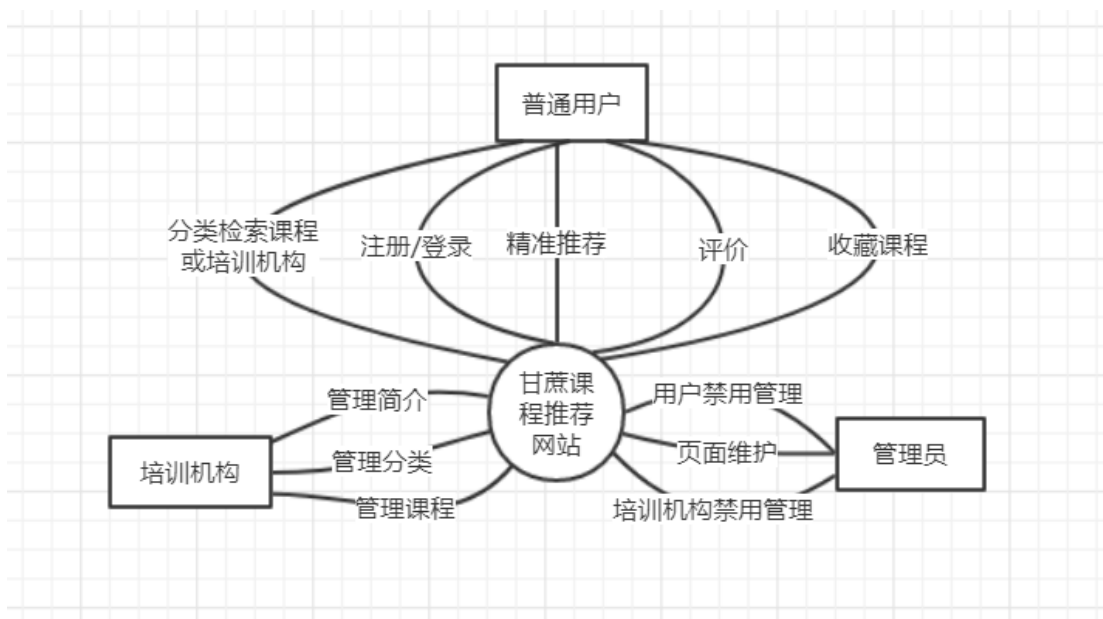


图 2.2.1 系统功能模块

## 2.3 运行环境

1. “甘蔗课程推荐网站”的操作将通过如下 WEB 浏览器完成：Google Chrome 最新版本，IE 最新版本，编译器自带浏览器；
2. “甘蔗课程推荐网站”的服务器为 Tomcat 服务器，该服务器的操作系统为 windows 和 Apache HTTP Server；
3. “甘蔗课程推荐网站”允许用户在任何地点访问该网站系统。

## 2.4 设计和实现上的约束

1. 系统的推荐功能必须用到数据分析与挖掘的技术；
2. 系统的编码必须符合统一的规范；
3. 系统的脚本语言以 JAVA 为主，特殊情况可以使用其他语言，前提是可以将代码进行整合。

## 2.5 假设和依赖

### ①假设：

- 1) 只要培训机构上传证件信息，系统将利用人工和机器结合的方式在 1-2 个工作日

内审核并予以答复

2) 培训机构对于简介和详情的内容上传之后, 管理员会在 1-2 个工作日内进行美工操作并展示在页面上。

**②依赖:**

- 1) 用户的收藏和评论依赖于登录
- 2) 用户的登录依赖于注册
- 3) 用户的修改信息依赖于登录和已有信息
- 4) 用户的评论发表依赖于评论审核通过
- 5) 培训机构修改和删除内容依赖于内容已存在
- 6) 培训机构的管理依赖于登录
- 7) 培训机构的登录依赖于注册
- 8) 培训机构的注册依赖于审核通过

### 3 概要设计

#### 3.1 系统功能框架

相关文字说明：

课程推荐系统分为用户端，培训机构端，管理员端，分别能够进行不同的操作。

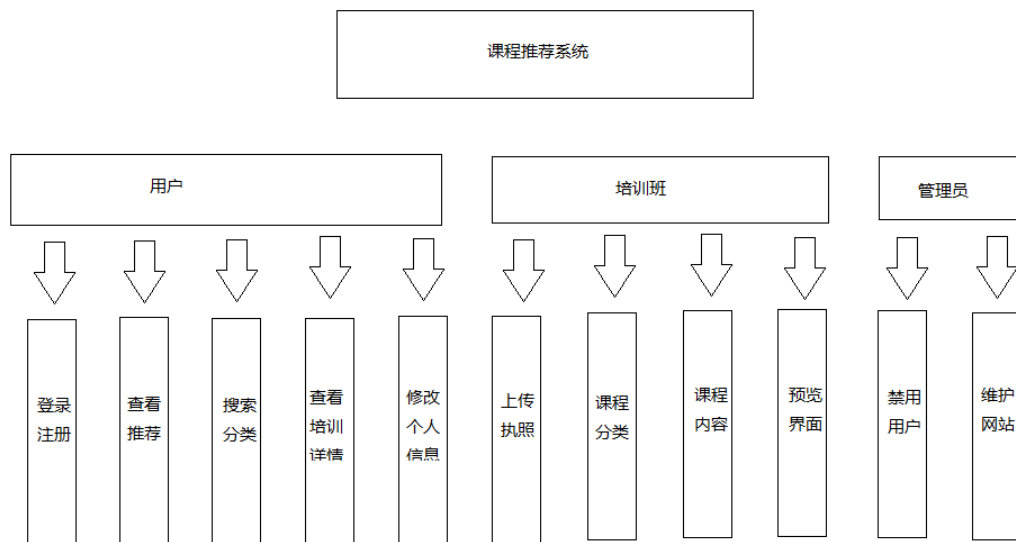


图 2.1 系统功能模块图

#### 3.2 功能模块说明

##### 3.2.1 用户：

##### 1. 注册登录

用户可以注册，登录，退出登录。登陆后可以查看自己个人信息，根据不同人的特点进行推荐。

##### 2. 查看推荐

用户可以看到系统根据用户的信息所完成的精准推荐，如：根据课程喜好、所在职业职务、学历等进行推荐。

##### 3. 搜索与分类

用户输入关键词，可以搜索出相关的课程或者培训机构，通过这个方法，可以快速找到用户自己想看的内容。

#### 4. 查看培训详情

用户点击页面上的培训班或者培训课程之后，可以进入培训详情进行查看，可以看到培训班的名称时间地点以及课程的价格讲师具体内容等等。。

#### 5. 修改个人信息

用户可以通过个人中心页面修改自己的名称个人信息，比如性别等信息，以及修改自己的学历职位等信息，推荐系统会根据这些信息来推荐出用户，想要看到适合他的课程。。

### 3.2.2 管理员：

#### 1. 禁用用户

管理员可以对不合法违规操作的用户以及培训班以及培训课程进行禁用，从而达到保护数据安全。

#### 2. 维护网站

管理员可以在后台维护网站的信息，比如网站的网址网站的一些基本信息，包括网站的一些美化内容都是可以维护的。

### 3.2.3 培训班：

#### 1. 注册登录

培训班注册的时候需要上传自己的营业执照以及相关的信息，经过审核之后方可注册成功，培训班注册成功后可以进行登录。

#### 2. 课程分类

培训班的使用者可以增加课程的分类查询分类修改分类。

#### 3. 课程内容

培训班管理员可以对培训内容的详情进行修改，增加删除查看，比如说增加课程显形的课程的价格以及课程的图片。。

## 4 详细设计

### 4.1 前端设计

#### 4.1.1 设计理念

程序员在开发过程中，发现 Servlet 做界面非常不方便，所以产生了 jsp 技术。JSP 其实是对 Servlet 进行了包装而已。

jsp + java 类（service、javabean）+ servlet，就会构成 mvc 的开发模式，mvc 模式是目前软件公司中相当通用的开发模式。

JSP 的全称是 Java Server Pages，它和 Servlet 技术一样，都是 SUN 公司定义的一种用于开发动态 web 资源的技术。

jsp 这门技术的最大的特点在于，写 jsp 就像在写 HTML，但：

它相对于 html 而言，html 只能为用户提供静态数据，而 jsp 技术允许在页面中嵌套 java 代码，为用户提供动态数据。

相比 Servlet 而言，Servlet 很难对数据进行排版，而 jsp 除了可以用 java 代码产生动态数据的同时，也很容易对数据进行排版。

#### 4.1.2 实现方法

JSP 是 HTML 代码与 Java 代码的混合体，其中 HTML 部分遵循 HTML 语法，Java 部分遵循 Java 语法

##### 1、JSP 脚本：

在 JSP 页面中写 Java 代码称为 JSP 脚本。JSP 脚本必须使用 "<% %>"括起来，Java 代码写在其中，并且必须遵循 Java 语法

##### 2、JSP 输出：

JSP 中使用 “<%= %>” 输出各种类型数据，包括 int、double、boolean、String、Object

##### 3、JSP 指令：JSP 指令用来声明 JSP 页面的一些属性等，例如编码方式、文档类型。

JSP 指令使用符号 “ <%@ %> ”。

### 4.1.3 关键代码：

#### 1. 首页导航：

```
<body>
<div class="overall">
<!--header-->
  <div class="header">
    <div class="banner">
      <!--logo-->
        <div class="logo">
          
        </div>
      <!--首页选项卡-->
        <div class="home">
          <a href="home.jsp">首页</a>
        </div>
      <!--全部课程选项卡-->
        <div class="allclass">
          <a href="../course/listAllCourse?coursePageIndex=1">全部课程</a>
        </div>
        <div class="allclass">
          <a href="../course/showCollections?collectionsPageIndex=1">个人中心
</a>
        </div>
      <!--搜索框-->
        <div class="sousuo">
          <form>
            <div class="s_img">
              <a href="${ctx }/front/search.jsp">搜索 </a>
            </div>
          </form>
        </div>
      <!--用户注册登录-->
        <div class="login_regist">
          <c:choose>
            <c:when test="${username == null }">
              <!-- 顶部未登录 -->
              <a href="javascript:void(0)" onClick="showBox()">登录</a>
              <a href="javascript:void(0)" onClick="registBox()">注册</a>
            </c:when>
            <c:otherwise>
              <!-- 顶部已登录 -->
              <a href="#"
```

## 2. 猜你喜欢:

```

    <!--猜你喜欢块-->
    <div class="first">
        <h1>
猜你喜欢</h1>

        <div class="first_love">
            <c:forEach items="{recommend }" var="s">
                <a href="../course/courseDetails?courseID={s.courseID }"></a>
            </c:forEach>
        </div>
    </div>;

```

## 3. 热门板块

```

    <!--热门综合推荐块-->
    <div class="second">
        <h1>热
门综合推荐</h1>

        <c:forEach items="{Sixrecommend }" var="six">
            <div class="second_hot">
                <div class="hot">
                    

                    <a
href="../course/courseDetails?courseID={six.courseID }"><h3>{six.courseName }</h3></a>
                    <p>{six.courseBrief}</p>
                </div>
            </div>
        </c:forEach>
    </div>

```

## 4.1.4 效果图示:



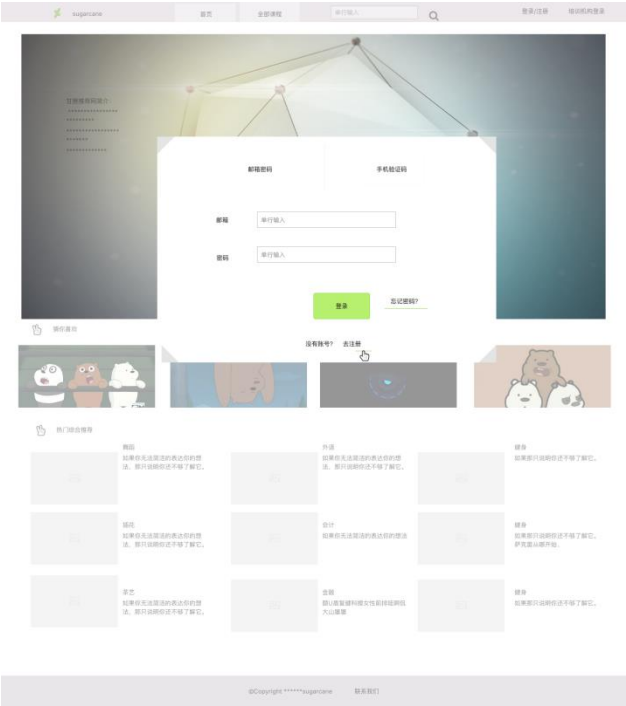


图 3.1 登录注册页背景效果



图 3.2 首页背景效果

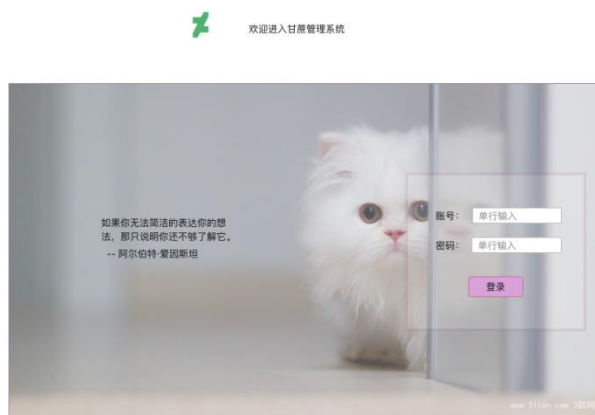


图 3.3 管理员登录背景效果

## 4.2 用户登录

### 4.2.1 数据设计

#### 1. 逻辑结构设计

实现用户登录需要用户先进行注册，注册的唯一标识为手机号码，手机号码验证可以先判断是否该号码已注册或者号码格式合理，如果可以注册，况且密码两次校验正确，则滑动滑块进行验证。验证成功则注册成功可以登录。

登录时密码错误或者是账号错误都会进行提醒，忘记密码则能去找回页面找回，找回方式为接收往该注册手机上发送的密码，如果验证正确，则可以重新设置新的密码，验证通过后再次登录。

如果用户出现某种特殊违法行为或不道德行为，管理员可以在线对其账号信息禁用，或者进行用户的信息修改，管理员查看用户信息时采用分页方式来方便查询。

#### 2. 存储结构设计

用户的登录需要记录用户的手机号码作为唯一性标识，在忘记密码的时候需要依次根

据该标识来找回密码。

#### 4.2.2 相关 jsp 实现

##### 1.注册

```

<div class="event" id="regist_box" style="display: none;">
    <div class="regist">
        <div class="title">
            <span class="t_txt">注册</span>
            <span class="del"
onClick="deleteRegist()">X</span>
        </div>
        <form action="" method="post">
            <input type="text" name="" id="email_regist"
value="" placeholder="请输入邮箱" class="email-b" onblur="check_box()" />
            <div class="bcd" align="center"></div>
            <input type="password" name="" id="pwd_regist"
value="" placeholder="密码为6-16位包含数字、字母、下划线" onblur="check()" />
            <div id="spinfo2" align="center"></div>
            <input type="password" name="" id="pwd_pwd"
value="" placeholder="再次输入密码" onblur="check_again()">
            <div id="spinfo3" align="center"></div>
            <input type="text" name="" id="petname" value=""
placeholder="昵称"/>
            <input type="text" name="" id="phone" value=""
placeholder="手机" onblur="Mous()" />
            <div id="spinfo1" align="center"></div>
            <input type="submit" name="" id="regist_1"
value="注册" class="btn">
        </form>
    </div>
</div>

```

##### 2.登录

```

<!--登录框-->
<div class="login">
    <form action="{ctx }/admin/login" method="post">
        <div>账号: <input type="text" value="单行输入" name="adminName"
class="text"/></div>
        <div class="pw">密码: <input type="password" value="单行输入"
name="password" class="text"/></div>
        <input type="submit" value="登录" class="submit"/>
    </form>
</div>

```

</div>

### 3.分页

```
String pageString=request.getParameter("page");
    if(pageString==null||pageString.equals("")){
        pageString="0";    }
    int mypage=Integer.valueOf(pageString);
    huiyuanservice huiyuans=new huiyuanservice();
    List<huiyuan> huiyuans2;
    huiyuans2=huiyuans.selecthuiyuan();
    int length=huiyuans2.size()/10;
    for(int i=0*mypage;i<huiyuans2.size()&&i<(mypage+1)*10;i++){
        huiyuan huiyuan3=huiyuans2.get(i); %>
        <tr class="TableDetail1">
            <td align="center"></td>
            <td align="center"><%=huiyuan3.getHuiyuantell() %></td>
            <td align="center"><%=huiyuan3.getHuiyuannname() %></td>
            <td align="center"><%=huiyuan3.getHuiyuanbi() %></td>
            <td align="center"><%=huiyuan3.getHuiyuansex() %></td>
```

#### 4.2.3 流程图

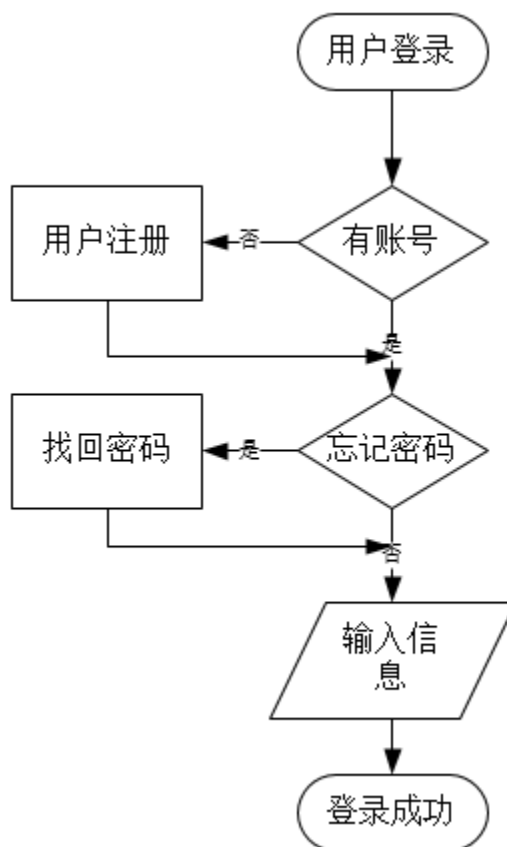


图.4.4 用户登录流程图

## 4.3 个性化推荐

### 4.3.1 数据设计

#### 1. 逻辑结构设计

课程的个性化推荐的基本逻辑是用户完善用户的个人信息，比如说职业学历以及个人喜好，之后提交给系统，后台系统根据用户的这些基本禁区为去匹配相应的课程查询出来，最后将这些课程显示在首页，猜你喜欢板块当中。

这样一来，就可以实现所有的课程推荐系统，其实是根据用户的个人信息和后台系统数据库中已有的课程数据相匹配，得出的结果，假如说用户不想要某论课程的时候，只需要去个人信息当中修改自己对应的信息，比如说我的收藏，我的喜欢的一些课程，减少该内容就可以减少类似的推荐。

每次单用户登录的时候会进入到首页，这个时候西藏会去后台数据库中通过申述语句，首先查取出用户的一些关键信息，然后通过模糊查询去有的课程数据中去匹配课程的一些标签，以及一些字段模糊查询出来的那些课程，数据就是用户适合用户想用户想要看到的数据。。

## 2. 存储结构设计

这一套推荐系统当中，总共涉及到的表有三张，依然是用户表，但是课程表还有战士用户喜好表，在用户表当中有抽这么一个字段职务职位，这个字段是代表用户的职位以及学历，这个字段代表用户目前所拥有的最高学历。。

课程表当中有一个字段叫做课程类型，这个字段是用来区分不同的课程对应不同的类别，适合什么样的人学习而进行分类的，不同的课程也会打上不同的标签，方便用户更宽更快的寻找。

### 4.3.2 代码描述

#### 1. 算法简述

用户登录之后，会根据单体浏览器中已存在的 session 去服务器中匹配相对应的 sessionid 的，从而来确定用户的身份，服务器获得用户身份之后，会去数据库中查询用户已有的关键词信息，然后根据这些关键信息再去查询用户喜欢的，适合用户的一些课程，再通过 servlet 返回数据给前端 JSP 页面，JSP 页面接收数据后在页面上渲染出来。。

#### 2. 代码实现

##### 1. Jsp 代码实现：

```
<div class="body_left_classify">
    <ul>
        <li><a
href="../course/listAllCourseByCourseType?coursePageIndex=1&courseTypeID=${courseType3
.courseTypeID}">${courseType3.courseTypeName }</a>
        <ul>
            <c:forEach items="${publicCourseTypeLists3 }"
var="publicCourseType3">
```

servlet 关键代码实现:

```
@RequestMapping("/toIndex")
```

```
public void ToIndex(@RequestParam(value = "coursePageIndex", defaultValue = "1")
```

```
Integer coursePageIndex,
```

```
HttpServletRequest response, HttpSession session) throws IOException {
```

```
if (session.getAttribute("userId") == null) {
```

```
List<Course> list = this.courseServiceImpl.mainfest();
```

```
session.setAttribute("recommend", list);
```

```
} else {
```

```
testPython test = new testPython();
```

```
List<String[]> list = test.recommend();
```

```
int a = this.courseServiceImpl.mainfestMostCourseById((int)
```

```
session.getAttribute("userId"));
```

```
int f[] = new int[4];
```

```
int r = 0;
```

```
int j = 1;
```

```
for (String[] strings : list) {
```

```
int b = (Integer.parseInt(strings[0]) + 1);
```

```
if (b == a) {
```

```
for (; r <= 3 && j <= 4; r++, j++) {
```

```
f[r] = (Integer.parseInt(strings[j]) + 1);
```

```
}
```

```
}
```

```
}
```

#### 4.3.3 流程图



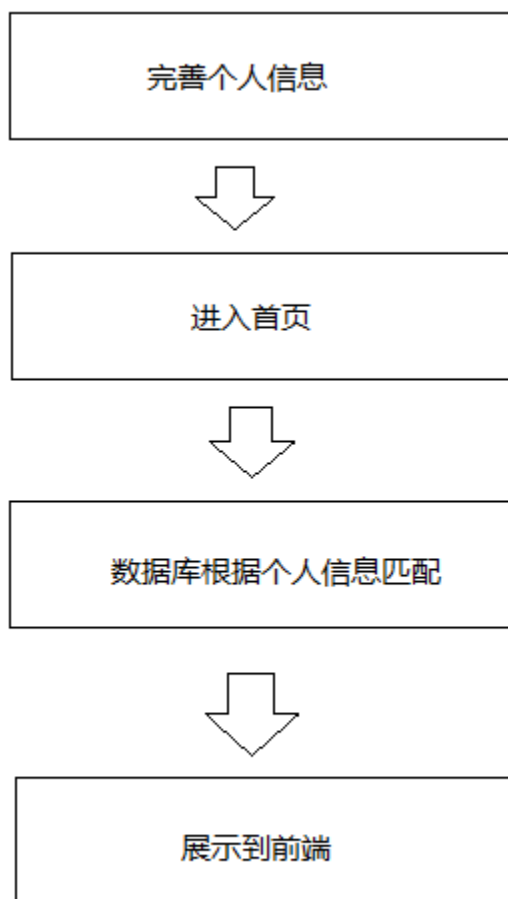


图 3.5 个性化推荐流程图

## 4.4 培训机构课程管理

### 4.4.1 数据设计

#### 1. 逻辑结构设计

培训机构可以对自己拥有的内容进行增删改查的操作，比如说培训机构可以新增课程修改课程删除课程以及查看课程以及编辑课程的详细内容，除此之外，培训课程培训机构还可以对课程进行分类，新增分类修改分类查看分类和删除分类。。

#### 2. 存储结构设计

培训机构相关的表组成了不同培训机构的基本信息名称法人地址等等，还有一张表是用来存储培训机构的课，课程的课程，单中的字段有课程的名称课程的价格课程的详情以及课程的类型等等。

#### 4.4.2 相关代码

```
@RequestMapping("/addOneCourse")
public void AddOneCourse(@RequestParam("courseName") String courseName,
    @RequestParam("price") double price,
    @RequestParam("video") String video, @RequestParam("phoneNumber")
String phoneNumber,
    @RequestParam("teacher") String teacher,
    @RequestParam("introductionImg1") String introductionImg1,
    @RequestParam(value = "sellerCourseTypeID", defaultValue = "1")
Integer sellerCourseTypeID,
    @RequestParam(value = "publicTypeID", defaultValue = "1") Integer
publicTypeID,
```

3.1.3流程图

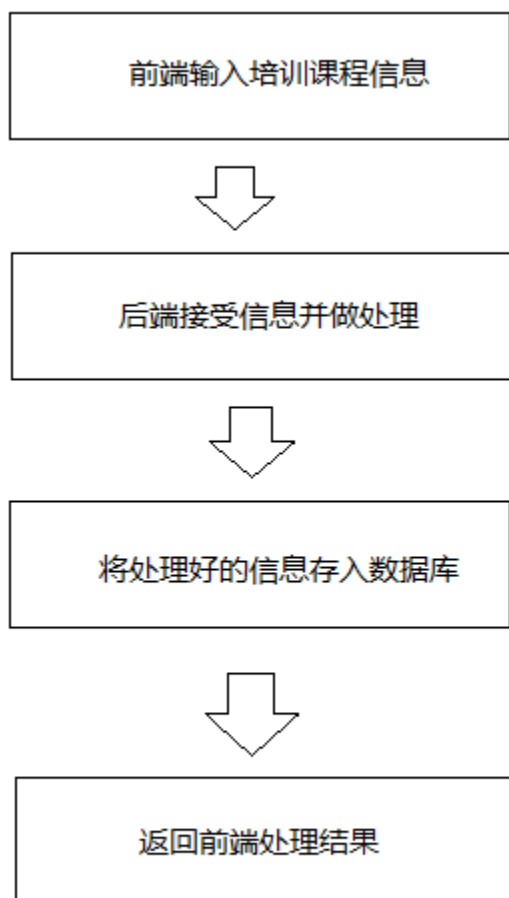


图 3.6 添加课程流程图

## 4.5 搜索中心

### 4.5.1 数据设计

#### 1. 逻辑结构设计

搜索中心是给用户用来搜索课程，以及他感兴趣的培训班来使用的课程名称以及培训班的名称，以及它们相关的标签等等，但用户对人比较感兴趣的时候，直接对搜索框中进行搜索，就可以直接跳到相关的课程，这里搜索采用的是模糊查询，通过用户输入的关键字进行模糊匹配。。

## 2. 存储结构设计

数据存储的类型主要采用的是四种类型，主要是对两张表进行搜索，一张表示课程表当中的课程名称，另外，一张表是培训机构中的培训机构的名称，这两张表中的搜索字段都会添加索引，用来优化查询的速度。。

### 4.5.2 算法描述

#### 1. 操作简述

通过导入 common-fileupload 和 common-io 两个包来实现文件的上传，使用 file 类型的 input 实现打开计算机中的本地文件，form 的 enctype 的值设为"multipart/form-data"，将上传路径设置为绝对路径来避免文件发布到 tomcat 的临时文件中而重启的消失。读取 form 内的每个 input 对象时判断该对象是 file 类型的还是普通的输入框类型，判断后如果是 file 类型，则去提交，否则提取出文件的 name 和 value 值，然后将 value 进行数据库的更新操作。

为了方便将文件选择框进行 css 美化，使用两个 input 类型的显示对象，分别设为 button 和 text 类型，添加点击事件。

#### 2.重要代码

##### 1. jsp:

```
$(function(){
    $(".bodys form").not(":first").hide();
    $(".searchbox ul li").mouseover(function(){
        var index = $(this).index();
        if(index==0){
            $(this).find("a").addClass("style1");
            $(".li").eq(1).find("a").removeClass("style2");
        }
        if(index==1){
            $(this).find("a").addClass("style2");
            $(".li").eq(0).find("a").removeClass("style1");
        }
        var index=$(this).index();
        $(".bodys form").eq(index).show().siblings().hide();
    });
});

function checkSearch1(){
```

```
    var text=document.getElementById("content_course").value;
    if(text==""){
        alert("搜索框为空");
        return false;
    }
    return true;
}
function checkSearch2(){
    var text=document.getElementById("content_org").value;
    if(text==""){
        alert("搜索框为空");
        return false;
    }
    return true;
}
</script>
<style type="text/css">
*{margin:0;padding:0;list-style-type:none;}
a,img{border:0;}
```

#### 4.5.3 流程图

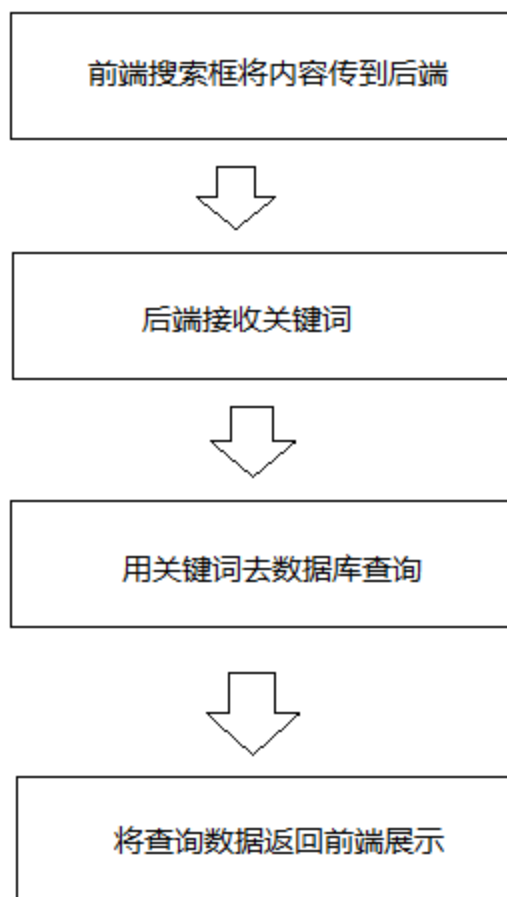


图 3.7 搜索流程图

## 4.6 管理员禁用普通用户或管理员

### 4.6.1 数据设计

#### 1. 逻辑结构设计

管理员禁用普通无辜者管理员这个操作是用来防止某些用户出现行为，之后管理员可以它采用金融的方式来防止它产生更大的危害。当然有时候也会出现误操作，比如说将一个正常的用户给禁用了，经过调查之后，他是账号被盗用了等情况，鉴于此情况可以采用加禁用的用户恢复正常的方式。

#### 2. 存储结构设计

这里用户的启用禁用的状态是通过一个字段的标识来进行完成的，给用户以及培训机

构增加一个 states 的状态，通过状态的启用和停用来判断这个用户或者培训机构是否是正常状态，或者是禁用状态，从此保保存他们的信息。

#### 4.6.2 相关代码

```
/**
 * forbid禁用操作
 * @author feng
 * @date 2018.6.13
 */
@RequestMapping("/forbidUser")
public String forbidUser(HttpServletRequest request,
@RequestParam(value="userID",required=false) String userIDString) {
    String listUserIDforbid = this.adminLoginServiceImpl.listUserforbid();
    HttpSession session = request.getSession();
    List<UserInfo> listUser = (List<UserInfo>) session.getAttribute("listUser");
    if(userIDString != null) {
        String delforbid = listUserIDforbid;
        String[] listuserforbid = listUserIDforbid.split(",");
        String[] userIDs = userIDString.split(",");
        for(String userforbid :listuserforbid) {
            for(String userIDStr : userIDs) {
                if(userforbid.equals(userIDStr)) {
                    delforbid = delforbid.replace(userIDStr, "");
                }
            }
            Integer userID = Integer.valueOf(userIDStr);
            UserInfo user =
this.adminLoginServiceImpl.findUserById(userID);
            UserLogin ul = user.getUserLogin();
            Integer forbid = ul.getForbid();
            if(forbid == null || forbid == 0) {
                ul.setForbid(1);
            }
        }
    }
}
```

### 4.7 数据库设计

#### 4.7.1 数据设计

##### 1. 背景

名称	说明
数据库名称	sugarcane
数据库用户名	root

密码	root
数据库系统	MySQL 5.0
客户端连接工具	SQLyog-64bit

表 3-1 数据库配置表

2. E-R 图

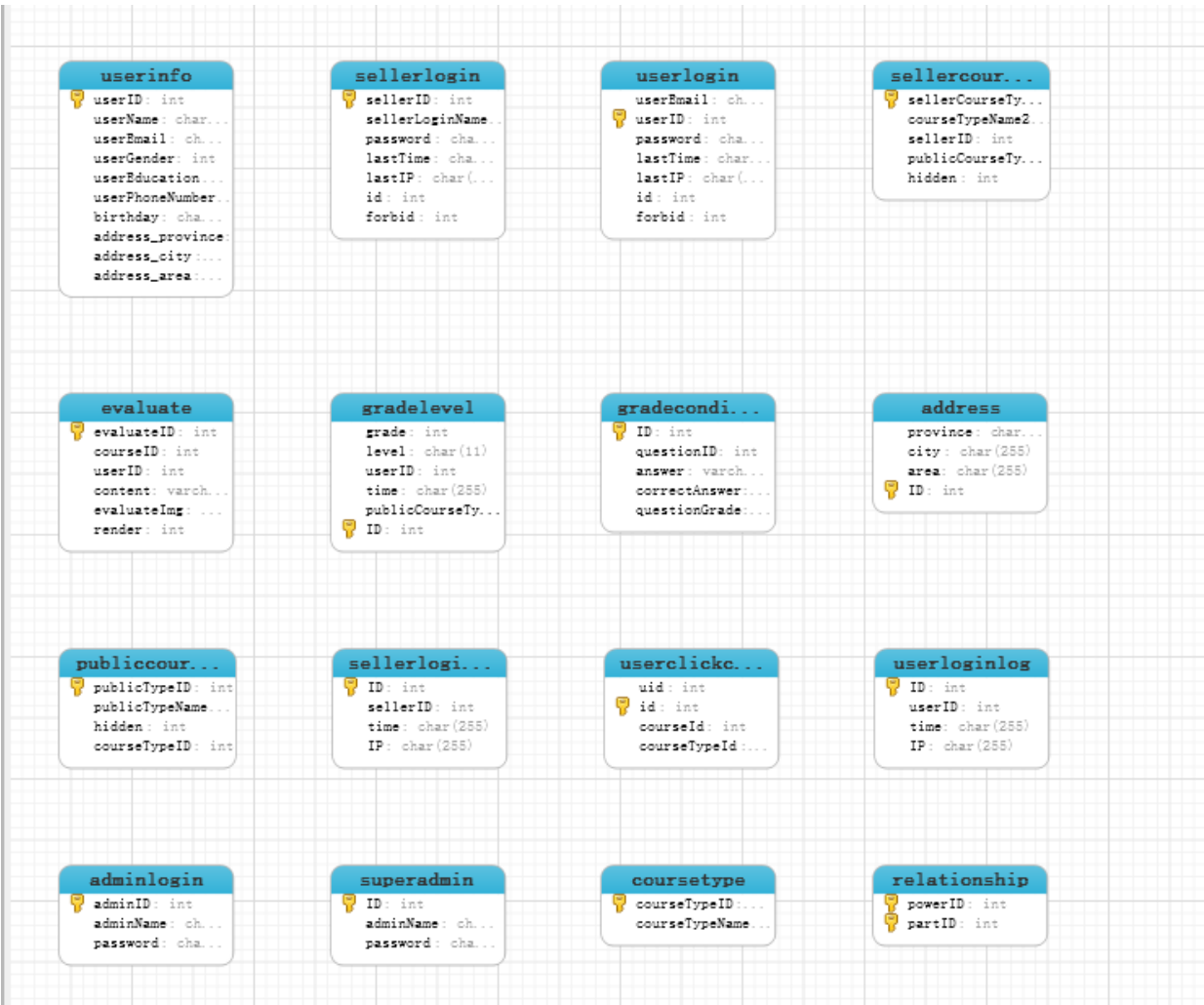


图 3.8 E-R 图

3. 存储结构设计

在编码过程中，如果 MySQL 数据结构设计不好的话，会大大影响开发人员编码效率。比如说 MySQL 数据库表设计不规范，创建时间字段设计成 cjsj，创建者字段设计成 cjr 或者 cjz。这样的数据库表可读性和表意性相当差。下面我们就来讲讲如何规范设计数据库结构。



## 数据库结构优化

数据库结构优化的目的有哪些？

减少数据冗余。

尽量避免数据维护中出现更新，插入，删除异常。插入异常是指如果表中的某个实体随着另一个实体而存在。更新异常是指如果更改表中的某个实体的单独属性时，需要对多行进行更新。删除异常是指如果删除表中的某一实体则会导致其他实体消失。

节约数据查询空间。

## 范式化与反范式化

反范式化设计就是为了性能和读取效率的考虑而适当的对数据库设计范式进行违反，而允许存在少量数据冗余。换句话说反范式化就是使用空间换时间。

范式化的优点：

- 1.可以减少数据冗余。
- 2.范式化的更新操作比反范式化要快。
- 3.范式化的表同样比反范式化的表要小。

范式化的缺点：

- 1.关联查询。
- 2.更难于索引优化。

反范式化优点：

- 1.减少表的关联。
- 2.更好的索引优化，覆盖索引。

反范式化缺点：

- 1.存在数据冗余及数据维护异常。

## 2.对数据的修改需要更多的成本。

### 如何为表选择字段类型

在物理设计中，我们要做到可读性，表意性，长名性。

当我们为表进行物理设计时，常常为对表中的字段选择合适的数据类型进行纠结。当一个列可以选择多种数据类型时，应该优化考虑数字类型，其次是日期或者二进制类型，最后是字符串类型。对于相同级别的数据类型，应该优化考虑占用空间小的数据类型。

Innodb 一页是 16K。

tinyint 占 1 个字节

smallint 占 2 个字节

mediumint 占 3 个字节

int 占 4 个字节

bigint 占 8 个字节

float 占 4 个字节，非精确

double 占 8 个字节，非精确

decimal 每 4 个字节存 9 个数据，小数点占 1 个字节。比如 decimal(18,9)需要 9 个字节来存储，最多支持 65 个数字。精确

year 占 1 个字节

time 占 3 个字节

date 占 3 个字节

datetime 占 8 个字节

timestamp 占 4 个字节

## 4.7.2 物理设计表汇总

MySQL 数据库表结构

## 1、表名：address

注释：

引擎： InnoDB

编码： utf8\_general\_ci

类型： BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	province		char	255	YES	null
2	city		char	255	YES	null
3	area		char	255	YES	null
4	ID		int	null	NO	null

## 2、表名：adminlogin

注释：

引擎： InnoDB

编码： utf8\_general\_ci

类型： BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	adminID		int	null	NO	null
2	adminName		char	255	YES	null
3	password		char	255	YES	null

## 3、表名：course

注释：

引擎： InnoDB

编码： utf8\_general\_ci

类型： BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	courseID		int	null	NO	null
2	courseName		char	255	YES	null
3	courseBrief		char	255	YES	null

4	price		double	null	YES	null
5	phoneNumber		char	255	YES	null
6	introductionImg1		char	255	YES	null
7	introductionImg2		char	255	YES	null
8	introductionImg3		char	255	YES	null
9	video		char	255	YES	null
10	sellerID		int	null	YES	null
11	sellerCourseTypeId		int	null	YES	null
12	pub_publicTypeId		int	null	YES	null
13	teacher		char	255	YES	null

## 4、表名：coursetype

注释：

引擎：

InnoDB

编码：utf8\_general\_ci

类型：

BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	courseTypeID		int	null	NO	null
2	courseTypeName		char	255	NO	null

## 5、表名：evaluate

注释：

引擎：

InnoDB

编码：utf8\_general\_ci

类型：

BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	evaluateID		int	null	NO	null
2	courseID		int	null	YES	null
3	userID		int	null	YES	null

4	content		varchar	255	YES	null
5	evaluateImg		char	255	YES	null
6	render		int	null	YES	null

## 6、表名: gradecondition

注释:

引擎: InnoDB

编码: utf8\_general\_ci

类型: BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	ID		int	null	NO	null
2	questionID		int	null	YES	null
3	answer		varchar	255	YES	null
4	correctAnswer		varchar	255	YES	null
5	questionGrade		int	null	YES	null

## 7、表名: gradelevel

注释:

引擎: InnoDB

编码: utf8\_general\_ci

类型: BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	grade		int	null	YES	null
2	level		char	11	YES	null
3	userID		int	null	YES	null
4	time		char	255	YES	null
5	publicCourseTypeID		int	null	YES	null
6	ID		int	null	NO	null

## 8、表名：publiccoursetype

注释： 引擎： InnoDB  
 编码： utf8\_general\_ci 类型： BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	publicTypeID		int	null	NO	null
2	publicTypeName		char	255	YES	null
3	hidden		int	null	YES	null
4	courseTypeID		int	null	YES	null

## 9、表名：questionbank

注释： 引擎： InnoDB  
 编码： utf8\_general\_ci 类型： BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	questionID		int	null	NO	null
2	questionStem		varchar	255	YES	null
3	options		char	255	YES	null
4	answer		char	11	YES	null
5	courseTypeID		int	null	YES	null

## 10、表名：relationship

注释： 引擎： InnoDB  
 编码： utf8\_general\_ci 类型： BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
----	------	------	------	----	-----	-----

1	powerID		int	null	NO	0
2	partID		int	null	NO	0

## 11、表名: sellercoursetype

注释:

引擎: InnoDB

编码: utf8\_general\_ci

类型: BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	sellerCourseTypeID		int	null	NO	null
2	courseTypeName2		char	255	YES	null
3	sellerID		int	null	YES	null
4	publicCourseTypeID		int	null	YES	null
5	hidden		int	null	YES	null

## 12、表名: sellerinfo

注释:

引擎: InnoDB

编码: utf8\_general\_ci

类型: BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	sellerID		int	null	NO	null
2	sellerName		char	255	YES	null
3	sellerEmail		char	255	YES	null
4	sellerPhoneNumber		char	255	YES	null
5	brief		char	255	YES	null
6	numberOfEmployees		int	null	YES	null
7	educationBureauApproved		char	255	YES	null

8	proofOfHouse		char	255	YES	null
9	fireSafetyCertificate		char	255	YES	null
10	businessLisense		varchar	255	YES	null
11	address_province		char	255	YES	null
12	address_city		char	255	YES	null
13	address_area		char	255	YES	null
14	address_detail		char	255	YES	null
15	qualified		int	null	YES	null

## 13、表名: sellerlogin

注释:

引擎: InnoDB

编码: utf8\_general\_ci

类型: BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	sellerID		int	null	NO	null
2	sellerLoginName		char	255	YES	null
3	password		char	255	YES	null
4	lastTime		char	255	YES	null
5	lastIP		char	255	YES	null
6	id		int	null	YES	null
7	forbid		int	null	YES	null

## 14、表名: sellerloginlog

注释:

引擎: InnoDB

编码: utf8\_general\_ci

类型: BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
----	------	------	------	----	-----	-----



1	ID		int	null	NO	null
2	sellerID		int	null	YES	null
3	time		char	255	YES	null
4	IP		char	255	YES	null

## 15、表名：superadmin

注释：

引擎： InnoDB

编码： utf8\_general\_ci

类型： BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	ID		int	null	NO	null
2	adminName		char	255	YES	null
3	password		char	255	YES	null

## 16、表名：userclickcourse

注释：

引擎： MyISAM

编码： utf8\_general\_ci

类型： BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	uid		int	null	NO	null
2	id		int	null	NO	null
3	courseId		int	null	NO	null
4	courseTypeId		int	null	NO	null

## 17、表名：usercollections

注释：

引擎： InnoDB

编码: utf8\_general\_ci

类型: BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	courseID		int	null	YES	null
2	userID		int	null	YES	null
3	collectTime		char	255	YES	null
4	collecting		int	null	YES	null
5	collected		int	null	YES	null
6	collectionCount		int	null	YES	null
7	ID		int	null	NO	null

## 18、表名: userinfo

注释:

引擎: InnoDB

编码: utf8\_general\_ci

类型: BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	userID		int	null	NO	null
2	userName		char	255	YES	null
3	userEmail		char	255	YES	null
4	userGender		int	null	YES	null
5	userEducation		char	255	YES	null
6	userPhoneNumber		int	null	YES	null
7	birthday		char	255	YES	null
8	address_province		char	255	YES	null
9	address_city		char	255	YES	null
10	address_area		char	255	YES	null

## 19、表名：userlogin

注释： 引擎： InnoDB  
 编码： utf8\_general\_ci 类型： BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	userEmail		char	255	YES	null
2	userID		int	null	NO	null
3	password		char	255	YES	null
4	lastTime		char	255	YES	null
5	lastIP		char	255	YES	null
6	id		int	null	YES	null
7	forbid		int	null	YES	null

## 20、表名：userloginlog

注释： 引擎： InnoDB  
 编码： utf8\_general\_ci 类型： BASE TABLE

序号	字段名称	字段描述	字段类型	长度	允许空	缺省值
1	ID		int	null	NO	null
2	userID		int	null	YES	null
3	time		char	255	YES	null
4	IP		char	255	YES	null

## 5 系统功能

5.1 用户角色

本项目的用户分为普通用户、培训机构、管理员这三类。

普通用户：可以在网站系统注册账号、登陆系统，进行相关推荐的查看、课程查询、培训机构查询、加入收藏、参与评论等操作，方便用户操作，为普通用户提供查询信息平台。

培训机构：可以在网站系统提供、修改、删除课程信息、培训机构信息等。

管理员：是有权进行后台管理的所有人员的统称，是在用户申请预购该软件时对其进行信息审核的人员。根据用户需求给其分配相应模块，管理用户信息等。

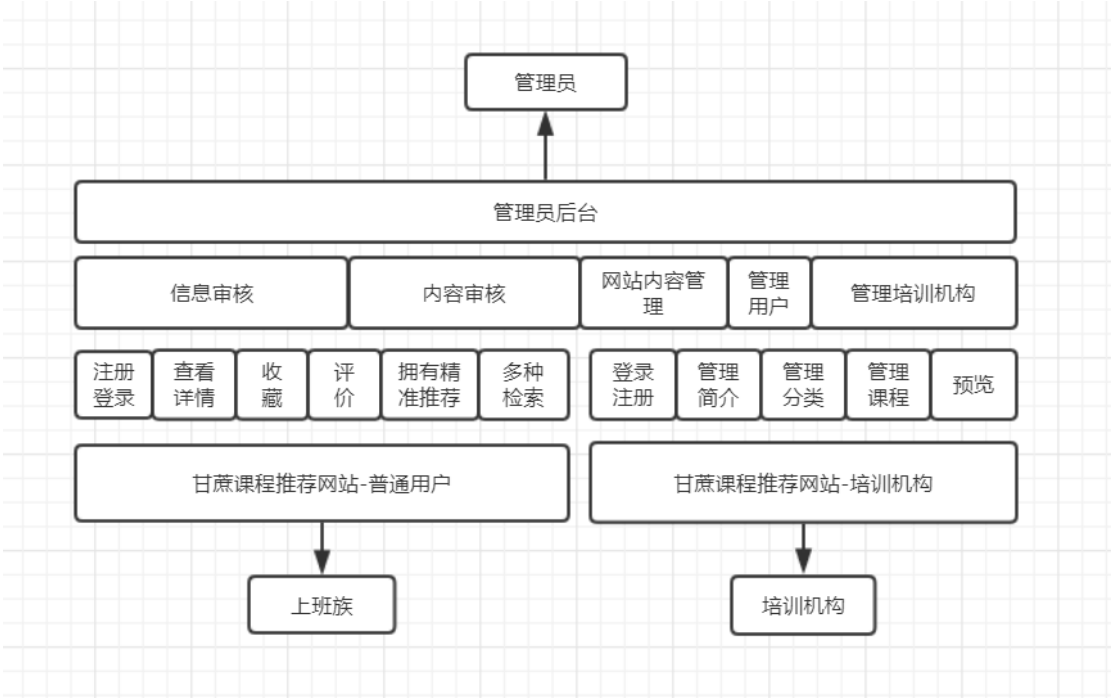


图 5.1.1 系统用户角色

5.2 功能列表与优先级

用户登录	高[1]
用户注册	高[1]
精准推荐	中[2]
分类检索	中[2]
搜索	中[2]

机构详情	中[2]
课程详情	中[2]
评论信息	低[3]
加入收藏	低[3]
培训机构注册	高[1]
培训机构登录	高[1]
管理分类	中[2]
管理简介	中[2]
管理课程	中[2]
预览	低[3]
管理员登录	高[1]
禁用用户账号	中[2]
禁用培训机构	中[2]

5.3 功能性需求

5.3.1 普通用户功能

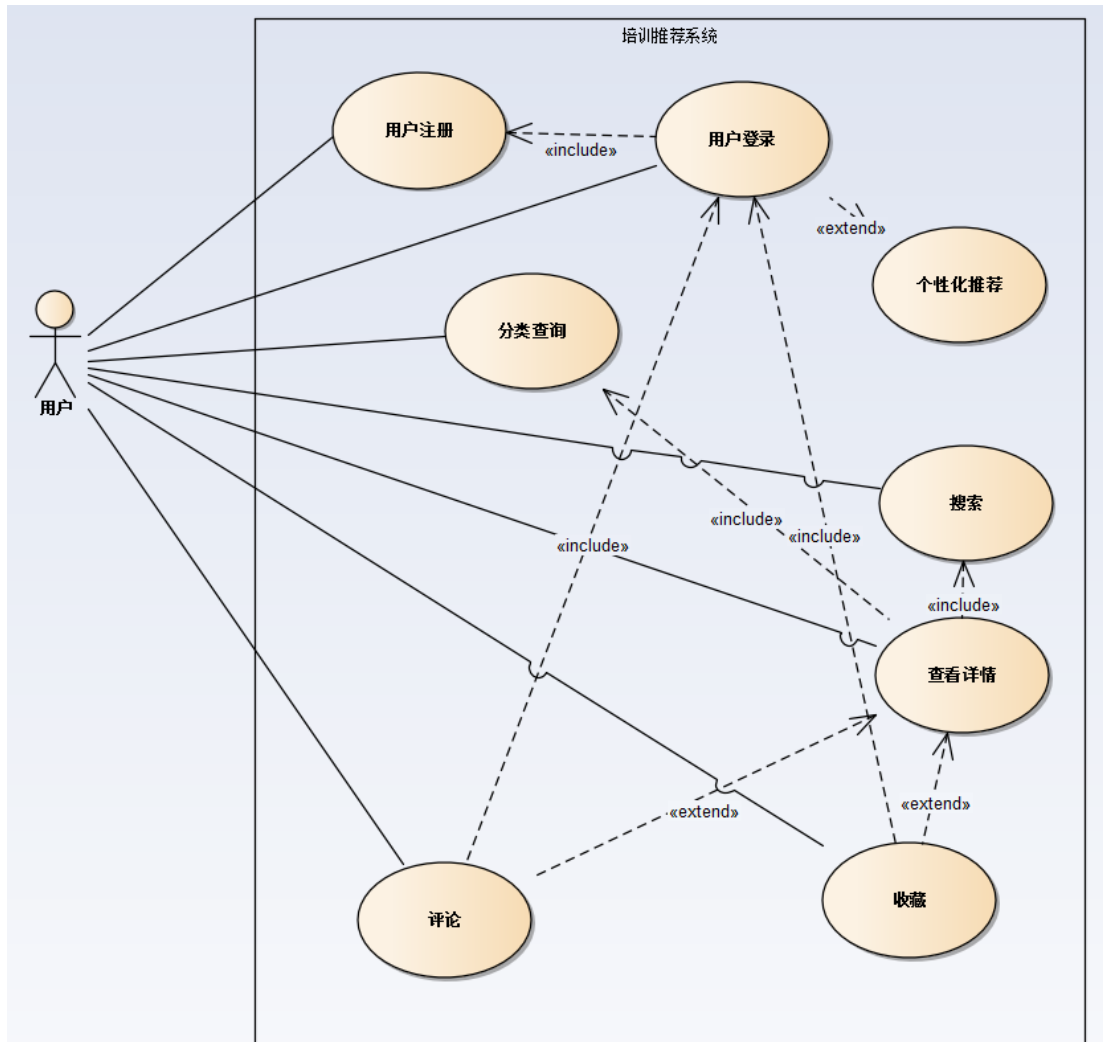


图 5.3.1 普通用户功能

### 1) 用例：注册

用例名称：注册

干系人利益：

用户：给予用户更多的权限，使其拥有更好的用户体验

基本路径：

- ①用户选择注册业务
- ②系统进入注册页面
- ③用户填写用户信息并提交
- ④系统验证用户信息
- ⑤系统跳转到登陆页面

## ⑥系统激活登录用例

扩展路径:

①用户输入的用户名不符合规则, 系统提示

②如果用户输入登录信息, 系统进入登录用例

业务规则:

①用户名不能重复

## 2) 用例: 登录

**用例名称: 登录**

干系人利益:

用户: 维护用户的正当使用权利

管理员: 便于后台管理

基本路径:

①用户选择登录业务

②系统显示登录页面

③用户填写用户信息并提交

④系统验证用户信息

⑤系统启动“个性化推荐”用例扩展点

⑥系统跳转进入首页

扩展路径:

①用户信息非法, 系统提示

②用户选择个性化推荐, 系统进入“个性化推荐”用例

业务规则:

①用户名必须存在, 密码必须与用户名匹配

## 3) 用例: 个性化推荐

**用例名称: 个性化推荐**

干系人利益:

用户: 获得精准推荐服务

基本路径:

①用户完善用户信息并提交

②系统根据用户信息查询课程

③系统将查询的课程结果显示在首页

#### **4) 用例：分类查询**

**用例名称：分类查询**

干系人利益：

用户：方便用户快速查找

基本路径：

①用户选择分类条件

②系统跳转到分类页面

③用户查看分类结果

④系统激活“查看详情”用例扩展点

扩展路径：

如果用户选择查看详情，系统进入查看详情用例

#### **5) 用例：搜索**

**用例名称：搜索**

干系人利益：

用户：快速搜索想要的信息

培训机构：提高曝光率

基本路径：

①用户点击搜索框

②用户输入搜索内容

③系统根据关键词进行搜索

④系统显示搜索结果

⑤系统激活“查看详情”用例扩展点

扩展路径：

①未搜索到内容，系统提示

②用户选择查看详情，系统进入“查看详情”用例

#### **6) 用例：查看详情**

**用例名称：查看详情**



干系人利益:

用户: 方便查看详情, 操作简单

基本路径:

①用户选择查看详情

②系统显示详情页面

③系统激活“收藏”用例的扩展点

④系统激活“评论”用例的扩展点

扩展路径:

①如果用户选择收藏, 系统进入收藏用例

②如果用户选择评论, 系统进入评论用例

## 7) 用例: 收藏

**用例名称: 收藏**

干系人利益:

用户: 方便用户查看喜欢的培训班信息, 操作简单

基本路径:

①用户选择收藏业务

②系统显示已收藏

## 8) 用例: 评论

**用例名称: 评论**

干系人利益:

用户: 方便用户获取客观评价信息

培训机构: 便于得到反馈

基本路径:

①用户在详情页输入评论

②系统筛选评论内容

③系统跳转到详情页面

扩展路径:

如果评论内容不合法, 系统提示

业务规则:

评论内容不能包含低俗词汇、政治倾向等

### 5.3.2 培训机构功能

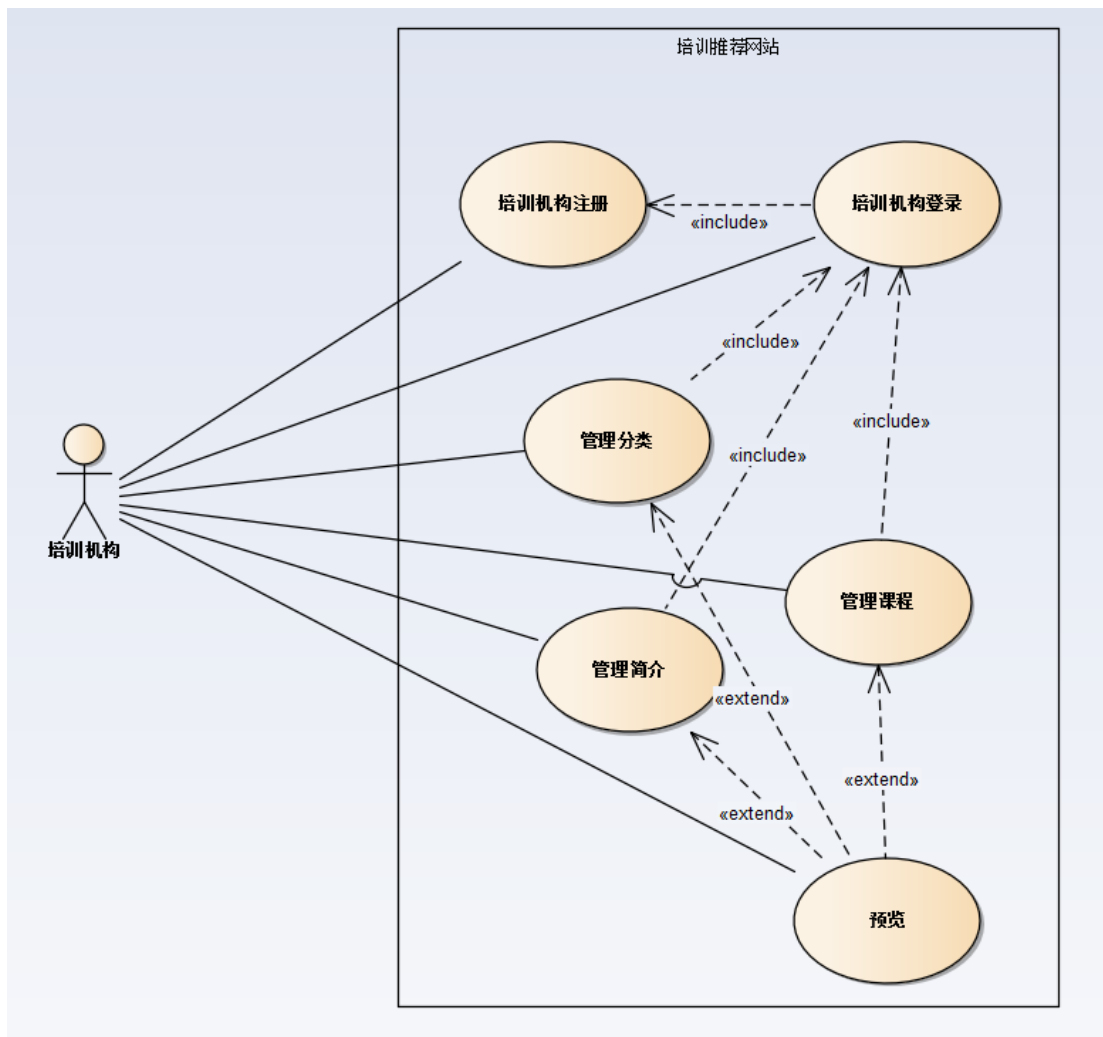


图 5.3.2 培训机构功能

#### 1) 用例：注册

用例名称：培训机构注册

干系人利益：

培训机构：给予培训机构更多的使用权限，使其拥有更好的使用体验

基本路径：

①培训机构选择注册业务

②系统进入注册页面

③培训机构填写培训机构信息并提交

④系统验证培训机构信息

⑤系统跳转到登陆页面

⑥系统激活登录用例

扩展路径:

①培训机构上传的信息不合法, 系统提示

②培训机构选择登录业务, 系统进入登录用例

业务规则:

①上传信息必须真是无误

## **2) 用例: 登录**

**用例名称: 培训机构登录**

干系人利益:

培训机构: 维护培训机构的正当使用权利

基本路径:

①培训机构选择登录业务

②培训机构填写培训机构信息并提交

③系统验证培训机构信息

④系统跳转进入首页

扩展路径:

①培训机构信息非法, 系统提示

业务规则:

①培训机构用户名必须存在, 密码必须与培训机构用户名匹配

## **3) 用例: 管理分类**

**用例名称: 管理分类**

干系人利益:

培训机构: 方便培训机构随时更新信息

用户: 方便获得最新的培训信息

基本路径:

①培训机构在培训机构管理首页选择管理分类业务

②系统显示管理分类页面

③培训机构更新分类并保存

④系统保存培训机构信息并显示管理分类页面

⑤系统激活“预览”用例扩展点

扩展路径:

如果培训机构选择预览, 系统进入预览用例

#### **4) 用例: 管理简介**

**用例名称: 管理简介**

干系人利益:

培训机构: 操作简单方便

用户: 方便用户获取信息

基本路径:

①培训机构在培训机构管理首页选择管理简介业务

②系统跳转到管理简介页面

③培训机构更新信息并保存

④系统保存培训机构信息并跳转到管理简介页面

⑤系统激活“预览”用例扩展点

扩展路径:

如果培训机构选择预览, 系统进入预览用例

#### **5) 用例: 管理课程**

**用例名称: 管理课程**

干系人利益:

培训机构: 操作方便简洁

用户: 避免得到过期信息

基本路径:

①培训机构选择管理课程业务

②系统显示管理课程页面

③培训机构更新课程信息

④系统保存课程信息

⑤系统激活“预览”用例扩展点

扩展路径:

如果培训机构选择预览，系统进入预览用例

**6) 用例：预览**

**用例名称：预览**

干系人利益:

培训机构：方便查看效果

基本路径:

①用户选择预览业务

②系统跳转到首页

③培训机构查看页面效果

6 外部接口展示

6.1 用户界面布局

6.1.1 主页



图 6.1.1 4.1.1 主页

6.1.2 课程列表页

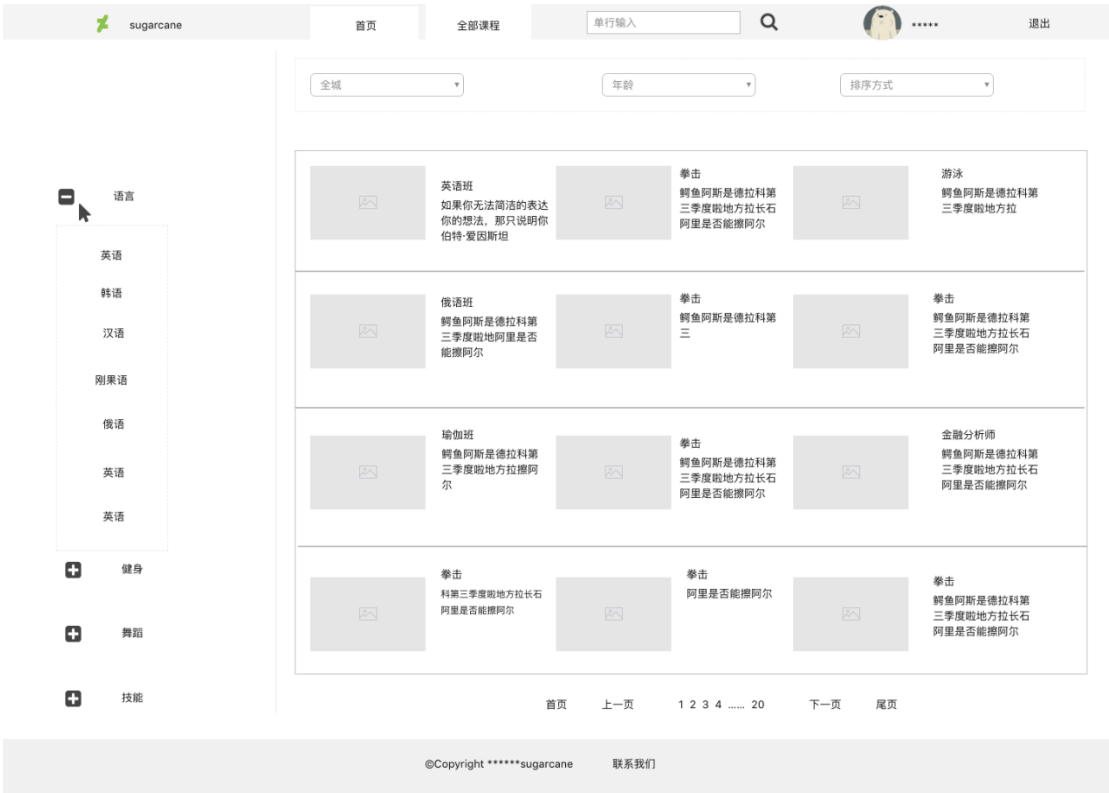


图 6.1.2 课程列表页

6.1.3 登录注册页面

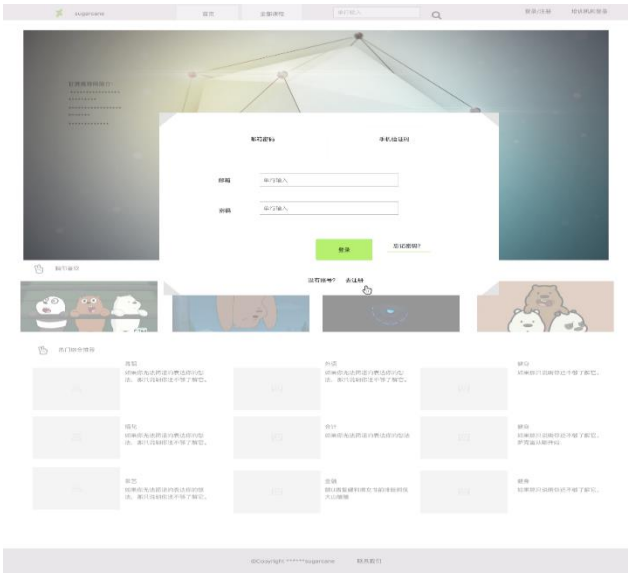


图 6.1.3 登录注册页面

6.1.4 完善个人信息页面

sugarcane

欢迎 \*\*\*\*\*

完善个人信息

性别

☐ 男 ☐ 女

年龄

学历

职业

地址

我是一条可爱的分隔线

兴趣爱好

☒

☒

☒

☒

☒

☒

☒

☒

☒

保存

图 6.1.4 完善个人信息页面

6.1.5 课程详情页面



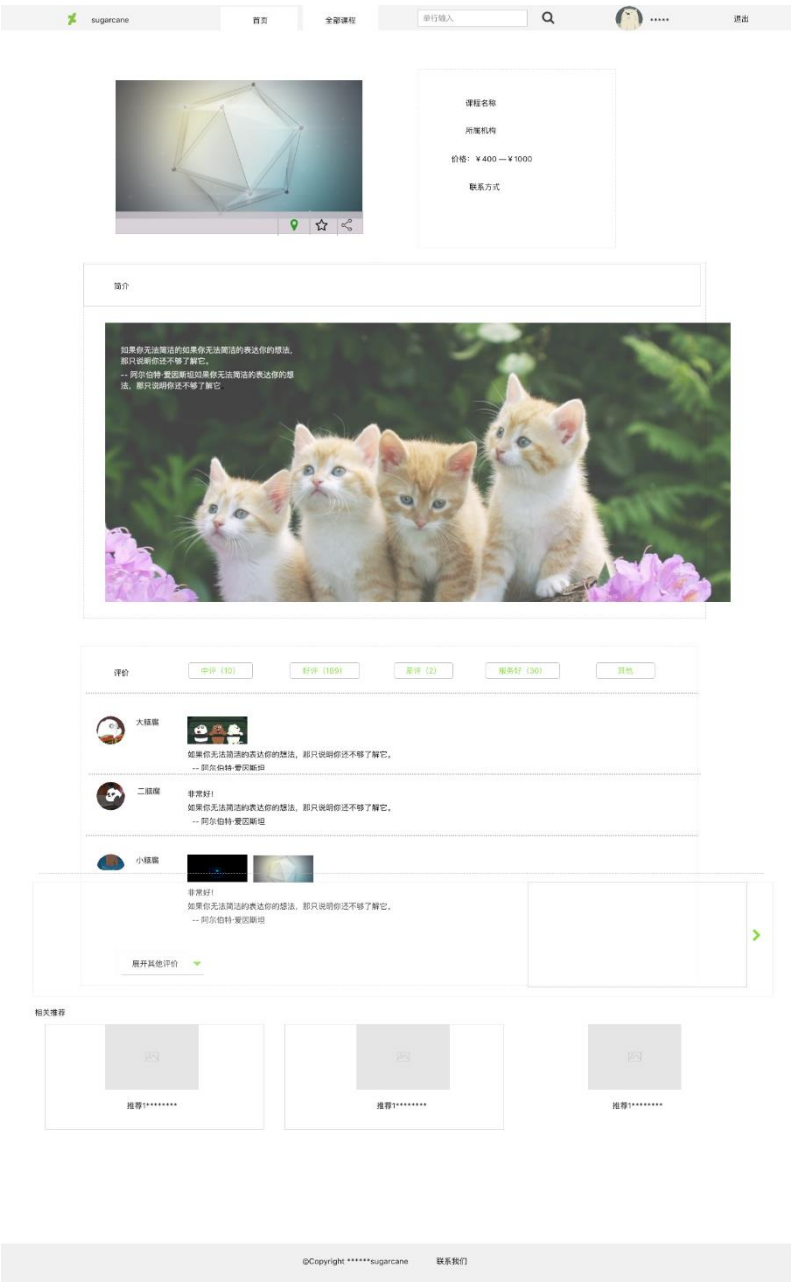


图 6.1.5 课程详情页面

6.1.6 培训机构详情页面



图 6.1.6 4.1.6 培训机构详情页面

6.1.7 个人主页

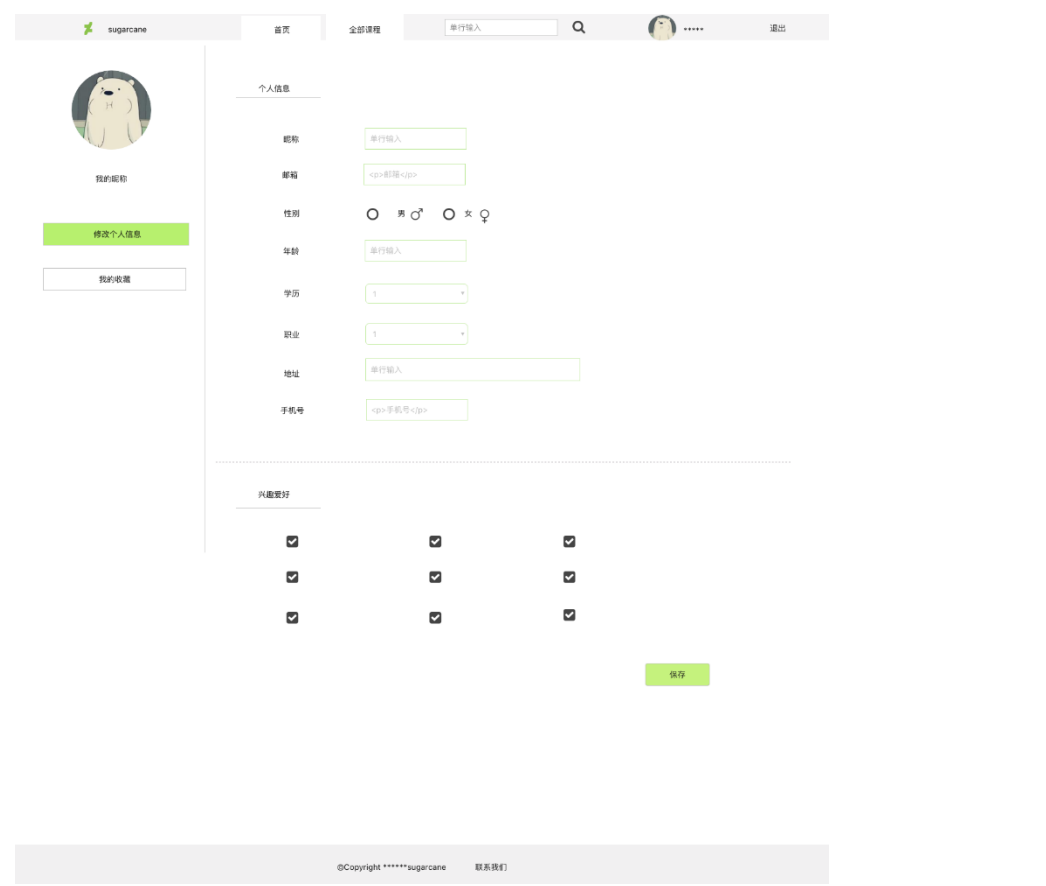


图 6.1.7 个人主页

6.1.8 我的收藏页面

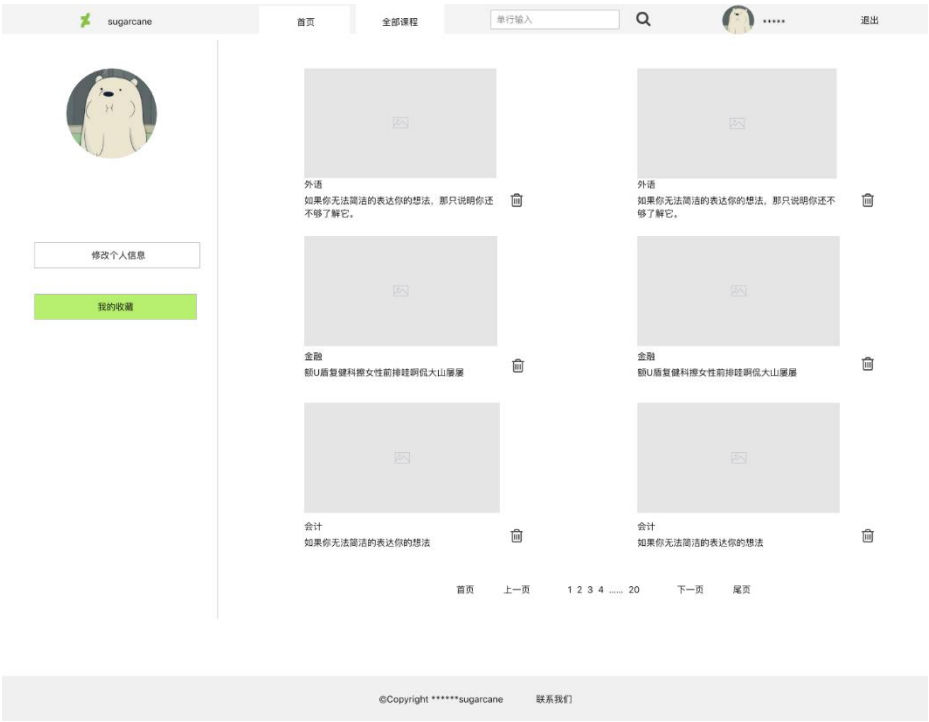


图 6.1.8 我的收藏页面

6.1.9 培训机构登录页面

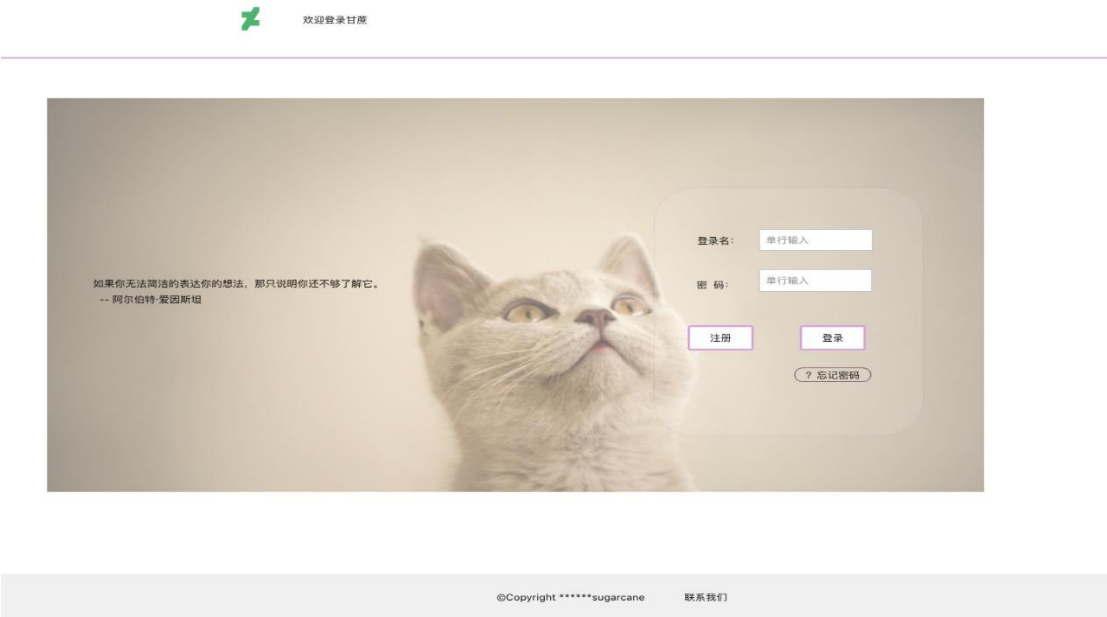



图 6.1.9 培训机构登录页面

6.1.10 培训机构注册页面



欢迎注册甘蔗

---

机构名称:

单行输入

邮箱账号:

单行输入

密 码:

单行输入

确认密码:

单行输入

联系方式:

单行输入

地 址:

单行输入

员工数量:

单行输入

上传教育局审批照片

+

上传房屋使用证明

+

上传消防安全证书

+

上传营业执照

+

提交信息

©Copyright \*\*\*\*\*sugarcane

联系我们

图 6.1.10 培训机构注册页面

6.1.11 培训机构管理简介页面

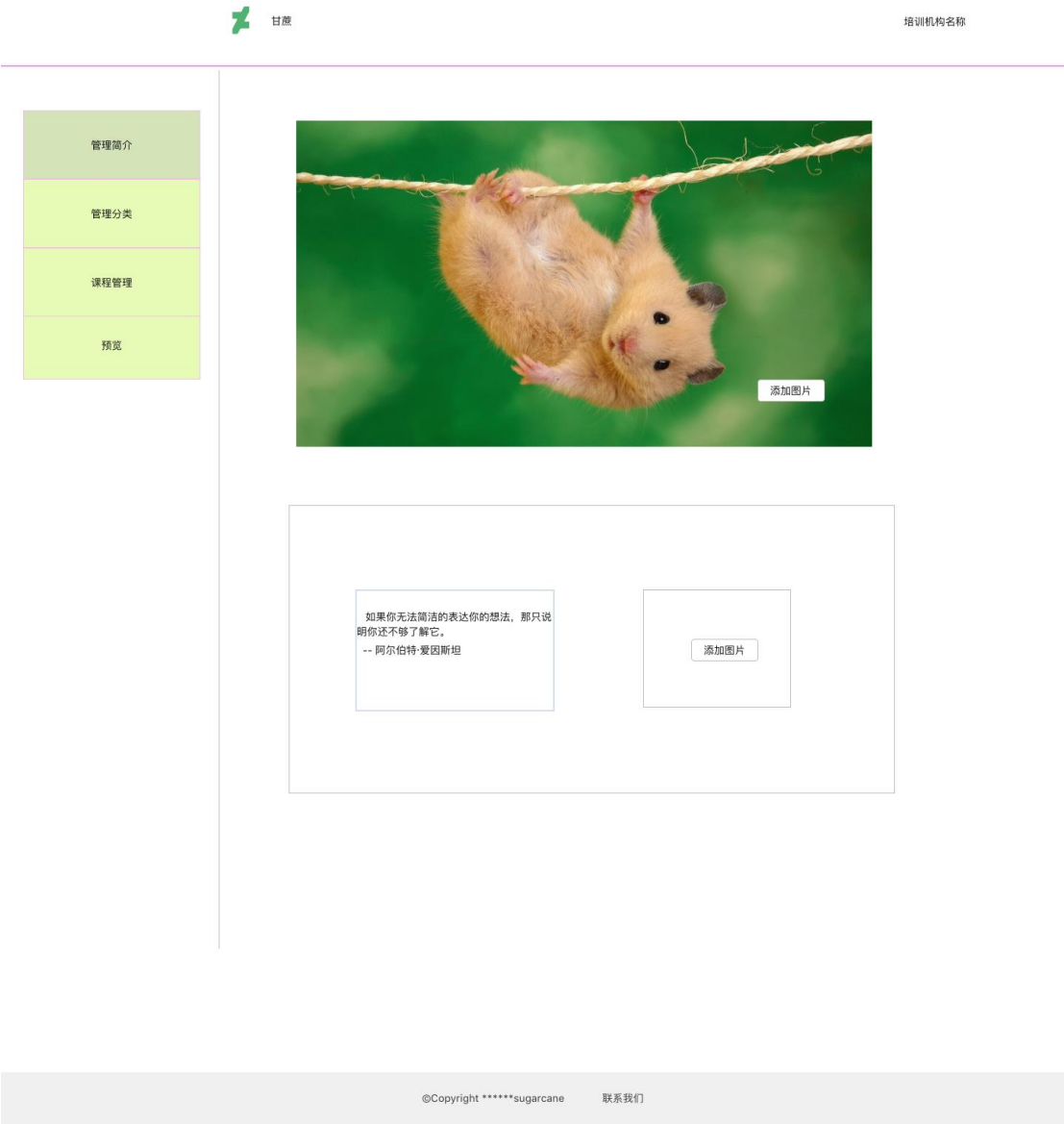


图 6.1.11 培训机构管理简介页面

6.1.12 培训机构管理分类页面

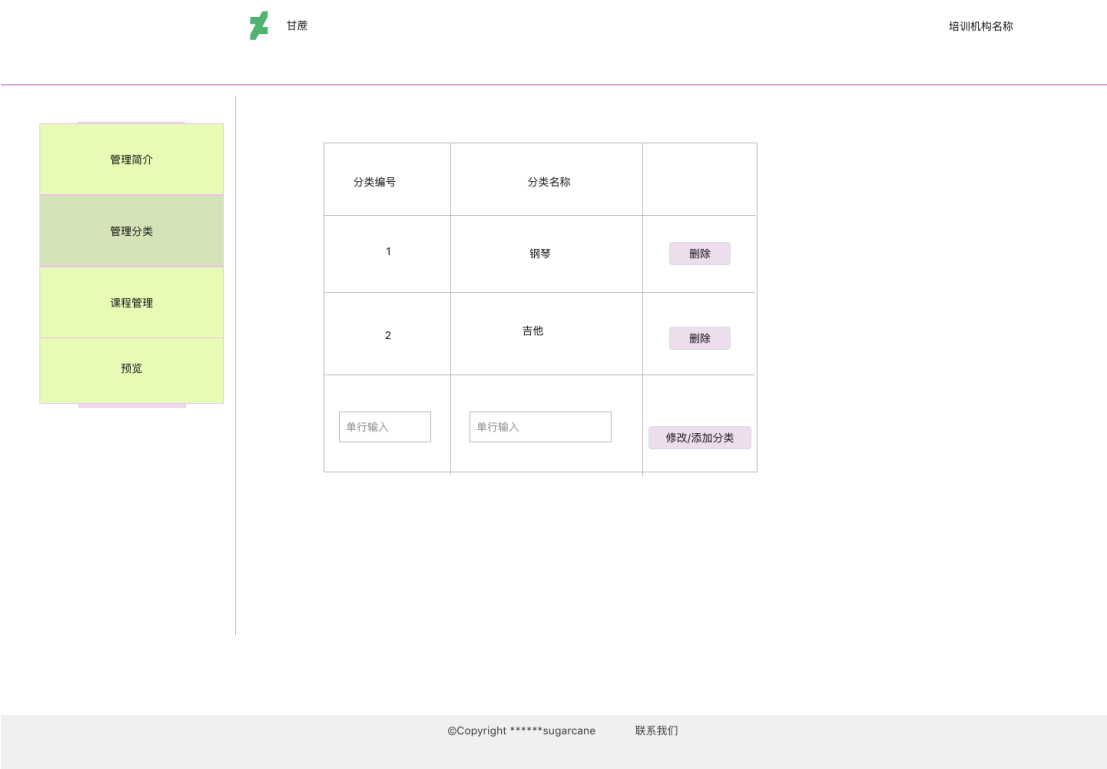


图 6.1.12 培训机构管理分类页面

6.1.13 培训机构管理课程页面

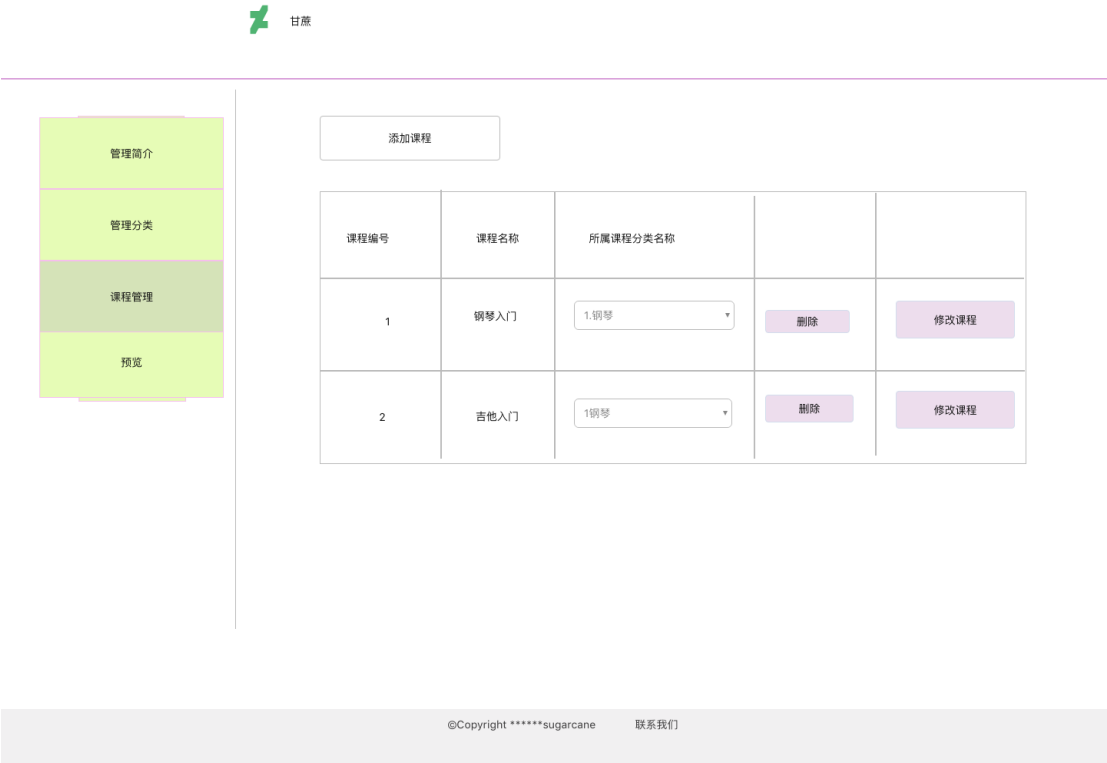


图 6.1.13 培训机构管理课程页面

6.1.14 培训机构添加课程页面



甘蔗

管理简介

管理分类

课程管理

预览

课程名称:

单行输入

分 类:

单行输入

价 格:

单行输入

授课教师:

单行输入

联系方式:

单行输入

地 址:

单行输入

简 介:

多行输入

添加图片

保存

©Copyright \*\*\*\*\*sugarcane

联系我们

图 6.1.14 培训机构添加课程页面

6.1.15 管理员登录页面

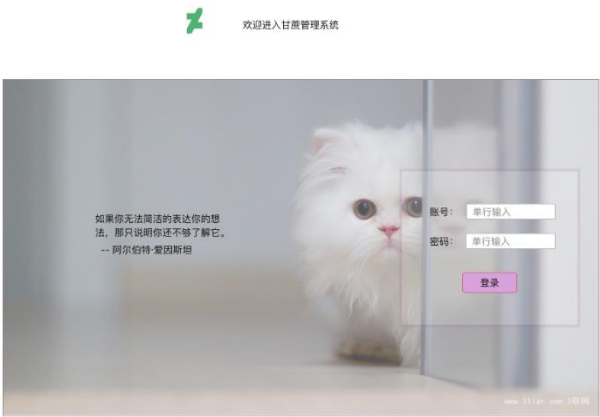


图 6.1.15 管理员登录页面

6.1.16 管理员管理用户页面

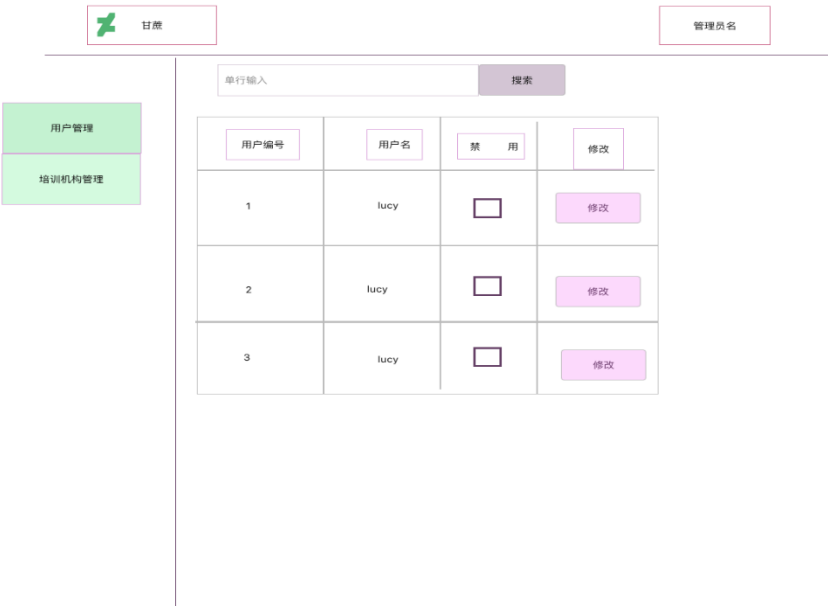


图 6.1.16 管理员管理用户页面

6.1.17 管理员管理培训机构页面

图 6.1.17 管理员管理培训机构页面

## 7 系统架构与模块实现

### 7.1 项目架构

本项目基于 Spring 框架实现，是一个开源的轻量级的 Java 开发框架，具有控制反转（IoC）和面向切面（AOP）两大核心。Java Spring 框架通过声明式方式灵活地进行事务的管理，提高开发效率和质量。

Spring 框架不仅限于服务器端的开发。从简单性、可测试性和松耦合的角度而言，任何 Java 应用都可以从 Spring 中受益。Spring 框架还是一个超级粘合平台，除了自己提供功能外，还提供粘合其他技术和框架的能力。

项目中所引用 Spring 模块如下所示。

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:context="http://www.springframework.org/schema/context"
  xmlns:jdbc="http://www.springframework.org/schema/jdbc"
  xmlns:jee="http://www.springframework.org/schema/jee"
  xmlns:task="http://www.springframework.org/schema/task" xsi:schemaLocation="
    http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-4.0.xsd
    http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-4.0.xsd
    http://www.springframework.org/schema/jdbc http://www.springframework.org/schema/jdbc/spring-jdbc-4.0.xsd
    http://www.springframework.org/schema/jee http://www.springframework.org/schema/jee/spring-jee-4.0.xsd
    http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx-4.0.xsd
    http://www.springframework.org/schema/task http://www.springframework.org/schema/task/spring-task-4.0.xsd">
  <description>Spring 公共配置 </description>
  <!-- 配置 Spring 上下文的注解 -->
  <context:annotation-config />
  <!-- 使用 annotation 自动注册 bean, 并保证@Required、@Autowired 的属性被注入 -->
  <context:component-scan base-package="com.one.sugarcane">
    <context:exclude-filter type="annotation"
      expression="org.springframework.stereotype.Controller" />
  </context:component-scan>

  <!-- 属性文件位置 -->
  <context:property-placeholder location="classpath:*.properties" />
  <bean id="dataSource" class="org.springframework.jdbc.datasource.DriverManagerDataSource">
    <property name="driverClassName" value="${jdbc.driver}" />
    <property name="url" value="${jdbc.url}" />
    <property name="username" value="${jdbc.username}" />
    <property name="password" value="${jdbc.password}" />
  </bean>
```

```

<bean id="sessionFactory" class="org.springframework.orm.hibernate5.LocalSessionFactoryBean">
  <property name="dataSource" ref="dataSource" />
  <property name="packagesToScan" value="com.one.sugarcane.entity" />
  <property name="hibernateProperties">
    <props>
      <!-- bonecp 设置 -->
      <prop
key="hibernate.service.jdbc.connections.spi.provider_class">com.jolbox.bonecp.provider.BoneCPCConnectionProvi
der</prop>
      <!-- 设置 connection 的空闲存活时间 -->
      <prop key="bonecp.idleMaxAgeInMinutes">240</prop>
      <!-- 设置测试 connection 的间隔时间 -->
      <prop key="bonecp.idleConnectionTestPeriodInMinutes">60</prop>
      <!-- 设置分区个数 -->
      <prop key="bonecp.partitionCount">3</prop>
      <!-- 设置分区中的 connection 增长数量 -->
      <prop key="bonecp.acquireIncrement">10</prop>
      <!-- 设置每个分区含有 connection 的最大个数 -->
      <prop key="bonecp.maxConnectionsPerPartition">60</prop>
      <!-- 设置每个分区含有 connection 的最小个数 -->
      <prop key="bonecp.minConnectionsPerPartition">20</prop>
      <!-- 设置 statement 缓存个数 -->
      <prop key="bonecp.statementsCacheSize">50</prop>
      <!-- 设置 connection 助手线程个数 -->
      <prop key="bonecp.releaseHelperThreads">3</prop>
      <prop key="hibernate.dialect">${hibernate.dialect}</prop>
      <prop key="hibernate.show_sql">${hibernate.show_sql}</prop>
      <prop key="hibernate.format_sql">${hibernate.format_sql}</prop>
      <prop key="hibernate.useUnicode">${hibernate.useUnicode}</prop>
      <prop key="hibernate.characterEncoding">${hibernate.characterEncoding}</prop>
      <prop key="current_session_context_class">thread</prop>
      <prop key="hibernate.connection.autocommit">true</prop>
      <prop key="hibernate.id.new_generator_mappings">false</prop>
    </props>
  </property>
</bean>
</beans>

```

项目中数据库与邮件配置信息如下所示。

```

#mysql
jdbc.driver=com.mysql.jdbc.Driver
jdbc.url=jdbc:mysql://127.0.0.1:3306/sugarcane?useUnicode=true&characterEncoding=UTF-8
jdbc.username=root
jdbc.password=

#hibernate
hibernate.dialect=org.hibernate.dialect.MySQLDialect
hibernate.show_sql=true
hibernate.format_sql=false
hibernate.useUnicode=true
hibernate.characterEncoding=utf-8

#mail 配置服务器邮箱账号信息
#服务器

```

```
mail.smtp.host=smtp.qq.com
#是否需要验证密码
mail.smtp.auth=true
#超时时间
mail.smtp.timeout=25000
#发件人信箱
mail.smtp.from=u25th_engineer@163.com
#用户名
mail.smtp.username= u25th_engineer@163.com
#密码
mail.smtp.password=xxx
```

管理员登录实体的示例代码如下所示。

```
package com.one.sugarcane.entity;

import java.util.HashSet;
import java.util.Set;

import javax.persistence.CascadeType;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.Id;
import javax.persistence.OneToOne;
import javax.persistence.Table;

import org.hibernate.annotations.GenericGenerator;

@Entity
@Table(name="ADMINLOGIN")
public class AdminLogin {
    private Integer adminID;
    private String adminName;
    private String password;
    private Set<Part>set = new HashSet<Part>();
    @Id
    @GeneratedValue(generator="a")
    @GenericGenerator(name="a",strategy="identity")
    public Integer getAdminID() {
        return adminID;
    }
    public void setAdminID(Integer adminID) {
        this.adminID = adminID;
    }
    public String getAdminName() {
        return adminName;
    }
    public void setAdminName(String adminName) {
        this.adminName = adminName;
    }
    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
}
```

```
@OneToMany(mappedBy="adminLogin",targetEntity=Part.class,cascade=CascadeType.MERGE)
public Set<Part> getSet() {
    return set;
}
public void setSet(Set<Part> set) {
    this.set = set;
}
public AdminLogin() {}
public AdminLogin(String adminName, String password, Set<Part> set) {
    super();
    this.adminName = adminName;
    this.password = password;
    this.set = set;
}
}
```

## 7.2 有关模块

项目总体架构如图 5.2.1 所示。

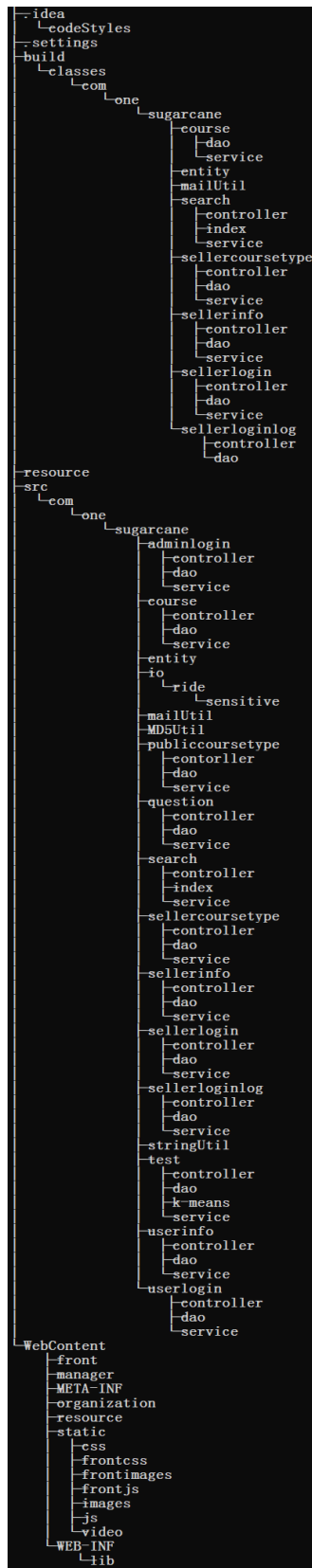


图 7.2.1 项目架构



### 7.3 开发环境

硬件：Dell G3579 笔记本 电脑

软件：IntelliJ IDEA 2020.1 (Ultimate Edition)、Navicat、Wampserver 64 bit、MySQL

主要编程语言：Java、ASP.NET

### 7.4 测试环境

前端：“甘蔗课程推荐网站”的操作将通过如下 WEB 浏览器完成：Google Chrome 最新版本，IE 最新版本，编译器自带浏览器

后台：“甘蔗课程推荐网站”的服务器为 Tomcat 服务器，该服务器的操作系统为 windows 和 Apache HTTP Server

## 8 项目成就

### 8.1 项目性能

- ①使用高峰期系统能适应 400 个用户，平均每个会话估计持续 8 分钟。
- ②系统生成的所有 Web 页面，通过速率为 40KBps 的调制解调器在不超过 10 秒的时间内可以全部下载下来。
- ③用户提交查询之后，对查询的响应时间不能超过 7 秒，在此时间内要将查询结果显示在屏幕上
- ④用户提交信息后，系统将在 10 秒内向用户显示确认消息

### 8.2 项目安全性

- ①所有个人身份信息加密
- ②除浏览主页和查看详情外，用户必须登录到网站系统才能完成其他操作
- ③顾客的登录受计算机系统访问控制策略的限制

### 8.3 个人总结

通过本次的上机实训，使我对网站的开发进一步熟练，对 MVC 开发模式有了更深刻的了解。对页面的设计更加的熟悉，能够使用 CSS 对页面进行修饰，使用各种不同的 js 使页面生动形象功能更加完善，将 Servlet, JSP 等技术和 Tomcat, My Eclipse, MySQL 有机结合起来。实现到了网页和传统网页的结合，了解到了怎么导入外包实现文件的上传功能，以及对目标文件的属性提取，在查看信息时进行分页处理，学习到了使用 ajax 来实现数据的提交和返回数据的接收，熟悉了 JavaScript 在网站开发中的一些重要的操作方法，以及使用 frame 界面实现子页面和父页面的相互传值控制功能，和 session 在实验中起到的全局变量的作用储存现登录的用户信息。同时，熟悉了使用 jquery 和 bootstrap 来完善界面功能。

本次实训使我们对软件开发的工程思想和实际问题的解决有了更深的理解。实训使我们实现的一个完全的网站的开发加深了我们对 javaweb 项目开发的了解，增加了我们的

经验，老师对我们的指导和评价更使我们明确了网站建设的评价标准和用户需求的关键所在，对网站的安全性也有了一定的了解。这些对于我们来说可能是一个非常有价值的经验。

最后，通过本次实训使我学到很多东西，这样便于今后让我提早进入工作状态。

#### **8.4 个人展望**

希望老师可以給个良。

## 附录

### 测试用例

#### 普通用户

用例编号	模块名	子模块	标题	前提	步骤	预期结果	相关用例
------	-----	-----	----	----	----	------	------

#### 培训机构

用例编号	模块名	子模块	标题	前提	步骤	预期结果	相关用例
1	登录		培训机构登录成功	已注册，当前页面为培训机构登录注册页	1.输入正确的用户名和密码 2.点击登录	登录成功	

#### 管理员

用例编号	模块名	子模块	标题	前提	步骤	预期结果	相关用例
------	-----	-----	----	----	----	------	------