计算机与信息工程学院

**专**

**业**

**实**

**训**

**报**

**告**

|  |  |
| --- | --- |
| 课程名称 | 专业实训 |
| 项目名称 | 在线音乐管理系统 |
| 班 级 | 2015级软件工程 |
| 学 号 |  |
| 姓 名 |  |
| 指导教师 |  |

2018年10月25日

|  |  |
| --- | --- |
| **课程设计主要目的** | 通过综合项目训练，让学生运用以前课程中学习过的前端框架Bootstrap和本次专业实训所讲授的SpringBoot设计并完成一个简单音乐管理网站。其目的在于检验学生对相关课程知识点的掌握情况，并培养学生实际分析问题、解决问题、编程和动手操作等多方面的能力。同时编写课程设计报告，锻炼软件开发文档撰写的基本方法。 |
| **课程设计任务要求** | 开发及运行环境要求：使用Eclipse Nero3或以上版本，数据库系统使用Mysql5.5，JDK1.8。页面使用HTML+Thymeleaf；控制层SpringMVC；持久层：SpringData JPA；安全控制：自己实现或使用框架SpringSecurity。要求界面美观大方，布局合理；代码编写规范；程序运行流畅，没有错误；对用户不当的操作，要有友好的提示；撰写规范的实训项目设计报告。 |
| **教师评语** | **成绩： 指导教师（签名）** |

## 设计任务说明

本系统角色分为：管理员 高级会员 普通会员 游客。

管理员：管理音乐（增删改查）：管理音乐类别（增删改）。删除该类别时也要删除该类别包含的音乐；管理员会员（删改查），按会员账号进行查询，可以修改会员的角色。

高级会员：浏览音乐（包括分类查询）；听音乐；收藏音乐；无限量下载音乐；

普通会员：浏览音乐（包括分类查询）；听音乐；收藏音乐；只可以下载10首音乐；

游客：浏览音乐（包括分类查询）；听音乐。

系统要求：

使用Spring Boot框架完成该项目。

表现层：HTML+Themeleaf；

控制层：Spring MVC；

持久层：Spring Data JPA；

安全控制：Spring Security（可选）；

数据库：MySql；

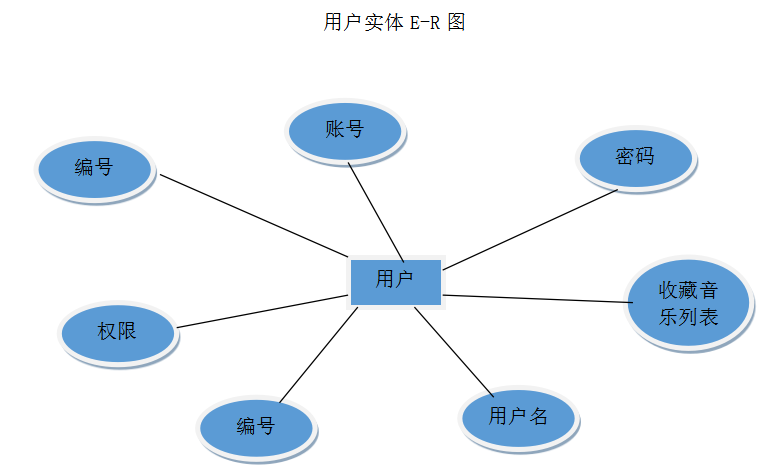
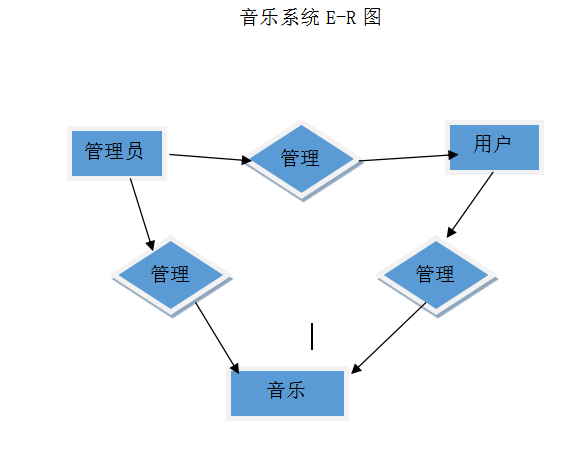
## 需求分析

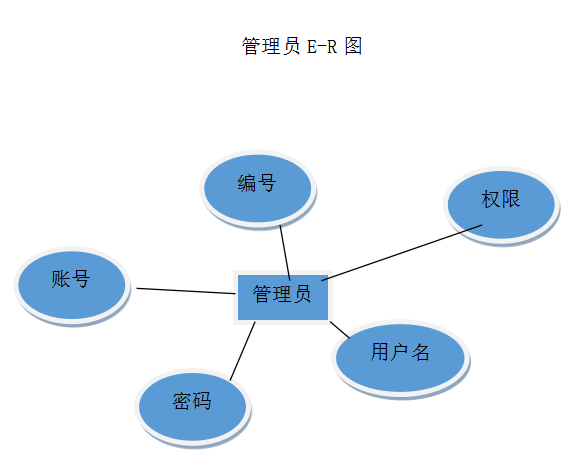
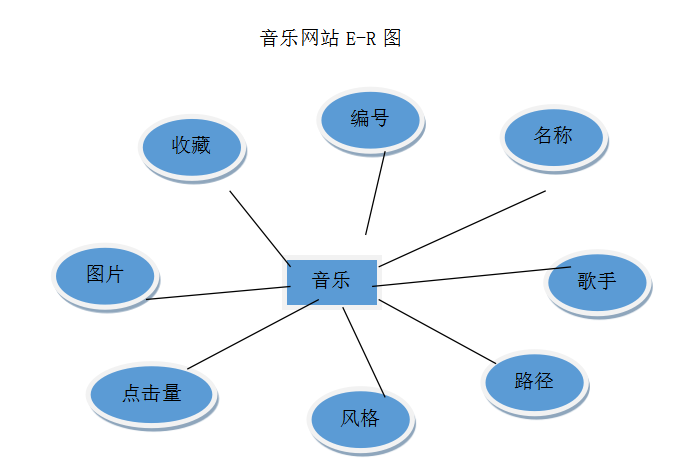
1. 功能需求

表1 功能需求表

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 功能名称 | 所需最低权限 | 功能描述 |
| 01 | 用户登录注册 | 无 | 实现用户的登录注册 |
| 02 | 音乐复杂搜索 | 游客 | 根据用户输入的信息和搜索类型进行相关的搜索 |
| 03 | 在线播放 | 游客 | 点击音乐可以在线播放 |
| 04 | 意见反馈 | 游客 | 对管理员提出对本系统的意见 |
| 05 | 添加音乐收藏 | 普通会员 | 用户可以往自己的音乐收藏界面添加自己喜欢的音乐 |
| 06 | 下载音乐 | 普通会员 | 用户可以往自己本地下载自己喜欢的音乐，下载次数根据用户权限等级进行限制 |
| 07 | 查看音乐收藏 | 普通会员 | 用户可以单独根据自己的喜欢的音乐进行管理，取消收藏，播放和下载 |
| 08 | 添加信息 | 管理员 | 管理员可以添加音乐，音乐分类 |
| 09 | 删除信息 | 管理员 | 管理员可以删除音乐，音乐分类和用户的信息 |
| 10 | 更改信息 | 管理员 | 管理员可以修改音乐，音乐分类和用户的信息 |
| 11 | 用户账号查询 | 管理员 | 管理员可以根据用户账号详细查询和管理 |
| 12 | 音乐信息导出Excel | 管理员 | 管理员将所有信息导出Excel文件 |

1. 数据分析





## 设计方案

总体设计阶段的基本目标就是概要地回答系统应该如何实现。在需求分析的基础之上规划出了系统的功能模块，以及在此分析基础上设计出的具体的、可以满足本系统全部需求的数据库。设计在任何工程产品或系统中，是开发阶段的第一步。可以定义为应用各种技术和原理，对一个设备、一个过程或一个系统，做出足够详细的决策，使之有可能在物理上得以实现的过程。

## 3.1 系统模块总体设计

在线音乐点播平台是一个典型的音乐网站，网站分为前台和后台2个大的模块，每个模块都有自己的权限功能。网站前台显示了音乐的详细信息，具体搜索指令内容和收藏工功能。网站提供了用户注册功能，注册了的用户可以登录网站，实现游客没有的功能，如对音乐的收藏功能和下载功能。对于用户还详细区分了两种角色vip和svip，前者仅拥有音乐下载10次的数量，而svip不限制下载次数。后台管理员对前台涉及到的所有信息进行管理。开发这样的一个网站实现了一般音乐网站的所有功能，本站的特色就在于前台分为4种角色可以具备不同的权限功能 。

## 3.1.1 音乐平台系统的结构

根据网站的角色对网站角色分为4种类型，一种是未注册用户，一种是网站vip，一种是网站svip，以及网站的后台管理员。

未注册的用户功能权限有：选择的音乐进行在线播放、歌手名搜索、点击量范围查询、音乐分类查询也可以进行组合搜索。

注册的用户除了未注册的功能之外还具有以下两种权限：

歌曲收藏：用户登录网站之后可以对自己喜欢的歌曲进行收藏，方便下次登录时查找。

歌曲下载：用户登录网站之后可以下载自己喜欢的歌曲在用户本地电脑上。

管理员权限：添加删除修改歌曲和音乐分类信息、用户信息管理、对网站前台注册了的用户进行查看和删除等操作。

退出账号：用户在后台可以退出，这样就清除缓存中的账号和密码，如果需要再进入后台需重新登录。

## 3.1.2 音乐平台系统的流程

打开网站首先展示的是本程序的首页，首页中包含了用户注册，登录和不需要帐号密码的游客登陆。接着就是音乐的歌手、排行榜、分类、专辑、我的音乐等选项，当用户点击歌曲时将会对音乐进行在线播放，也能对当前歌曲的一些播放操作、播放列表、用户收藏模块以及歌曲下载模块等。

## 3.2 数据库设计

数据库设计阶段将具体地设计目标系统，得出新系统软件的详细信息。同时，要求设计出的信息简明易懂，便于下一阶段用某种程序设计语言在计算机上实现。数据库设计的注意事项：

1.定义标准的对象命名规范；

2.方便的数据查询，支持多条件查询；

3.了解用户需求，根据业务需要创建数据表；

4.创建数据字典和 E－R 图表；

5.数据库维护。

## 3.2.1 数据字典

数据字典则是系统中各类数据描述的集合，是进行详细的数据收集和数据分析所获得的主要成果。数据字典通常包括数据项、数据结构、数据流、数据存储和处理过程五个部分。数据库数据字典不仅是每个数据库的中心，而且对每个用户也是非常重要的信息，用户可以用SQL语句访问数据库数据字典。

数据字典的作用是在软件分析和设计的过程中给人提供关于数据的描述信息。数据流程图配以数据字典，就可以从图形和文字两个方面对系统的逻辑模型进行完整的描述。它主要是对数据流图中的数据流、处理逻辑、外部实体、数据存储和数据项等方面进行具体的定义，我们不能手工修改数据字典里的信息。

## 3.2.1.1 数据元素的定义

数据元素（data element）是数据的基本单位，在计算机程序中通常作为一个整体进行考虑和处理。包括如下内容：数据项描述＝｛数据项名，数据项含义说明，别名，数据类型，长度，取值范围，取值含义，与其他数据项的逻辑关系，数据项之间的联系｝。其中，后两项定义了数据的完整性约束条件，用于数据校验。具体内容如表2所示。

表2 音乐网站数据元素定义表

|  |  |  |  |
| --- | --- | --- | --- |
| 数据项编号 | 数据项名 | 数据项含义 | 数据型类型 |
| DI-01 | music\_id | 音乐编号 | integer |
| DI-02 | music\_name | 音乐名称 | nvarchar |
| DI-03 | music\_singer | 音乐歌手 | nvarchar |
| DI-04 | music\_src | 音乐路径 | nvarchar |
| DI-05 | music\_clicks | 音乐点击量 | integer |
| DI-06 | music\_image | 音乐图片 | nvarchar |
| DI-07 | classs\_id | 类别编号 | integer |
| DI-08 | classs\_name | 类别名称 | nvarchar |
| DI-09 | username | 用户名称 | nvarchar |
| DI-10 | role\_id | 权限编号 | integer |
| DI-11 | role\_name | 权限名称 | nvarchar |

## 3.2.1.2 数据结构

数据结构指的是数据之间的相互关系，即数据的组织形式。一般包括数据的逻辑结构存、储结构和数据的运算三个方面的内容。具体内容如表3所示。

表3音乐网站基本数据结构

|  |  |  |  |
| --- | --- | --- | --- |
| 数据结构编号 | 数据结构名称 | 含义说明 | 组成 |
| DS-1 | 管理员 | 描述管理员信息 | （id,username,loginname,password,role) |
| DS-2 | 用户 | 描述用户信息 | （id,username,loginname,password,role) |
| DS-3 | 音乐 | 描述音乐信息 | （id,number,name,class,singer,clisks,crc,image） |
| DS-4 | 音乐类别 | 描述音乐类别信息 | （id,number,name） |

## 3.2.2 数据表的设计

本系统数据库采用Navicat for MySQL数据库，系统数据库名称为“joa\_test”。数据库“joa\_test”中包含7张表。在Navicat for MySQL数据库设计中，表是Navicat for MySQL 数据库中最基本的对象之一，这个基础对象对于数据库来说，非常重要，因为其设计是否合理，直接跟数据库的性能相关。所以，要重视表的设计，从基础做起，为提升数据库的性能而努力。

数据库“joa\_test”由下面多个表格组成，各个表格的设计结果如下表格所示，每个表格表示在数据库中相对应的一个表。

表4 音乐分类表——tb\_class

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 意义 | 允许为空 | 备注 |
| code | int | ID | 否 | 主键 |
| classname | varchar | 名称 | 否 | 外键 |

表5 音乐表——tb\_music

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 意义 | 允许为空 | 备注 |
| id | int | ID | 否 | 主键 |
| class | varchar | 类别 | 否 | 外键 |
| name | varchar | 名称 | 是 |  |
| image | varchar | 图片 | 是 |  |
| number | int | 编号 | 否 |  |
| singer | varchar | 歌手 | 是 |  |
| src | varchar | 路径 | 是 |  |
| cloect | int | 收藏 | 是 |  |

表6 权限表——tb\_role

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 意义 | 允许为空 | 备注 |
| id | int | ID | 否 | 主键 |
| authority | varchar | 权限 | 否 |  |

表7 用户建议表——tb\_sug

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 意义 | 允许为空 | 备注 |
| id | int | ID | 否 | 主键 |
| suggest | varchar | 建议 | 是 |  |
| name | varchar | 名称 | 是 |  |
| tel | varchar | 电话 | 是 |  |

表8 用户表——tb\_us

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 意义 | 允许为空 | 备注 |
| id | int | Id | 否 | 主键 |
| username | varchar | 用户名 | 是 |  |
| loginname | varchar | 登录账号 | 否 |  |
| password | varchar | 密码 | 否 |  |
| downloadnums | varchar | 下载次数 | 否 |  |

表9 收藏音乐表——tb\_user\_music

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 意义 | 允许为空 | 备注 |
| user\_id | int | 用户ID | 否 | 外键 |
| music\_id | int | 音乐ID | 否 | 外键 |

表10 用户权限表——tb\_user\_role

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 意义 | 允许为空 | 备注 |
| user\_id | int | 用户ID | 否 | 外键 |
| role\_id | int | 权限ID | 否 | 外键 |

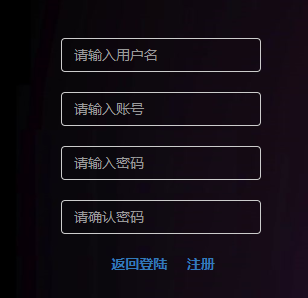
## 主要模块的实现

## 4.1 用户注册登陆模块

表11 用户管理模块功能表

|  |  |
| --- | --- |
| **功能名称** | **定 义** |
| 用户注册 | 实现用户注册功能 |
| 用户登录 | 实现用户登录功能 |
| 游客登陆 | 实现无账号登陆 |

图1 用户注册模块界面



用户注册是一个多本文给予用户注册时填写的信息，当用户点击确定后，前台处理获取文本框的内容并检查内容符合后传入后台再写入数据库，具体实现代码如下：

|  |
| --- |
| **if(usRepository.findByLoginName(user.getLoginName())!=null){**  **return "redirect:/regist?error";**  **}else{**  **List<FKRole> roles=new ArrayList<FKRole>();**  **FKRole role=roleRepository.findByAuthority("ROLE\_VIP");**  **roles.add(role);**  **user.setRoles(roles);**  **user.setDownloadnums(0);**  **BCryptPasswordEncoder b=new BCryptPasswordEncoder();**  **user.setPassword(b.encode(user.getPassword()));**  **userService.save(user);**  **return "login";** |

## 4.2 音乐播放模块

播放模块主要包含了歌曲列表、播放列表、暂停、播放以及为已登录的用户添加的下载与收藏功能等

表12 我的音乐在线播放功能

|  |  |
| --- | --- |
| **功能名称** | **定 义** |
| 音乐列表 | 显示当前页面下可供选择的音乐列表 |
| 播放功能 | 点击即刻在线播放 |
| 下载功能 | 将网络端音乐下载到本地 |
| 收藏功能 | 将音乐收藏到自己的收藏目录之下 |

图2 音乐首界面



后台代码

|  |
| --- |
| **String user=getUsername();**  **String role=getAuthority();**  **List<Music> musicnum=musicService.findAll();**  **Page<Music> musics= musicService.getMusicByPage(music,matchmusic,pageIndex , pageSize); webRequest.setAttribute("downloadMusic",downloadMusic,RequestAttributes.SCOPE\_SESSION)**  **webRequest.setAttribute("role", role,RequestAttributes.SCOPE\_SESSION);**  **List<Classs> musicclass = classService.findAll();**  **webRequest.setAttribute("musicclass", musicclass, RequestAttributes.SCOPE\_SESSION);**  **if (musicnum.size() >= 100) {**  **webRequest.setAttribute("disabled\_add", true, RequestAttributes.SCOPE\_SESSION);**  **} else {**  **webRequest.setAttribute("disabled\_add", false, RequestAttributes.SCOPE\_SESSION);**  **}**  **model.addAttribute("success","欢迎！！欢迎！！您是《第"+peop+"位》光临本系统的贵人");**  **return "other/musicList";** |

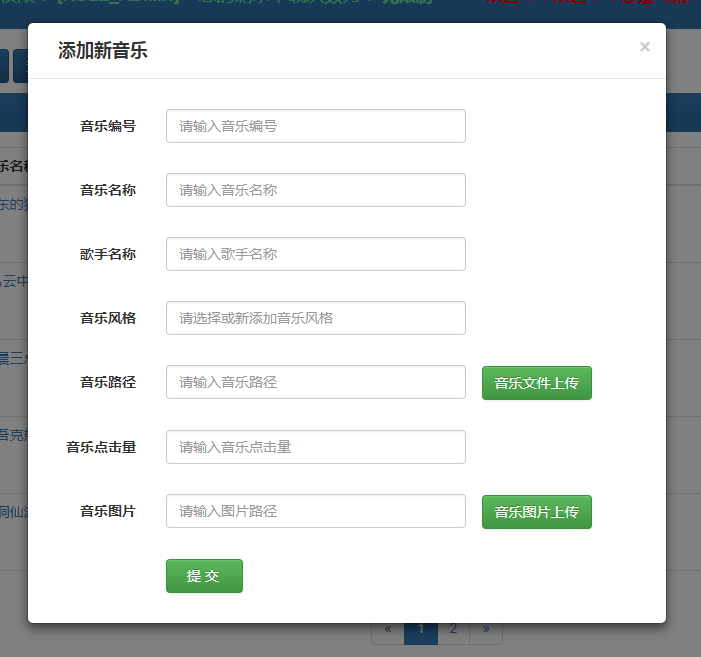
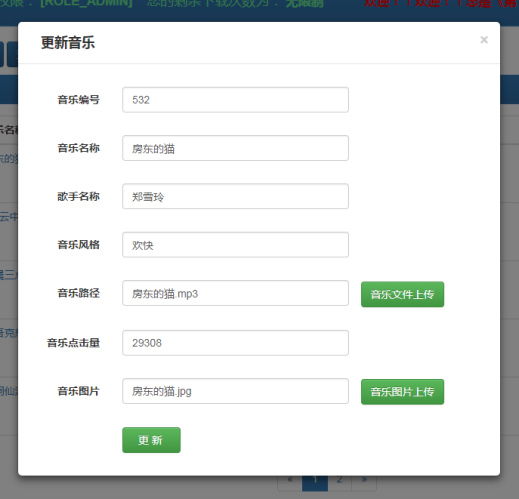
前台代码

|  |
| --- |
| **<table class="table table-bordered table-hover">**  **<tr>**  **<th class="text-center">音乐序号</th>**  **<th class="text-center">音乐编号</th>**  **<th class="text-center">音乐名称</th>**  **<th class="text-center">歌手名称</th>**  **<th class="text-center">音乐风格</th>**  **<th class="text-center">音乐路径</th>**  **<th class="text-center">音乐</th>**  **<th class="text-center">音乐点击量</th>**  **<th class="text-center">音乐图片</th>**  **<th id="operation" class="text-center">操作</th>**  **</tr>**  **<tbody class="text-center" id="musicdetiles">**  **<tr th:each="musicone,musicstat:${session.Objects}">**  **<td th:text="${musicstat.count}"></td>**  **<td th:text="${musicone.number}"></td>**  **<td>**  **<a th:href="@{|/musicDetails/${musicone.src}|}"**  **th:text="${musicone.name}"></a>**  **</td>**  **<td th:text="${musicone.singer}"></td>**  **<td th:id="${musicone.classs.code}"**  **th:text="${musicone.classs.classname}"></td>**  **<td th:text="${musicone.src}"></td>**  **<td>**  **<audio th:src="@{'/music/'+${musicone.src}}"**  **th:id="${musicstat.count}"></audio>**  **<input type="button" th:id="${musicstat.count}+.1"**  **class="btn btn-primary music" value="播放"></input>**  **&nbsp;&nbsp;**  **<input id="downloadbtn" type="button" value="下载音乐" class="btn btn-primary" />**  **&nbsp;&nbsp;**  **<b class="cloectclass" th:text="${musicone.clloect}" style="display: none"></b>**    **<input id="collectmusic" type="button" class="btn btn-primary clo"**  **th:onclick="'javascript:clo('+${musicone.number}+')'"**  **value="未知收藏"></input>**  **</td>**  **<td th:id="${musicstat.count}+.2" th:text="${musicone.clicks}"**  **th:style="'color:'+@{(${musicone.clicks}>10000?'red':'blue')}"></td>**  **<td>**  **</tr>**  **</tbody>** |

## 4.3 模态框增加或修改模块

添加信息通过用form表单信息来往后台传递数据。而修改信息先在前台通过jqery获取当前已显示信息自动填写再此基础上进行修改。

图3 添加音乐模态框 图4更新音乐模态框

前端代码

|  |
| --- |
| **<div class="form-group" style="margin: 30px 0px 0px 20px;">**  **<label for="firstname" class="col-sm-2 control-label">歌手名称</label>**  **<div class="col-sm-10">**  **<input type="text" class="form-control" name="singer" style="width: 300px" placeholder="请输入歌手名称" />**  **</div>**  **</div>**  **<div class="form-group" style="margin: 30px 0px 0px 20px;">**  **<label for="firstname" class="col-sm-2 control-label">音乐风格</label>**  **<div class="col-sm-10">**  **<input type="text" class="form-control" id="classs" name="classname" style="width: 300px" placeholder="请选择或新添加音乐风格"**  **onclick="$('#dropdown-1').trigger('click')"/>**  **&nbsp;&nbsp;**  **<input id="dropdown-1" style="display: none" data-toggle="dropdown" />**  **<ul class="dropdown-menu" role="menu" style="margin: 0px 0px 0px 14px;" aria-labelledby="dropdownMenu1">**  **<li class="col-sm-10" role="presentation" style="width: 300px">**  **<a class="menu" role="menuitem" tabindex="-1" value="1" >复古</a><a class="menu" role="menuitem" tabindex="-1" value="2" >经典</a><a class="menu" role="menuitem" tabindex="-1" value="3" >欢快</a></li>**  **</ul>**  **</div>** |

前台脚本（js）

|  |
| --- |
| **function c(id) {//修改信息自动填写**  **var childs = $("#musicdetiles").children("tr").eq(id - 1).children(**  **"td");**  **var img = childs.eq(8).children("img").attr("src");**  **$("#updateModal").modal("show");**  **img = img.substr(5);**  **$("#number").attr("value", childs.eq(1).text());**  **$("#name").attr("value", childs.eq(2).find("a").text());**  **$("#singer").attr("value", childs.eq(3).text());**  **$("#classss").attr("value", childs.eq(4).text());**  **$("#srcfield1").attr("value", childs.eq(5).text());**  **$("#clicks").attr("value", childs.eq(7).text());**  **$("#imagefield1").attr("value", img);**  **}** |

后台代码

|  |
| --- |
| **List<FKRole> roles=new ArrayList<FKRole>();**  **FKRole role=roleRepository.findByAuthority(r.getAuthority());**  **roles.add(role);**  **user.setRoles(roles);**  **if(user.getPassword().length()<=50) {**  **BCryptPasswordEncoder password=new BCryptPasswordEncoder();**  **user.setPassword(password.encode(user.getPassword()));**  **}**  **usRepository.save(user);**    **Page<FKUser> users=userService.findAll(0,5);**  **webRequest.setAttribute("Objects", users, RequestAttributes.SCOPE\_SESSION);**  **roles=roleRepository.findAll();**  **webRequest.setAttribute("roles", roles, RequestAttributes.SCOPE\_SESSION);**  **model.addAttribute("success","更新成功");**  **return "admin/UserInfo";** |

## 项目总结

系统初期要学习和理解好3S框架，必须要熟悉音乐网站的开发流程，初步了解音乐网站的相关知识。从开始熟悉这些知识到对整体设计有了解，再从概要设计、详细设计到开始编码、以及最后的测试，整个过程感觉很充实，虽然遇到了不少困难，但当我通过自己查资料、或是向指导老师请教以及向同学请教，而设计出解决方案并成功实现时，那种成就感和满足感足以鼓励自己加班加点的辛苦。

通过本网站系统的开发，了解了Spring Boot框架的优势和便捷之处以及网站的开发过程，同时对制造业的流程有了一个系统的认识，极大地拓宽了我的知识面，感到收获不小。同时也深感自己知识的欠缺。虽然即将毕业，但在以后的学习工作中，一定要继续坚持不段地学习新兴的专业知识及相关的非专业知识，只有这样才能紧跟时代的潮流。

本系统投入使用后，放到互联网上供广大用户试听音乐。让音乐网站用户得到很好的音乐服务。在此程序的设计中，仍有许多不够完善的地方，如上传的歌曲只能是管理员，这对于计算机的处理能力与人为输入数据时效严重失衡。为解决这一问题，应该在此程序的前台开放一个用户上传歌曲的功能，以积分领取的形式奖励积分给上传者，而这些积分用于下载网站上的歌曲。为再进一步完善程序，可为歌曲信息页面中添加评论以及用户之间的好友添加，方便用户之间的讨论。

本系统采用Eclipse集成开发环境为前台开发工具，结合Navicat for MySQL数据库管理系统。既考虑了系统的需求，又兼顾了运行的效率，以及今后系统的功能拓展等。具有运行效率、开发周期短、扩展能力强、技术规范等特点。