**南京航空航天大学金城学院**

**信息工程学院**

**专业实习报告**

实习项目/岗位 音乐歌曲网站

实习单位/地点 南航金城

学生姓名 汤亚东

学 号 2018027421

专 业 软件工程

联系方式 15261176327

实习日期： 2021 年 6 月 15 日至 2021年 8月 30 日

**音乐歌曲网站实训报告**

# 一.概述

实习单位：南航金城学院

实习岗位：升学发展组（java组）

实习内容：音乐歌曲网站

实习任务的完成情：通过使用ssm框架实现了整个的搭建。其中通过mybatis实现对数据库的访问操作，实现增、删、改、查功能。通过Spring初始化Bean，处理类与类之间的关系等。通过SpringMVC提供API接口，前端通过API接口与后台进行交互。使用Tomcat部署网站，使得用户可以访问网站。用户通过用户登录界面登录，登录成功后进入音乐网站页面，用户还可以使用注册功能注册账号后登录。其中音乐网站页面可以对音乐播放下载，还有上传音乐添加与查看收藏等功能。管理员可以通过控制台登录页面登录。登录后可以管理用户信息、音乐信息、音乐分类等。

# 二．项目主体

# 2.1 需求分析与总体设计

# 1.基本功能

1）音乐播放：进入音乐网站，点击歌曲进入音乐播放界面，打开音乐播放键，歌曲开始播放；

2）音乐上传：用户个人中心中，进行音乐的添加，输入音乐名称，类型以及图片等信息之后便可以上传音乐文件；

3）音乐查询：通过音乐名称，音乐分类，音乐所唱的语言进行音乐查询；

4）音乐收藏：选中收藏按键，音乐收藏成功；

5）用户注册登录：输入用户名、姓名、出生日期等信息并且提交，进行用户注册；

6）账号管理： 管理员成功登录后，可以查看账号信息、增添账号、查看已经注册的用户、添加新的用户以及对管理员账号的密码进行相应的修改等；

7）音乐分类管理：对音乐的类别进行增添和查询操作。添加分类时输入分类名称并提交，可在首页添加新的音乐分类，音乐分类查询可以查看已经拥有的音乐分类；

8）音乐管理：对音乐进行增删改查操作，添加音乐时将音乐名称，音乐类型，音乐语言等信息一并输入，即可进行音乐的增添操作。

# 2.开发环境

IDE：IDEA

Java版本：JDK8

Maven版本：Maven3.6

Web服务器：Tomcat8.5.6

数据库：MariaDB10.3.16

# 3.数据库

数据库类型：MySQL

数据库管理工具：Navicat

2.2 数据库设计

表 1 数据库设计

|  |  |
| --- | --- |
| 表名称 | 说明 |
| admin | 管理员信息表 |
| user | 用户信息表 |
| genre | 各区分类 |
| song | 歌曲信息表 |
| collect | 收藏表 |
| comment | 评论表 |

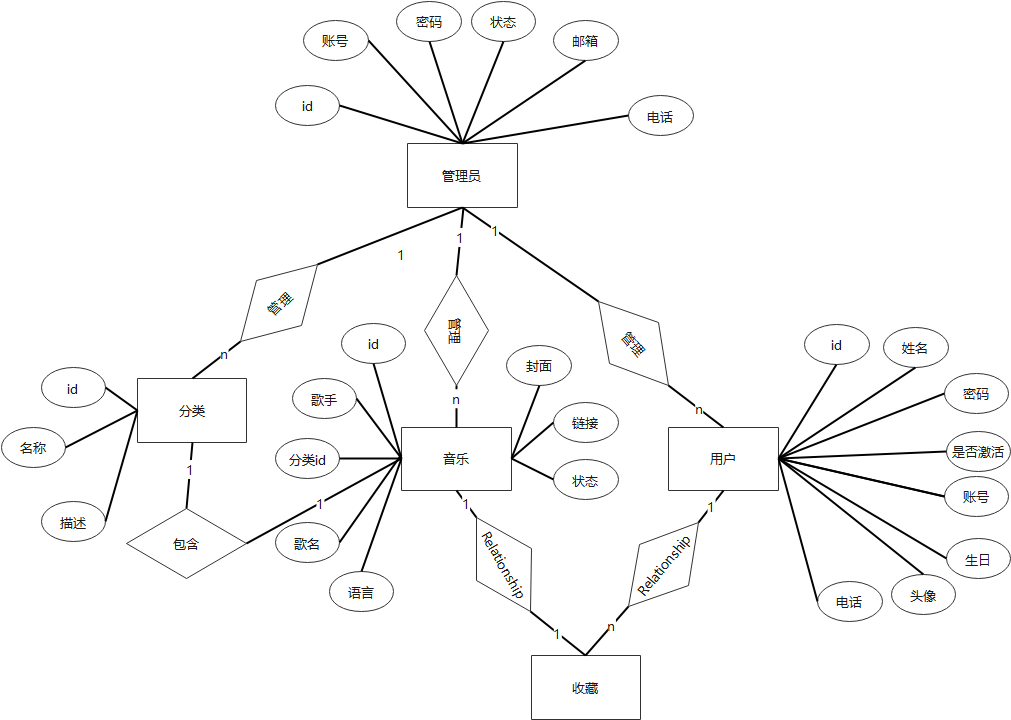


图 1 数据库ER图

1. 管理员表

表名：admin

登录：用户名+密码

表 2 管理员表

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 约束 | 说明 |
| id | int | 非空唯一 | 主键 |
| username | varchar(30) | 非空唯一 | 用户名 |
| password | varcha(50) | 非空 | 密码 |
| mobile | varchar(11) |  | 手机号 |
| email | varchar(25) |  | 邮箱 |
| enable | varchar(1) | 非空 | Y：启用 N：禁用 默认启用 |

1. 用户表

用户名：user

登录：邮箱+密码

表 3 用户表

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 约束 | 说明 |
| id | int | 非空唯一 | 主键 |
| mobile | varchar(11) | 唯一 |  |
| mail | varchar(25) | 非空唯一 | 邮箱最为用户登录账号 |
| password | varchar(50) | 非空 | 密码 |
| nickname | varchar(20) |  | 昵称（可以使用昵称登录） |
| avatar | varchar(300) | 非空 | 用户头像 |
| locked | char(1) | 非空 | Y：锁定 N：正常 默认正常 |

1. 音乐表

表名：song

表 4 音乐表

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 约束 | 说明 |
| id | bigint | 非空唯一 | 主键 |
| genre\_id | int |  | 外键分类ID |
| admin\_id | int |  | 外键上传歌曲管理员ID |
| title | varchar(255) | 非空 | 歌曲名称 |
| source | varchar(300) | 非空 | 音乐链接 |
| poster | varchar(300) | 非空 | 音乐图片 |
| author | varchar(20) |  | 音乐作者，使用“,”间隔 |
| collections | bigint | 非空 | 音乐收藏量 |
| status | int | 非空 | 发布状态0:未发布1:已发布 2:已下线 |
| publish\_time | timestamp |  | 发布时间 |
| offline\_time | timestamp |  | 下线时间 |
| create\_time | timestamp | 非空 | 创建时间 |

1. 分类表

表名：genre

表 5 分类表

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 约束 | 说明 |
| id | int | 非空唯一 | 主键 |
| name | varchar(20) | 非空唯一 | 名称 |
| desc | varchar(50) |  | 分类描述 |

1. 收藏表

表名：collect

一个用户收藏多首音乐，一首歌可以被多个用户收藏

表 6 收藏表

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 约束 | 说明 |
| id | int | 非空唯一 | 主键 |
| song\_id | int | 非空 | 歌曲ID |
| user\_id | int | 非空 | 用户ID |

1. 文件表（单独数据库）

表名：fileinfo

表 7 文件表

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 约束 | 说明 |
| id | int | 非空唯一 | 主键 |
| name | varchar(255) | 非空 | 文件名 |
| type | varchar(20) |  | 文件类型 |
| size | double |  | 大小 |
| uuid | varchar(32) |  | 自动生成的uuid，防止重复 |
| MD5 | varchar(255) |  | 防止重复上传 |
| path | varchar(255) |  | 文件路径 |
| createtime | date |  | 创建时间 |
| url | varchar(255) |  | 链接 |

### 数据库与服务器连接

数据库与后台服务器通过mybatis连接，使用mybatis的逆向工程自动生成mapper文件与pojo（Plain Ordinary Java Object）类之后通过services层进行数据库交互。

### 音乐网站服务器

整体架构：ssm（Spring+Spring MVC+Mybatis）架构

Spring MVC工作流程图

* DispatcherServlet：前端控制器，是Spring MVC 整体框架控制的中心
* HandlerMapping：处理器映射器，负责根据用户请求找到对应具体的Handle处理器
* Handler：处理器，业务处理的核心类，通常又开发者编写，描述具体的业务
* HandlAdapter：处理器适配器，通过它对处理器进行执行
* View Resolver：试图解析器，将结果生成View试图
* View：视图，最终产出结果，常用试图如jsp，html

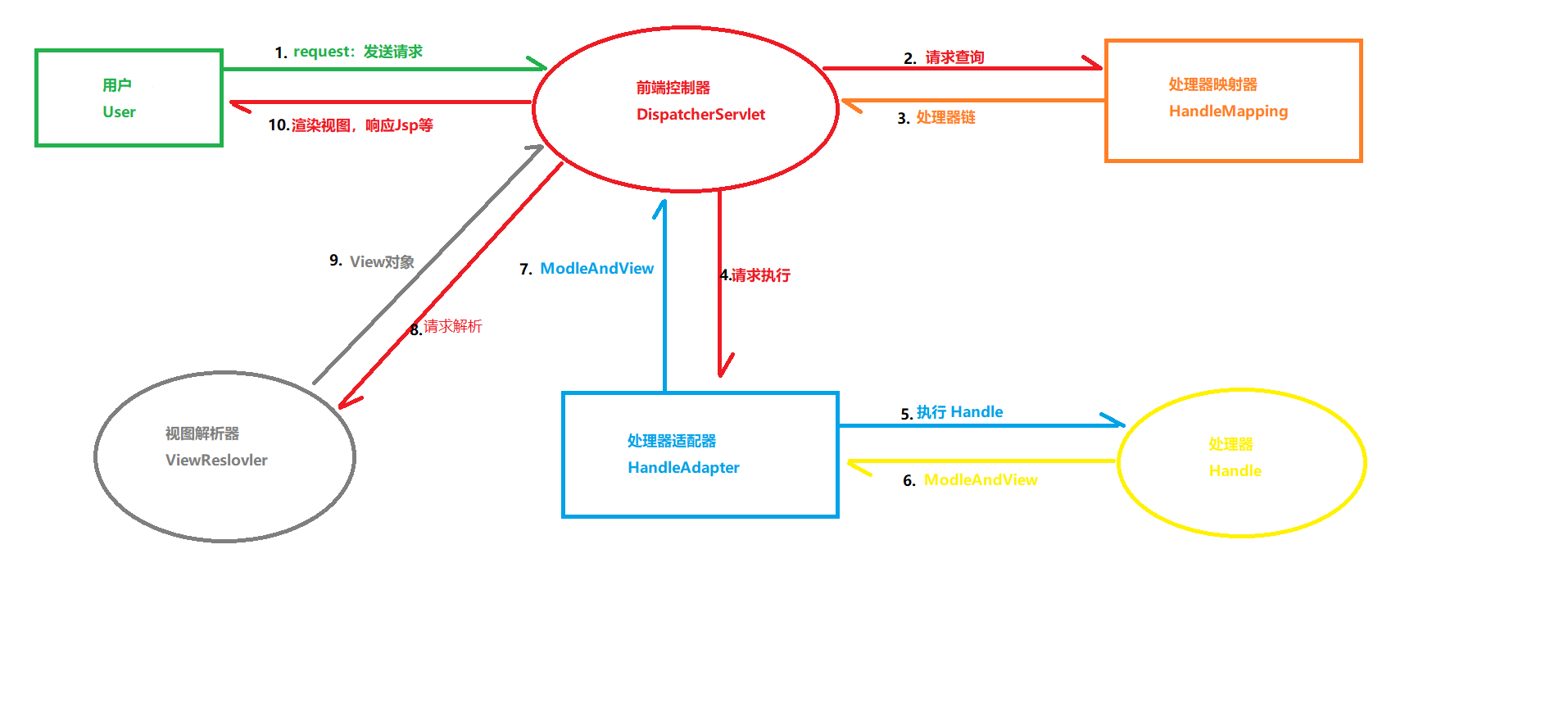


图 2 Spring MVC运行流程

日志管理：log4j

测试类：junit

前后端数据交互：

方式一：通过传输jsp或html

方拾二：通过json数据传输（数据通过jackson插件自动转换为json数据，前段通过解析json数据渲染页面）

### 文件服务器

主要功能：文件服务器负责接受上传文件，并将文件生成链接供音乐网站访问。

实现原理：后端同样适用ssm架构，通过Spring MVC将文件保存到服务器的uploads目录，并将数据存放到数据库中，返回前端上传文件的链接。用户将链接返回给音乐服务器保存以便下次访问。

### 前端

前端包含html,Jsp,Css,Javascript等构建，

使用Bootstrap对页面进行布局与美化。使用了其中的栅格布局，使得网站排版更加清晰明了。还有其中的一些表单样式，将原本的样式进行了修改，使得表单更加美观。还是用了一些js插件如模态框等用来输入数据与消息提示。还有一些提供的图标将网站锦上添花。

使用CPlayer插件实现音乐播放器功能。这个插件可以使得歌曲可以展示进度条，实现播放列表功能，后台播放等。

其中使用了JQuery的ajax请求实现异步请求，使得网站可以在不刷新网页的情况下更新网页的局部数据以实现前端与后台的交互。

## 2.3 实现过程

### 1.构建webapp项目

使用maven创建webapp项目，会自动生成对应的目录

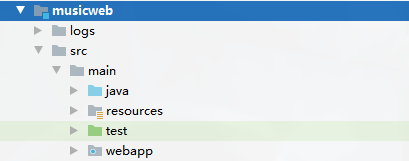


图 3 项目主要目录

### 2.Pom文件相关依赖

* 1. Junit：测试
  2. Log4j: 日志
  3. Spring与Spring MVC相关依赖
  4. Mybatis与Mybatis++：数据库管理
  5. Mysql驱动
  6. Servlet：处理请求，前端控制器
  7. Pagehelper：查询分页

### 3.配置文件与后端相关目录

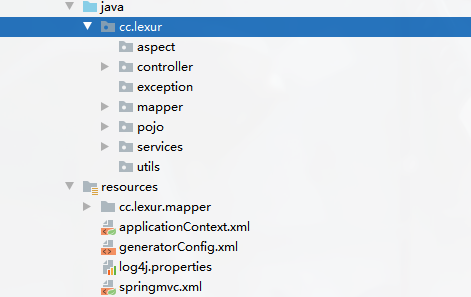


图 4 相关包与配置文件

1. log4j.properties文件为lo4j的配置文件，配置后会自动输出日志文件
2. applicationContext.xml该文件是Spring的配置文件，由Spring对包与类进行管理，开启注释，管理数据库连接，自动生成Mapper的代理类，进行事务管理等。

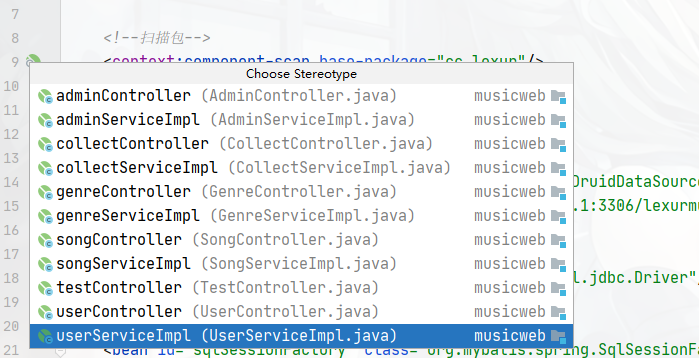


图 5 Spring自动管理

1. springmvc.xml为Spring MVC的配置文件，实现静态资源过滤，扫描controller包，视图解析器，配置异常处理器等。
2. generatorConfig.xml为Mybatis逆向工程配置文件，配置数据库连接，生成文件格式，生成的数据库与其中的表等。

### 4.mybatis逆向工程

（1）编写generatorConfig.xml配置文件

（2）使用测试类自动生成相关mapper文件与pojo类

### 5.services层业务逻辑编写

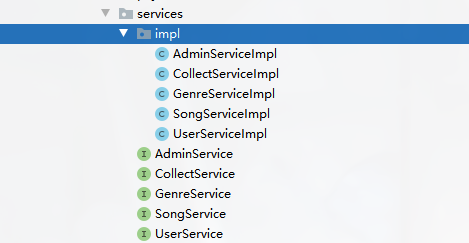


图 6 services层

* 1. Service层：负责业务模块的逻辑应用设计。
  2. 这里使用了接口实现，并通过spring的注释对这些类进行管理，自动生成类
  3. Services层通过调用mapper的方法进行数据库交互

### 6.controller层编写

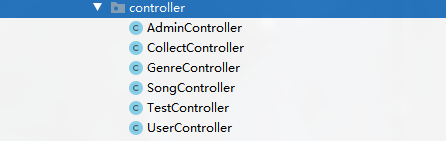


图 7 controller层

1. Controller：前端控制器。接受用户请求响应用户数据。为前端提供API接口，实现前后端数据交互。
2. 这里通过调用services的方法实现API的功能，其中通过注释自动生成对应实现，通过Spring管理各个组件。

### 7.前端编写

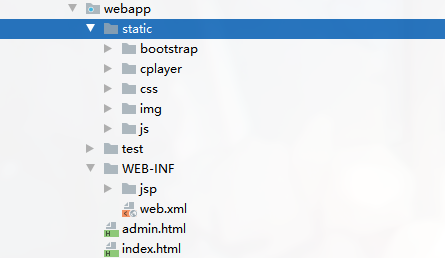


图 8 前端相关文件

1. static中存放了相关的静态资源
2. Boostrap：是一种前端框架，用来快速开发响应式布局，减少前端代码。
3. Cplayer：音乐播放组件
4. Css，img，js：网页相关样式，素材，javascript文件等
5. WEB-INF中存放了后端网页，用户无法直接访问
6. Jsp：存放相关网页，包括html文件与jsp文件
7. web.xml：初始化配置信息
8. index.html：网站主页
9. admin.html：控制台页面

## 2.4 系统实现

### 用户登录

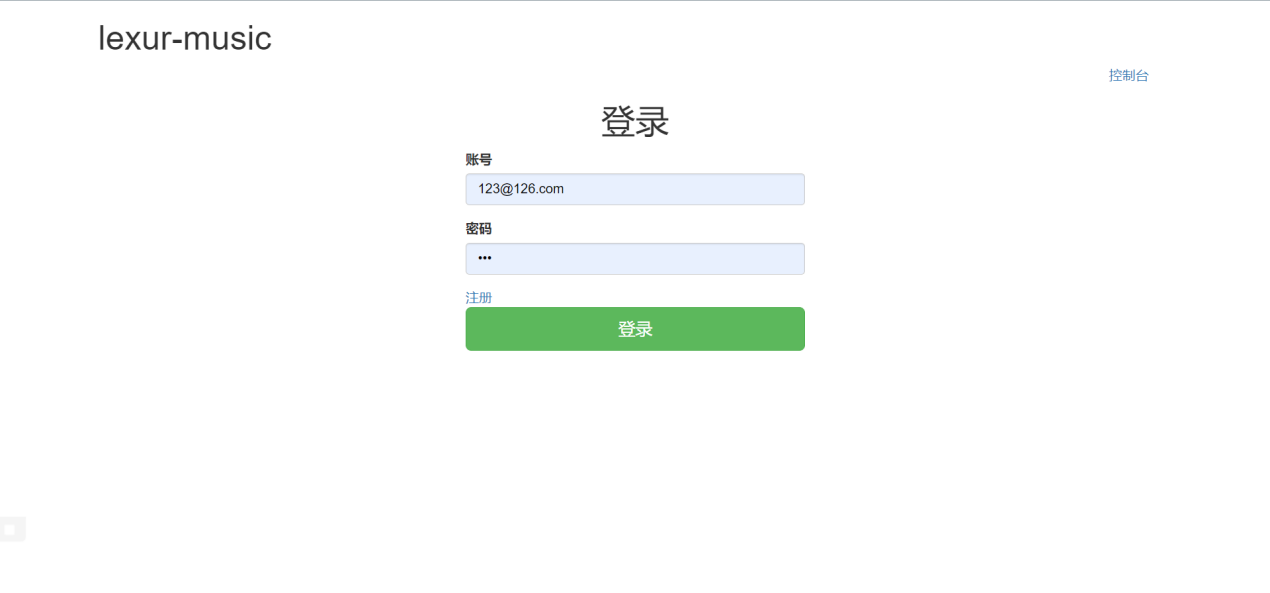


图 10 用户登录

### 用户注册

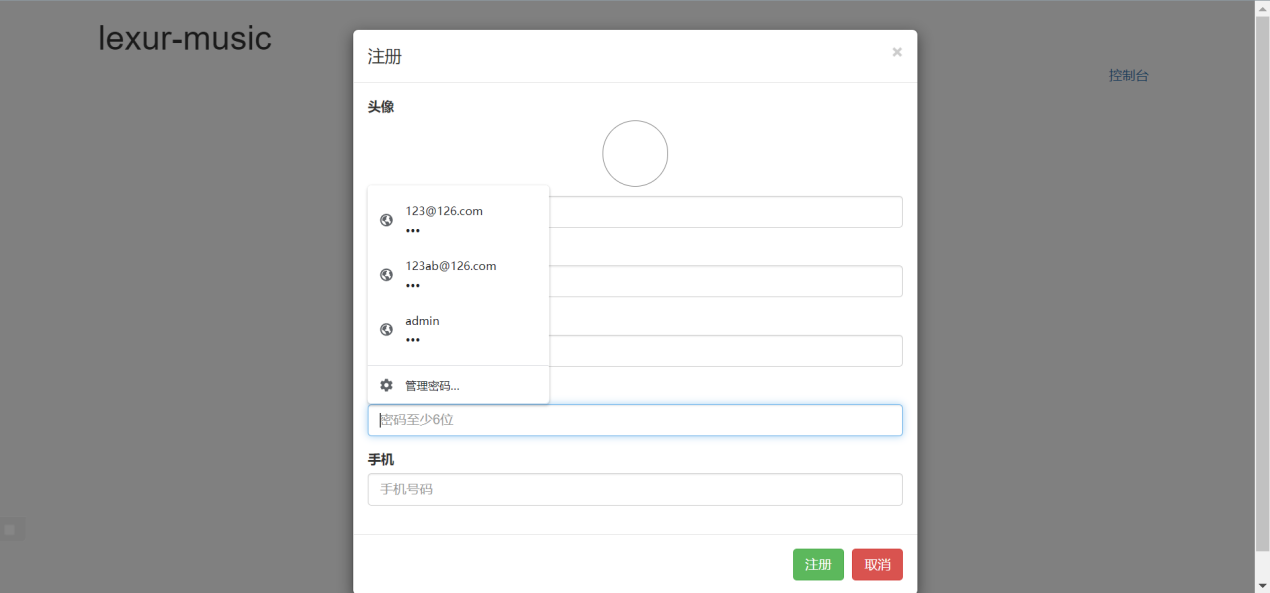


图 11 用户注册

### 音乐网站主页

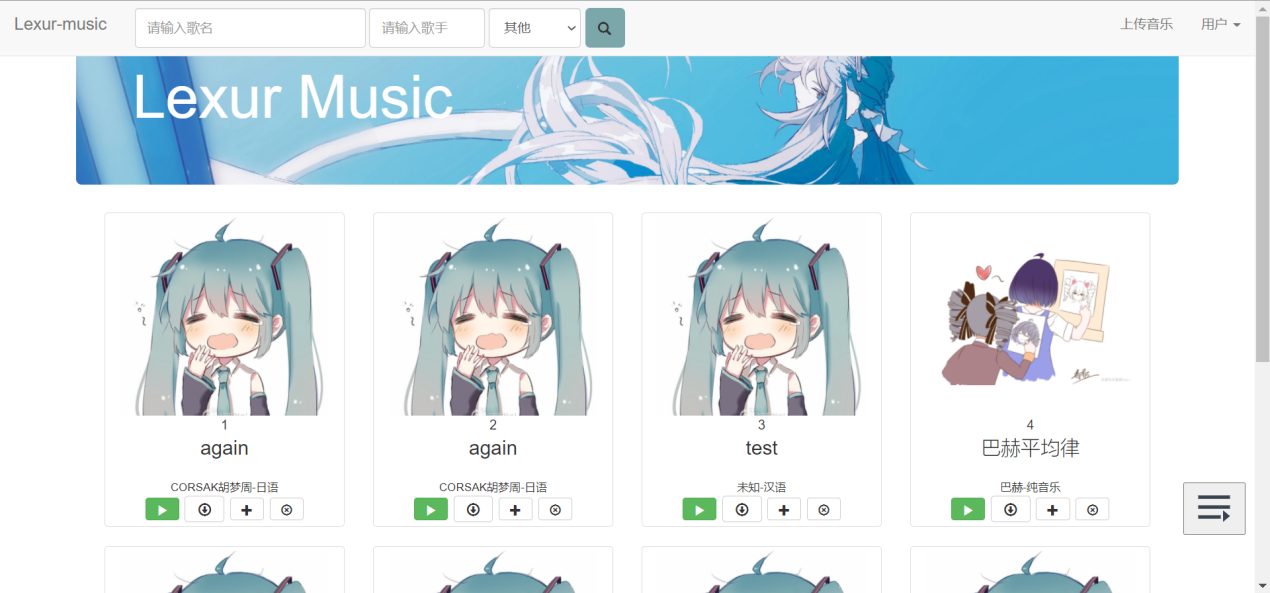


图 12 音乐网站主页

### 音乐播放器

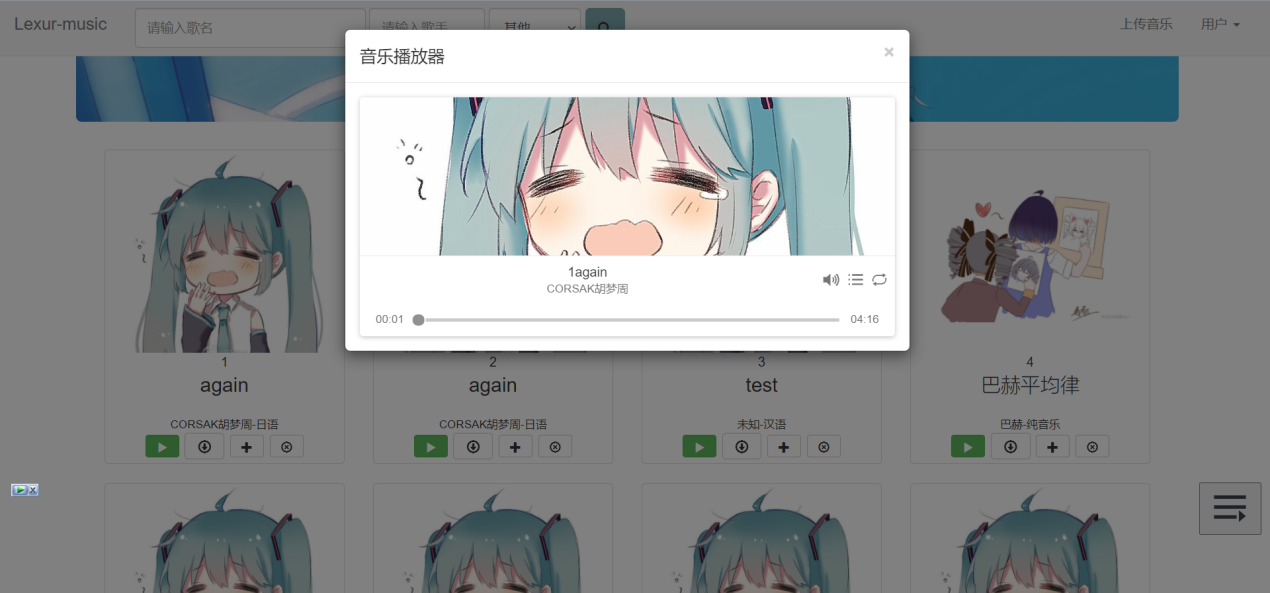


图 13 音乐播放器

### 上传音乐



图 14 上传音乐

### 搜索音乐

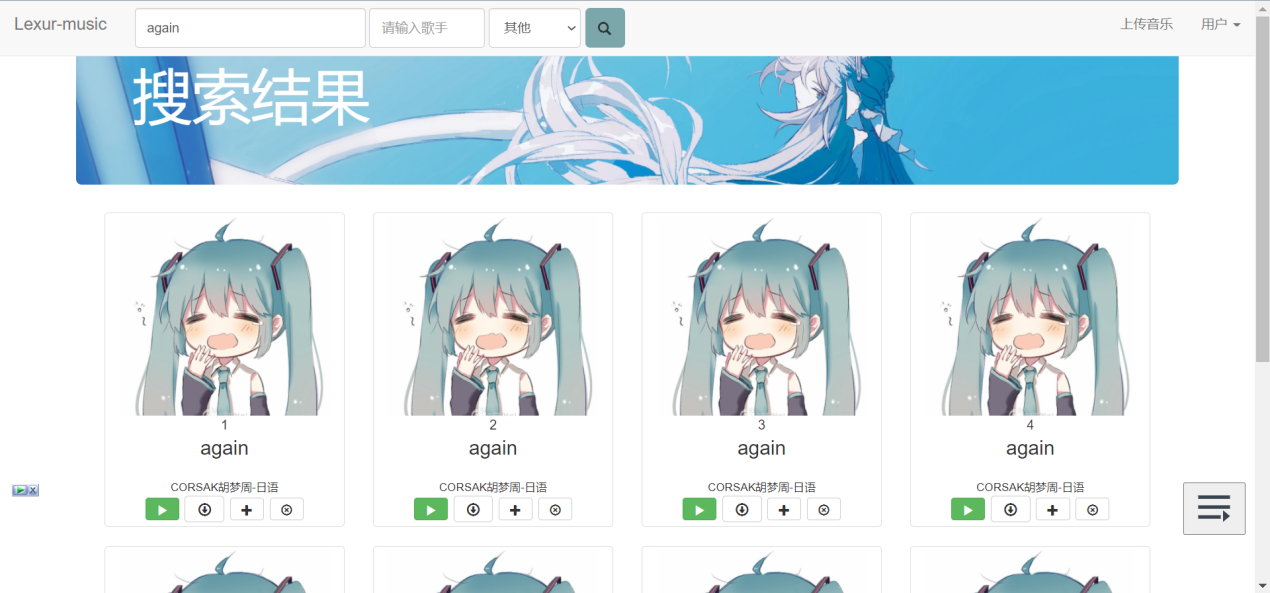


图 15 搜索音乐

### 管理员登录界面



图 16 管理员登录界面

### 管理用户界面



图 17 管理用户界面

### 管理音乐界面

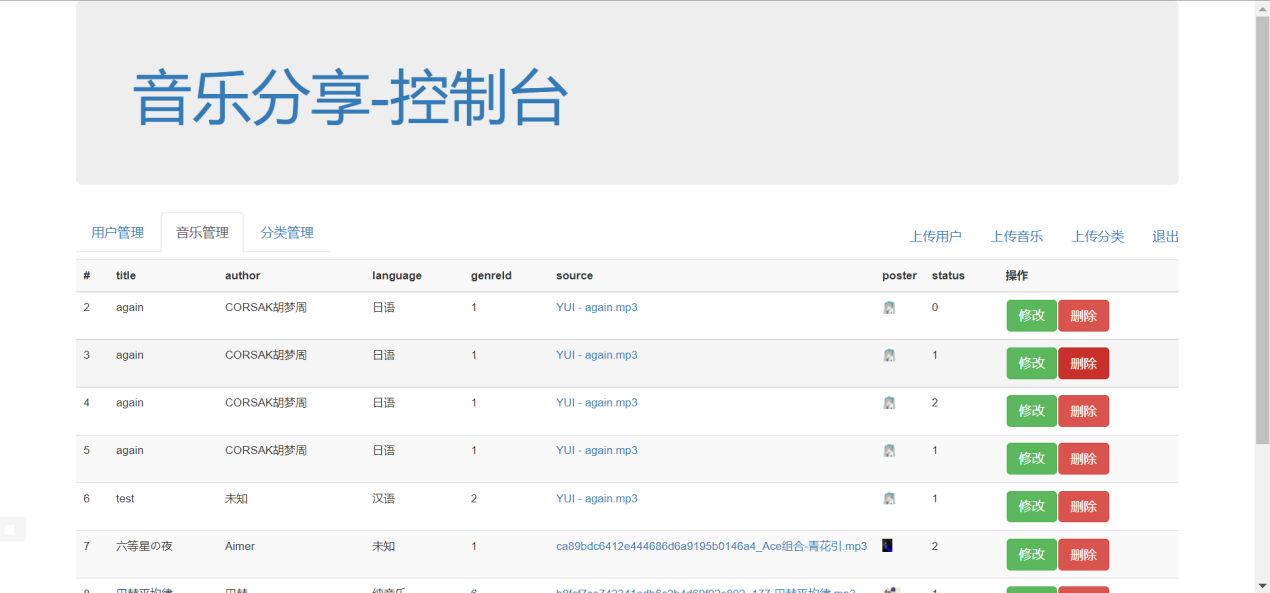


图 18 管理音乐界面

### 管理分类界面



图 19 管理分类界面

### 管理员添加用户、音乐、分类

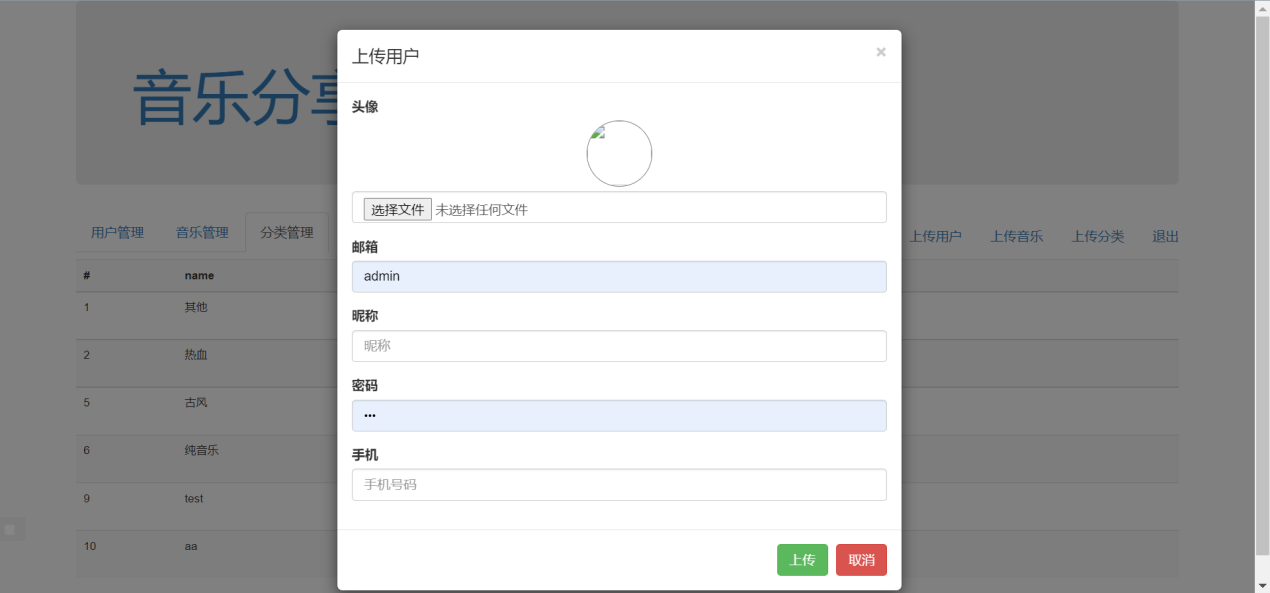


图 20 添加用户



图 21 添加音乐

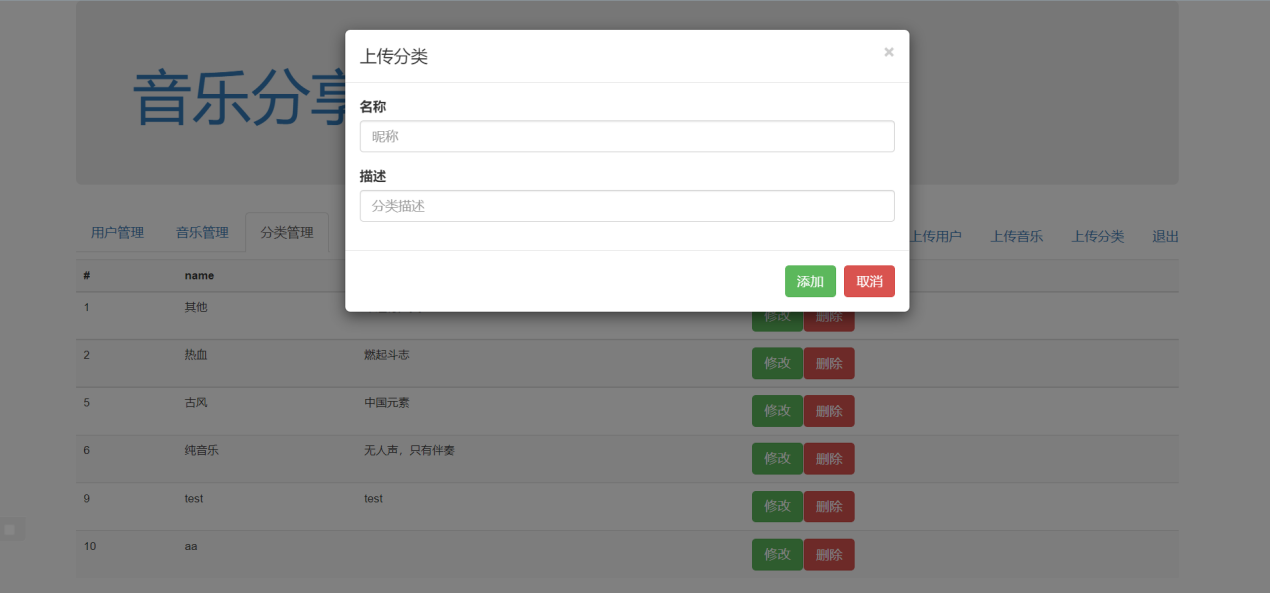


图 22 添加分类

## 2.5重难点

### 1.applicationContext.xml配置

Spring配置是Spring的核心，规定了很对功能的使用。如annotation-config注解实现配置，component-scan 自动扫描类将其转换为对象，同时也是Spring核心功能IoC,AOP功能等的配置。Spring可以通过xml文件对项目进行javabean，组件等的管理，也可已通过注释来实现。



图 10 applicationContext.xml配置

### 2.springmvc.xml配置

Spring MVC是基于servlet封装而成的框架，而DispatcherServlet是SpringMvc的访问点，这里是初始DispatcherServlet的配置。servlet-mapping是对请求的拦截,“/”拦截的类型是servlet，servlet请求会通过DispatcherServlet分发。

如果此文件配置出错，网页将无法打开。同时通过此文件配置可以对前端发来的请求进行处理，同时也会对传输回前端的数据与页面等进行管理。



图 11 springmvc.xml配置

### 3.消息传递

消息传递通过Msg类进行消息传递。通过状态码Code确认消息是否正确，通过Msg传递简单的消息。Msg类使用了HashMap，可以将传输的数据分装到其中。其中从后台获取的数据会通过jackson自动转换为json对象，前端就可以进行解析了。

同时使用了链式编程，可以快速将数据分装到Msg类。

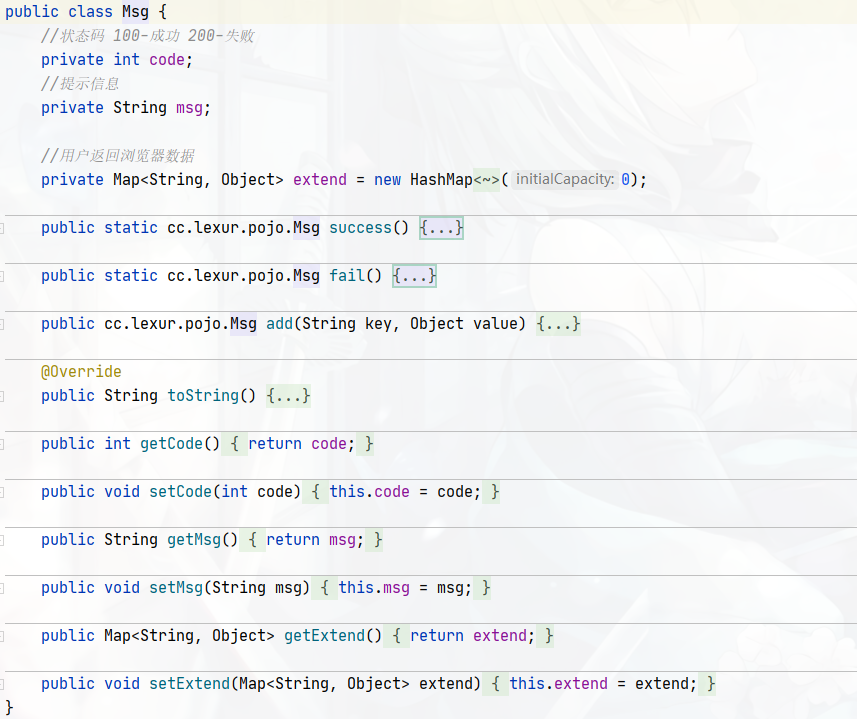


图 12 Msg类

### 4.ajax请求

Ajax：可以实现异步查询，同步查询，同时对返回数据进行处理。

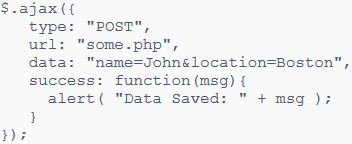


图 13 ajax请求

通过ajax发送异步请求，同时在成功或失败后进行相应的处理。其中有些情况异步请求可能带来一些资源共享的错误，有时数据未请求到已经执行了需要使用数据的函数，因此自请求过程中需要制定好先后顺序，以免不必要的错误。

# 三.总结

# 3.1 实习的收获

通过使用IDEA我了解到了怎么构建项目与IDEA常用的配置，快捷键，快速生成代码来减轻编程负担。本次使用的是maven对项目包进行管理，了解了怎么配置maven环境，添加依赖，通过maven将项目打包成jar包或war包。

通过这次实训我了解到了ssm框架如何运行，初步了解了spring的一些特性：控制反转（IOC），面向切面编程（AOP），依赖注入（DI），容器等特性。

这次项目我使用git来管理项目学会了一些项目管理的技巧，可以通过git提交每次的修改，可以创建分支来进行测试，测试成功后将测试的内容合并到主分支。如果在项目出现无法解决的问题可以通过回溯之前的版本来解决问题。

# 3.2 存在不足

* 未能将war包部署到服务器，服务器的数据库无法通过Navicat管理
* 未能实现评论功能
* 未对传输密码与文件加密
* 链接格式不够统一
* 可以将网站做成前后端分离，这样可以减轻主服务器的压力

# 3.3 今后的努力方向

1.今后需要学习一些前端框架如Vue，Rect等适应现在前端的开发。通过框架可以快速构建前端页面，同时也方便今后对网页的修改，添加插件等功能。通过框架可以快速搭建安全可靠的页面，同时也会更加美观。再大大加快效率的同时也会使得项目更加符合团队开发的要求。

2.需要了解Spring 的一些其他新技术，很多新技术改进了以前的短板与问题。

# 四.致谢

在这次实训中我收获了很多项目开发中相关的知识。首先非常感谢我的导员崔璨与李瑞元对我的培养，在学校的方方面面的帮助。非常感谢我的导师闵芳带我学习了编程相关知识，让我逐步了解了编程的乐趣，让我越来越喜欢编程，学习了c++、java等基础的知识，为以后的编程学习打下了牢固的基础。还有我的数据库原理老师李莹和数据库实现李兰兰教会了我数据库的设计和使用，让我在网站数据库设计中游刃有余。算法设计与分析老师李莹让我了解了数据结构预算法的巧妙。软件工程老师刘瑾让我知道了软件开发的流程，如何将软件开发变得井井有条，如何对自己的项目妥善的管理。还有我的javaEE老师王磊，教会了我如何使用ssm搭建java web项目，让我了解到了框架开发的便捷。通过这次实训我学习到了很多的新知识，同时也逐步了解到了开发java web的流程。感谢这些帮助我的老师和朋友们。