

**毕业设计（论文）**

|  |  |
| --- | --- |
| **题　　目:** | 基于WEB的经典音乐网站 |
|  | 的设计与实现 |
| **学院:** | 信息工程学院 |
| **专业班级:** | 软工（本）1701班 |
| **姓　　名:** | 辛锦康 |
| **学　　号:** | 20171205153 |
| **指导教师:** | 李胜家 教授 吕宝华 助教 |

**二〇二一年五月**

原创性声明

本人郑重声明：本人所呈交的毕业论文，是在指导老师的指导下独立进行研究所取得的成果。毕业论文中凡引用他人已经发表或未发表的成果、数据、观点等，均已明确注明出处。除文中已经注明引用的内容外，不包含任何其他个人或集体已经发表或撰写过的科研成果。对本文的研究成果做出重要贡献的个人和集体，均已在文中以明确方式标明。

本声明的法律责任由本人承担。

论文作者签名： 日期：

关于毕业论文使用授权的声明

本人在指导老师指导下所完成的论文及相关的资料（包括图纸、试验记录、原始数据、实物照片、图片、录音带、设计手稿等），知识产权归属山西应用科技学院。本人完全了解山西应用科技学院有关保存、使用毕业论文的规定，同意学校保存或向国家有关部门或机构送交论文的纸质版和电子版，允许论文被查阅和借阅；本人授权山西应用科技学院可以将本毕业论文的全部或部分内容编入有关数据库进行检索，可以采用任何复制手段保存和汇编本毕业论文。如果发表相关成果，一定征得指导教师同意，且第一署名单位为山西应用科技学院。本人离校后使用毕业论文或与该论文直接相关的学术论文或成果时，第一署名单位仍然为山西应用科技学院。

论文作者签名： 日期：

指导老师签名： 日期：

摘 要

我国有很多知名的音乐网站，但是这些音乐网站大多包含各种流派的音乐，很少有专注只做一种音乐流派的音乐网站，而用户往往都有自己最喜欢的音乐流派，因此个性化的音乐网站具有一定的市场需求，所以经典音乐网站便应运而生。

本系统使用SSM框架开发。使用SpringSecurity框架对用户进行认证授权，从而提高系统的安全性。

系统的核心目标是为喜欢经典音乐的用户提供个性化音乐播放服务。用户可以在网站听自己喜欢的音乐，管理员可以上传音乐，同时也可以对音乐信息进行管理。

**关键词：Java；SSM；经典音乐**

**Abstract**

There are a lot of famous music website in our country, but most of these music website contains various genres of music, do a few focus on music genres of dmusicweb site, and users often have their own favorite music genre, personalized music web site there is a market demand, so classical music website arises at the historic moment.

This system is developed using SSM framework. Use the SpringSecurity framework to authenticate and authorize users to improve system security.

The core goal of the system is to provide personalized music playback service for users who like classic music. Users can listen to their favorite music in the website, the administrator can upload music, but also to the music information management.

**Key words:Java;SSM;Classical music**

目 录

[1 引 言 1](#_Toc14857)

[1.1 研究背景 1](#_Toc20445)

[1.2 国内外现状 1](#_Toc8473)

[1.3 研究的目的及意义 1](#_Toc25201)

[2 开发工具与技术 3](#_Toc17682)

[2.1 IDEA开发工具简介 3](#_Toc22814)

[2.2 Java语言概述 3](#_Toc6495)

[2.3 JSP概述 3](#_Toc4830)

[2.4 SSM框架概述 4](#_Toc10574)

[2.4.1 Spring框架概述 4](#_Toc2086)

[2.4.2 SpringMVC框架概述 4](#_Toc17499)

[2.4.3 Mybatis框架概述 5](#_Toc2032)

[2.5 SpringSecurity框架简介 5](#_Toc3391)

[2.6 关系型数据库的概述 5](#_Toc8989)

[2.7 Mysql数据库简介 6](#_Toc28244)

[2.8 Tomcat服务器概述 6](#_Toc11175)

[3 系统分析 7](#_Toc1654)

[3.1 功能需求分析 7](#_Toc11938)

[3.1.1 前端模块功能需求分析 7](#_Toc31107)

[3.1.2后端模块功能需求分析 7](#_Toc18213)

[3.2 外接口需求分析 7](#_Toc10780)

[3.2.1 用户界面 7](#_Toc22926)

[3.2.2 软件相应接口 8](#_Toc20073)

[3.3 系统用例图 8](#_Toc438)

[3.4 系统开发环境 8](#_Toc8161)

[3.5 可行性分析 8](#_Toc5148)

[3.5.1 市场可行性分析 8](#_Toc9830)

[3.5.2 技术可行性分析 9](#_Toc20749)

[3.5.3 经济可行性分析 9](#_Toc24886)

[4 系统设计 10](#_Toc8781)

[4.1 系统主要工作流程图 10](#_Toc25734)

[4.2 系统功能设计 11](#_Toc16081)

[4.2.1 前台功能设计 11](#_Toc30908)

[4.2.2 后台功能设计 11](#_Toc11658)

[4.3 系统数据流图分析 12](#_Toc28830)

[4.4 数据库的设计 14](#_Toc10072)

[4.4.1 数据库的设计 14](#_Toc22012)

[4.4.2 数据库表的设计 14](#_Toc18584)

[4.5 系统的技术要求 15](#_Toc21938)

[5 系统实现 16](#_Toc2392)

[5.1 用户模块功能的实现 16](#_Toc8313)

[5.1.1 注册功能的实现 16](#_Toc15371)

[5.1.2 登录功能的实现 17](#_Toc8425)

[5.1.3 首页的实现 18](#_Toc31809)

[5.1.4 点歌功能的实现 19](#_Toc31311)

[5.1.5 播放歌曲功能的实现 19](#_Toc28295)

[5.2 管理员模块功能的实现 20](#_Toc31147)

[5.2.1 音乐管理功能的实现 20](#_Toc4998)

[5.2.2 歌手管理功能的实现 21](#_Toc14671)

[5.2.3 专辑管理功能的实现 22](#_Toc15961)

[6 系统测试 23](#_Toc5873)

[6.1 测试类型 23](#_Toc12895)

[6.2 测试结果 24](#_Toc9075)

[6.2.1 测试用例执行结果 24](#_Toc27599)

[6.2.2 测试问题解决 24](#_Toc20198)

[7 总结与展望 25](#_Toc1104)

[参考文献 26](#_Toc20084)

[致 谢 27](#_Toc23950)

# 

# 

# 基于WEB的经典音乐网站的设计与实现

# 1 引 言

随着网络技术以及计算机的发展，网友们对网络的要求也日益加强，在国内众多音乐网站中，有的网站因库曲过多而导致响应速度慢，有的网站歌曲信息不完整，有的网站只有流行音乐比较齐全，而其他流派的音乐难以满足用户的需求。这都使得不少喜欢经典音乐的用户不得不来回访问多个音乐网站。在这种形势下，经典音乐网站便应运而生。网站使用了一些安全框架进行认证授权，为用户建立一个自由、自主、安全的音乐网站。

## 1.1 研究背景

### 现在已经有很多出名的音乐网站，这些音乐网站大多是包含各种流派的歌曲，很少有专注于做一种音乐流派的音乐网站，而用户往往都有自己偏爱的音乐流派，个性化的网站是有一定的市场需求的，所以经典音乐网站应运而生。而经典音乐这一流派，受到很多用户的喜爱，因此有良好的市场前景。

## 1.2 国内外现状

我国在线音乐网站发展风起云涌，我国比较大的音乐品牌有QQ音乐，酷狗音乐，全民k歌，千千音乐，爱音乐，华为音乐，咪咕音乐，虾米音乐。QQ音乐目前已与超过二百家唱片公司进行合作，在线音乐网站方便了人们的生活，有很好的市场。据统计，经典音乐已经开始被越来越多的用户所喜爱。在国外，音乐市场的规模不断发展，音乐市场高度互联网化。

## 1.3 研究的目的及意义

本系统最核心的目标是为喜欢经典音乐的用户提供个性化音乐播放服务。前端用户模块实现用户登录门户网站、账号密码验证功能，实现用户在门户网站进行搜歌、听歌的功能。管理员模块用于管理员管理歌曲，实现管理员对歌曲的上传、删除、查询等操作。

用户和创作者因为线上音乐网站的出现，而省了许多时间和花销，对于用户来说，他们不需要用钱买各种唱片和播放器，而且软件的客户端也不用下载，就能够享受高品质的音乐。对于音乐的创作来说，他们不用和唱片公司签约合作，可以在网站上发布自己的作品。这对部分创作者是非常重要的，不仅可以提升自己的收入，而且不会限制自己的创作思维。

# 

# 2 开发工具与技术

本系统采用SSM框架来开发。开发工具使用的是IDEA，数据库部分使用的是适合小型WEB应用开发的Mysql数据库。

2.1 IDEA开发工具简介

IDEA主要是用于Java编程的一款开发工具，之前很多Java开发者是是用Eclipse，但是由于用IDEA比Eclipse有更多的优势，所以不少开发者开始使用IDEA这款开发工具。特别是在在GIT项目管理、代码自动生成、调试、代码测试等方面上的功能可以说是很优秀的。

## 2.2 Java语言概述

和C++语言一样，Java也是一门面向对象语言。Java在C++的基础上做了很多简化。使学生更容易学习掌握。Java的应用也很广泛，市场上有很多Java方面的岗位。所以，学习Java的人群越来越多。

## 2.3 JSP概述

JSP的真身是Servlet，JSP在第一次受到访问时就会被编译成HttpJspPage类(这个类继承了HttpServlet),它是一个的视图,他可以把从后端获取的数据展示在前端页面，在JSP中，可以用EL表达式获取数据,一个页面可以静态包含另一个页面也可以动态包含另一个页面。SUN公司研制出过一种新的技术，是一种基于浏览器的网页技术，主要用来设计开发BS架构的，这种技术兼容性强，可以跨平台操作，扩展性极强，极大的提高的编程的速度，提高了效率。还可以对动态的信息使用JSP技术来标记直接生成，生成的内容还可以直接封装起来，直接用于其它相同部分，不用再次进行编写，直接可以复制使用，方便简单，大大提高了效率，这样做可以结合在JSP脚本中，在服务器端使用所有的脚本，直接生成使用。在本系统中，系统的所有界面都是用JSP实现的，根据用户的操作界面上的数据时刻都在变化者，充分体现了JSP的动态性。

JSP技术的特点主要包括：

（1）封装性好，可以把通用的语句封装到一起，就可以实现哪里需要再进行编写的时候，重复使用自己提前写好的，这样编写效率得到了改善。

（2）兼容性好,可以在多个地方使用，跨平台的性能很好。在很多的开发工具上都支持JSP技术。

（3）伸缩性能好，可用于较大的war文件或者较小的war文件，具有很大的伸缩性，可大可小，利用率高；

## 2.4 SSM框架概述

SSM是Spring、SpringMVC、MyBatis这三个框架的缩写，不少企业都在使用这个框架进行开发项目。

### 2.4.1 Spring框架概述

在Java这门编程语言中，如果想调用类里面的非静态方法，首先需要new 一个该类的实例化的对象。Spring的IOC容器可以简化这一步的操作，Spring的IOC容器可以对Java类进行统一的管理。当开发者需要用到这些类的实例化对象时，可以直接从容器里面拿，不用再去new对象了。通过Spring的IOC，可以使得程序的耦合性降低，方便开发者进行程序的开发与维护。Spring还有一大特性是AOP特性。通过Spring的AOP特性，可以对重复代码进行集中管理，对方法进行增强。提高开发者的开发效率。

### 2.4.2 SpringMVC框架概述

SpringMVC用于处理业务逻辑层，是最接近用户的框架。在SpringMVC中，其核心是DispatcherServlet。这个框架本质上是对Servlet要做的事情的一种集成。SpringMVC可以降低WEB层的耦合性，使开发者的开发更简单高效。SpringMVC可以无缝地与Spring融合在一起。

SpringMVC框架的优点

（1）SpringMVC框架提供了一整套完善的组件。

SpringMVC不只是提供一个接口，让用户编写自己的实现类来实现功能，而是提供了一组可以直接使用的完整组件。以前的框架为开发者提供了开发环境，开发者必须构建自己的工具，而SpringMVC为开发者提供了现成的工具，开发者可以直接拿来用。

使用SpringMVC框架提供的这些工具是很方便的，只需要添加一些简单的注释，而不需要编写繁琐的代码。

（2）框架的配置简单又不失灵活性。

SpringMVC框架的常用组件已经满足了项目中80%的配置需求，简单的项目不需要怎么配置就可以进行开发。极大方便了开发人员的工作。

（3）代码的可重用性很高。

一个上线稳定运行过的SpringMVC项目，经过简单修改，甚至不用修改，就可以作为另一个新项目开发的基础。这样能在开发新项目时节省很多时间，提高开发的效率。

（4）可扩展性好。

如果SpringMVC框架提供的组件不能满足开发者的需求，开发者可以根据自己的需求，自己编写相应的组件。SpringMVC框架提供了相应的接口，可以进行工具的升级。这样开发者在遇到不常见的问题时，照样可以用SpringMVC框架来解决。

### 2.4.3 Mybatis框架概述

正常情况下，当开发者编写Java代码的时候，如果需要通过Java代码操作数据库中的数据，就需要编写JDBC代码来连接数据库。然后编写对应的SQL语句进行操作。但是编写JDBC代码，有很多繁琐的步骤。而MyBatis的出现，就较好的解决了这一问题。Mybatis底层封装了JDBC，开发者仅仅需要编写SQL语句，就可以实现相应的操作。在编写实体类时，每个类都对应一个映射文件。具体的添加、删除、修改和检查都写在mapper.xml中。XML文件帮助开发者执行SQL语句并在调用时返回数据。一般来说,使用MyBatis有很多优势,如:数据库操作使用XML文件配置,使SQL语句和代码分离,方便开发者对系统的维护，支持编写动态SQL语句。

## 2.5 SpringSecurity框架简介

SpringSecurity框架用于对用户进行认证和授权，它可以被Spring框架整合。使用起来比较方便，开发者只需要编写一些配置文件和配置类就可以实现对应的功能。

## 2.6 关系型数据库的概述

关系型数据库是传统的数据库。SQL语言就是一种基于关系数据库的语言，每一个关系型的数据库都会把数据是放在不同的表中，使得结构非常清晰简单，单表的数据量也大大减小，从而对数据的存取速度大大加快，另外，对数据的维护就变成了对表的维护，性能整体上得到了很大的改善。

## 2.7 Mysql数据库简介

在Mysql中可以把数据放在不同的表中，使得结构非常清晰简单，单表的数据量也大大减小，从而对数据的存取速度大大加快，另外，对数据的维护就变成了对表的维护，性能整体上得到了很大的改善。  
　　Mysql是开放源码的数据库，这是Mysql最大的亮点，由于它的数据库源码是开源的，所以我们完全可以根据自己的需要使用Mysql的源码，在源码的基础上可以做一些小的修改满足我们的需要。

## 2.8 Tomcat服务器概述

Tomcat是一种小型服务器，不适用于大型的网站开发，对于高并发的情况下，需要使用其它的服务器，本经典音乐网站属于一个小型网站，Tomcat服务器足够使用。Tomcat服务器性能稳定，在Apache公司的支持下持续更新升级，功能越来越强大。所以，在校学生大多都会用这个服务器。出于对服务器的使用熟练度，还有系统开发的高效性，本系统决定使用Tomcat服务器。

# 3 系统分析

## 3.1 功能需求分析

### 3.1.1 前端模块功能需求分析

系统使用SpringSecurity安全框架进行认证授权，提高系统的安全性。用户只能进入网站首页听歌，管理员除了可以进入网站首页还可以进入后端管理页面进行相关信息的管理。用户可以注册自己的账号密码登录该网站，登录后可以根据音乐名称的关键字进行搜索，在线播放音乐。

### 3.1.2后端模块功能需求分析

系统的后台模块分为音乐管理模块、歌手管理模块、专辑管理模块。在音乐管理模块，管理员可以上传音乐到服务器，把对应的音乐路径存储在数据库中。可以对音乐的相关信息进行管理。在歌手管理模块，管理员可以对歌手的姓名、年龄等相关信息进行管理，在专辑管理模块，管理员可以对专辑的名称和专辑的发行时间进行管理。

## 3.2 外接口需求分析

### 3.2.1 用户界面

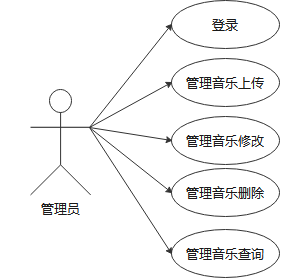
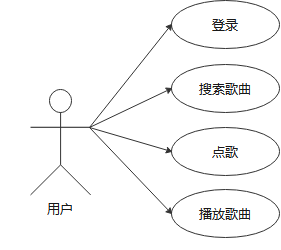
本系统针对不同的角色设计了不同的用户界面，都是采用了可视化的简单易操作的设计思想进行的设计，方便不同用户的功能操作运行，本系统采用了固定的设计布局，对于每个不同的用户功能都采用了单独的JSP编程，每个页面又都用到这些JSP编程，这样保证了系统的整体统一性，每个用户界面都设计好了接口，方便各个权限用户快速分别登陆。

### 3.2.2 软件相应接口

本系统设计开发时对于平台并没有特别的要求，有较好的兼容性，对于Linux系统和Windows系统都可以完美运行，不需要再额外进行编写。对服务器也没有特别的要求，谷歌和现在流行的360等都可以使用，对电脑硬件方面，CPU,内存，硬盘容量和运行内存都没有特别的要求，只要系统可以安装支持IDEA和Mysql就可以使用。

## 3.3 系统用例图

用例图描述了用户可以怎样使用系统。其中，角色用一个小人来表示。在网站中角色用来代表网站的用户，但角色并不仅仅只能代表用户，它还可以用来表示与系统交互的实体。用例描述了系统可以做什么，在真实的系统开发过程中，一个系统里经常要用到许多用例。在UML中，用例用椭圆表示。当角色和用例有一些关系时，它们可以通过一个带有箭头的直线连接起来，这个图叫做用例图。

 图3-1 管理员用例图 图3-2 用户用例图

3.4 系统开发环境

（1）使用Windows10操作系统。

（2）使用Tomcat8.5.24服务器。

## 3.5 可行性分析

### 3.5.1 市场可行性分析

我国在线音乐网站发展风起云涌，我国知名的音乐品牌有QQ音乐，酷狗音乐，全民k歌，千千音乐，爱音乐，华为音乐，咪咕音乐，虾米音乐。QQ音乐目前已与超过二百家唱片公司进行合作，在线音乐网站方便了人们的生活，有很好的市场。据统计，经典音乐已经开始被越来越多的用户所喜爱。在国外，音乐市场的规模不断发展，音乐市场高度互联网化。

现在已经有很多出名的音乐网站，这些音乐网站大多是包含各种流派的歌曲，很少有只做一种音乐流派的音乐网站，而用户往往都有自己偏爱的音乐流派，个性化的网站是有一定的市场需求的，所以经典音乐流派网站应运而生。而经典音乐这一流派，受到很多用户的喜爱，因此有良好的市场前景。

### 3.5.2 技术可行性分析

本系统使用IDEA开发工具开发系统。IDEA是现在主流的Java开发工具。有很好的兼容性和高效性。系统使用Maven进行管理项目，使用8.5版本的Tomcat作为系统的服务器。使用5.7版本的Mysql存储数据。这些软件基本都时开源的，适用于中小型系统的开发。而且自己对这些软件非常熟练。所以，本系统在技术可行性分析上来说是可行的。

### 3.5.3 经济可行性分析

系统使用的软件基本上都是开源的，所以开发成本比较小，在自己可以接受的范围之内。系统是一个小型的经典音乐网站，后台模块只有一个管理员的角色。不需要有很多的人力。所以，系统在经济可行性分析上是可行的。

# 

# 4 系统设计

## 4.1 系统主要工作流程图

此流程图描述了网站的工作流程，用户可以注册账号密码，登录网站听歌。管理员可以对歌曲进行上传，修改，删除等操作。

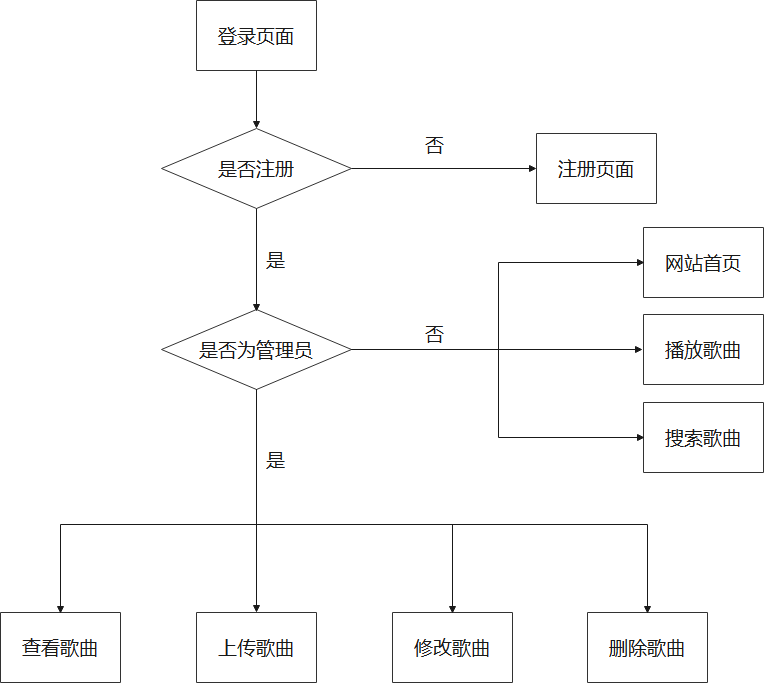


图4-1 系统主要工作流程图

## 4.2 系统功能设计

### 4.2.1 前台功能设计

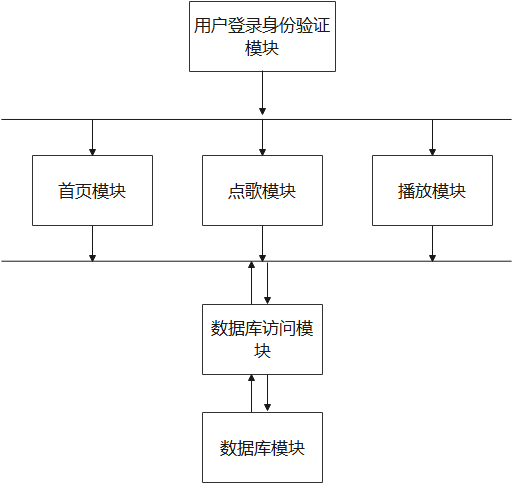
前台功能图如图4-2所示：

图4-2 前台功能模块图

前台功能模块描述：

1. 首页模块：普通用户可以登陆自己的账号密码，可以通过搜索音乐的关键字进行查询。
2. 点歌模块：用户可以挑选自己想听的音乐进行试听。还可以按需求向朋友和家人分享歌曲。
3. 播放模块：用户可以选择自己想听的音乐进行顺序播放。

### 4.2.2 后台功能设计

后台功能模块如图4-3所示：

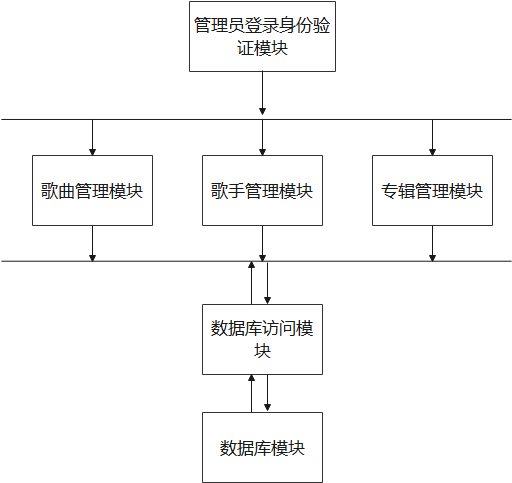


图4-3 后台功能模块

后台功能模块描述：

（1）歌曲管理模块：该模块可以对歌曲进行上传、查询、修改、删除操作。

（2）专辑管理模块：该模块可以进行添加专辑的相关信息，显示专辑的发行时间等信息。

（3）歌手管理模块：该模块可以添加歌手信息，可以对歌手信息进行统一管理。

## 4.3 系统数据流图分析

本系统根据使用该网站的用户角色的不同分为下面两种用户：

（1）管理员：管理员需要手动往数据库中添加，然后根据账号密码进入网站，负责对音乐信息、歌手信息、专辑信息的管理，其中包括音乐信息的增加、删除、修改，以及歌手信息和专辑信息的增加，删除和修改。

（2）用户：用户需要先注册，然后才能登录到网站，在网站中可以在线浏览音乐、搜索音乐信息，以及在线播放音乐。

根据用户和需求进行分析后，可以归纳出整个系统的数据流图如下：

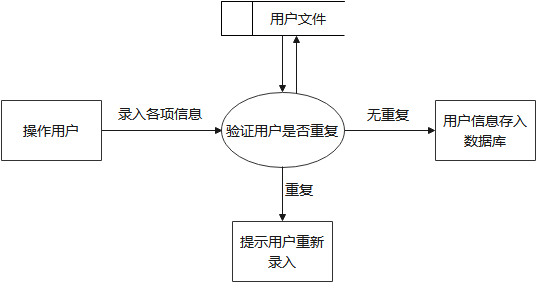


图4-4 用户注册流图

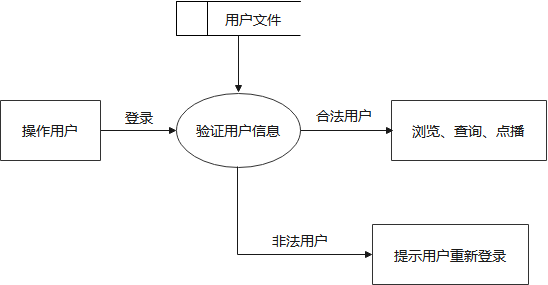


图4-5 用户登录流图

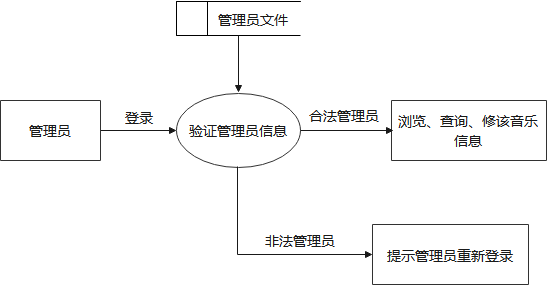


图4-6 管理员登录流图

## 4.4 数据库的设计

### 4.4.1 数据库的设计

### 要设计数据库，设计者必须首先根据需求规范标识实体、关系和属性。利用设计E-R图的方式进行数据库概念设计，设计E-R图时，首先应设计局部E-R图，然后合并局部E-R图，解决可能的冲突。最后，重构初始E-R图，消除冗余部分，最终得到全局E-R图，即概念模式。设计全局E-R模式的目的是消除冲突，使其成为整个系统所有用户都能理解和接受的统一概念模型。

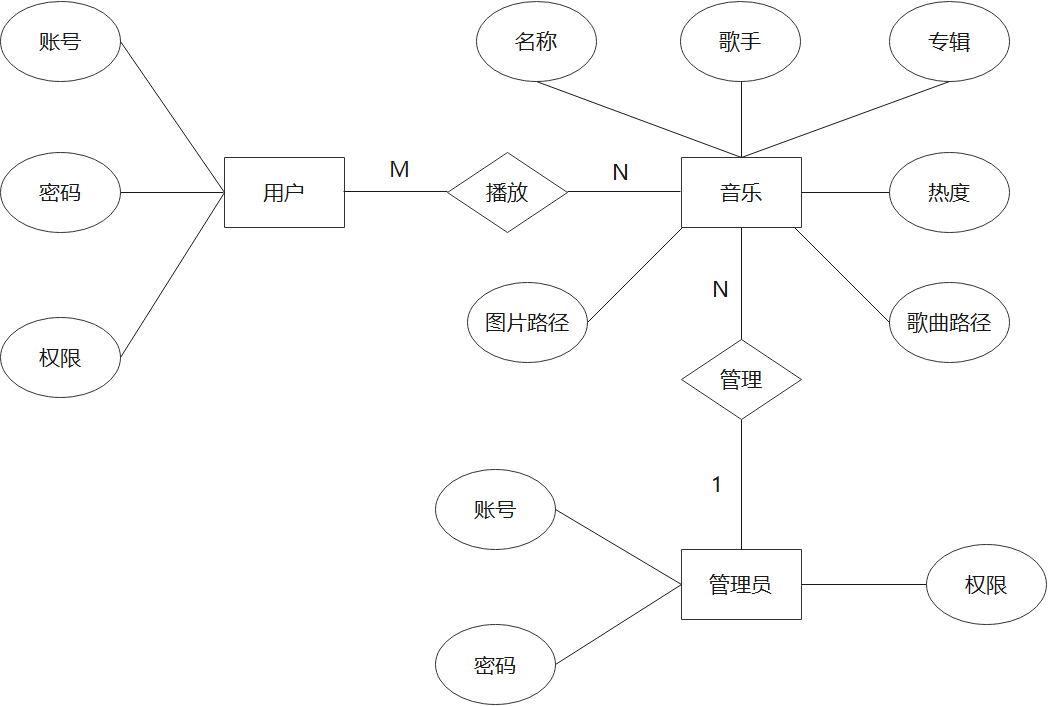


图4-7 系统E-R图

本系统所采用的数据库为Mysql数据库。在Mysql中可以把数据放在不同的表中，使得结构非常清晰简单，单表的数据量也大大减小，从而对数据的存取速度大大加快，另外，对数据的维护就变成了对表的维护，性能整体上得到了很大的改善。

### 4.4.2 数据库表的设计

如表4-1所示，music\_id字段为主键，音乐信息表的每个字段分别代表音乐的一个具体信息。如表4-2所示，id为用户的主键，用户信息表，记录的是注册用户和管理员的所有信息，记录了注册用户和管理员的所有的个人信息包括系统的登录账户和密码。empower字段用于区分用户和管理员。当empower为1时，该账户为用户。wmpower不为1时，该账户为管理员。 如表4-3所示，歌手信息表，记录的是歌手的相关信息。

表4-1 音乐信息表

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **类型** | **长度** | **说明** |
| music\_id | int | 20 | 主键 |
| music\_name | varchar | 50 | 音乐名称 |
| music\_singer | varchar | 50 | 歌手 |
| create\_date | datetime | 50 | 上传时间 |
| album | varchar | 50 | 专辑 |
| cover\_path | varchar | 100 | 图片路径 |
| music\_path | varchar | 100 | 歌曲路径 |
| hits\_nums | int | 20 | 热度 |
| is\_del | int | 20 | 标识音乐是否删除 |

表4-2 用户信息表

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **类型** | **长度** | **说明** |
| id | int | 20 | 主键 |
| username | varchar | 50 | 用户名 |
| password | varchar | 50 | 用户密码 |
| empower | int | 5 | 权限标识 |

表4-3 歌手信息表

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **类型** | **长度** | **说明** |
| id | int | 20 | 主键 |
| name | varchar | 10 | 歌手名称 |
| sex | varchar | 10 | 歌手性别 |
| age | int | 10 | 歌手年龄 |

## 4.5 系统的技术要求

（1）功能要求：

①用户可以快速浏览网站。②用户和管理员有不同的权限。③系统使用分页查询，减少数据库的压力。④用户可以在线听歌。管理员可以管理歌曲信息、专辑信息、歌手信息。

（2）编码要求：

系统统一使用UTF-8编码，支持中文格式，系统能够把数据库中的中文数据展示到前端页面。能把前端的中文数据传给后端，存入数据库。

（3）界面要求：

①网站的各个页面设计美观大方。②首页的导航设计简单清晰，方便用户跳转到其他页面。

5 系统实现

## 5.1 用户模块功能的实现

### 5.1.1 注册功能的实现

用户需要先注册一个账号，才能登录到首页播放音乐，账号密码的长度至少为6，为了保证用户密码的安全性，系统使用MD5对密码进行了加密。注册成功后会重定向到登录界面。实现代码如下：  
if(username!=null&&username.length()>=6&&password!=null&&password.length()>=6)  
{String securityPassword = bCryptPasswordEncoder.encode(password);  
User user=new User(0,username,securityPassword,1);  
int i = userDao.addUser(user);  
response.sendRedirect("/login.JSP");

注册界面如图5-1所示：



图5-1 注册界面

### 5.1.2 登录功能的实现

用户需要通过登录才能进入首页听歌。管理员需要通过登录才能进入后台管理页面，进行对音乐信息的管理。本系统使用SpringSecurity安全框架进行认证与授权，用户可以听歌，但是没有管理音乐信息的权限。管理员拥有最高权限。能既能听歌也能对音乐信息进行管理。在设计user表的时候，为了区分用户和管理员，设置了emppwer字段，当emppwer=1时，为用户，当emppwer=2时，为管理员。用户登录时，后台会在user表中查询有无对应的数据。倘若用户的账号和密码不对应，会给出提示，在登录框下方显示账号或密码不正确，这种交互式的方式，可以提高用户体验。

登录界面如图5-2所示：



图5-2 登录界面

### 5.1.3 首页的实现

用户进入首页可以搜索自己喜欢的歌曲，进行播放。用户进入首页时，首页会显示5条音乐信息。当用户点击查看个多歌曲后，音乐信息的条数会增加5条。这样可以减轻数据库的压力，提高查询数据的速度。系统在前端页面，通过使用forEach标签把后端传来的音乐信息进行遍历展示，实现的部分代码如下：<c:forEachitems="${result}" varStatus="vs" var="music">  
<trdata-index="${vs.index}"><inputtype="checkbox"name="chartids"value="${music.musicId}"><imgsrc="${music.coverPath}"alt="${music.musicName}"height="55"width="55"></c:forEach>

首页如图5-3所示：



图5-3 首页

### 5.1.4 点歌功能的实现

### 在首页，每个歌曲的前面都有对应的一个复选框，复选框的值是对应歌曲的id，用户可以通过点击复选框，选择对应的歌曲id，然后点击播放，对应music.jsp页面会使用window对象把所选的id传给播放页面。实现的部分代码如下：

function playsongs() {  
 var chk\_value =[];  
 $('input[name="chartids"]:checked').each(function(){  
 chk\_value.push($(this).val()); });  
 var ids = chk\_value.join(',');  
 window.location.href = '/portal/play.do?ids='+ids;  
}

### 5.1.5 播放歌曲功能的实现

用户在首页选好自己想听的音乐后，点击播放，把用户所选的id传给播放页面，播放页面会发送ajax异步请求，向后台传入用户所选音乐的id，然后通过使用Mybatis在数据库查到对应的歌曲，封装到Music类中，以Json数据格式返回给前台。实现的部分代码如下：

$.ajax({  
url:'/portal/loadPlayList.do?ids=’+ids,  
type:'get',  
dataType:'json',success:function(data){}});

播放歌曲页面如图5-4所示：

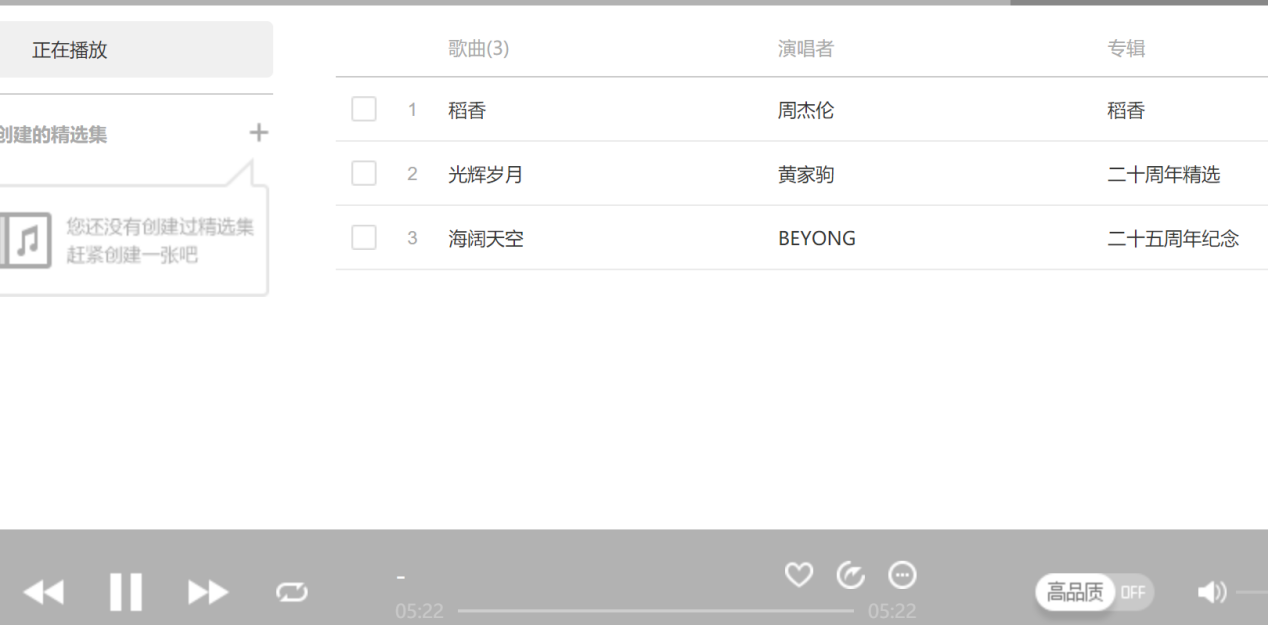


图5-4 播放页面

## 5.2 管理员模块功能的实现

### 5.2.1 音乐管理功能的实现

后台管理员有进入音乐管理页面对歌曲信息管理的权限，系统使用SpringMVC框架进行上传歌曲，把歌曲文件上传到服务器中，把歌曲文件存储在数据库。上传图片时，把图片文件上传到服务器。把图片存储路径存储在数据库。

为了提高系统的性能，提高客户的体验，音乐信息采用分页展示的方法。编写了Page类，里面有当前页、总页数、总记录数、每页展示的歌曲数量，使用SQL的语句：select \* from music limit a，b进行查询。a表示起始值。b表示查询的数量。

用户在页面点击上一页时，a减去每页展示的歌曲数量，然后传给后台，后台进行分页查询后把查询的数据返回给前台。点击下一页时，a加上每页展示的歌曲数量。然后进行对应的操作。

歌曲管理页面如图5-5所示：



图5-5 歌曲管理页面

### 5.2.2 歌手管理功能的实现

管理员可以对歌手姓名、年龄等相关信息进行管理。歌手有一个id属性，在数据库中设置为自动递增。所以管理员每次往数据库添加歌手信息时，无需手动添加id。这样可以使管理员更加方便的管理系统。

歌手管理模块如图5-6所示：



图5-6 歌手管理页面

### 5.2.3 专辑管理功能的实现

管理员可以对专辑名称和发行时间进行管理。专辑发行时间显示具体的年月日。管理员对专辑相关信息进行管理时，先会访问GScontroller类，然后调用业务层，实现相应的业务逻辑，最后访问数据库，进行相关的操作。操作完成后前端控制器把结果返回给前端页面。

专辑管理模块如图5-7所示：



## 

## 、

图5-7 专辑管理页面

# 6 系统测试

系统测试是指在系统开发后为了验证系统预期的效果所执行的环节。测试的主要目的是检查系统纰漏，通过检测看系统运行中是否出现了可调式错误，发现错误修正错误，直到程序符合预期。系统出错率越低，安全性稳定性就越高，用户使用率也就越高。测试的结果也对设计者有重要意义，因为系统软件的品质、性能、可靠性、安全性等质量要求均需通过测试来实现，测试结果与理论预期的偏差程度直接关系到系统实现现实长足的发展。

## 6.1 测试类型

|  |  |  |  |
| --- | --- | --- | --- |
| **测试类型** | **测试内容** | **测试目的** | **测试方法** |
| 功能测试 | 用户：注册新用户、登录系统、浏览首页、搜索歌曲、播放歌曲。  管理后台：管理员登录系统，管理音乐信息、专辑信息、歌手信息。 | 核实系统核心功能  流程检测：每个业务流程能正常进行，用户使用时不会报错。  数据精准：对音乐的数量，用户的个数统计精准。 | 黑盒测试、手工测试 |
| 界面测试 | 导航、链接、网页背景、颜色、字体大小、按钮大小、响应信息等。 | 检测网页界面的设计是否合理，总体感觉是否美观，是否能够满足用户的需求。 | 黑盒测试、白盒测试 |
| 安全性测试和访问测试 | 分别登录用户的账号和管理员的账号。通过修改URL非法访问。 | 检测用户是否有过多的权限。系统对权限管理是否正常。检测系统会不会对没有登录的用户进行拦截。 | 黑盒测试 |
| 性能测试 | 大量用户进行访问系统，注册新用户，搜索歌曲。在线听歌。 | 检测系统在大流量的数据与多用户操作时系统性能的稳定性。 | 黑盒测试 |

表6-1 测试类型

## 6.2 测试结果

### 6.2.1 测试用例执行结果

表6-2 测试用例执行结果

|  |  |  |
| --- | --- | --- |
| **测试用例名称** | **用例状态** | **测试结果** |
| 用户注册 | 已执行 | 已测试通过 |
| 用户登录 | 已执行 | 已测试通过 |
| 试听音乐 | 已执行 | 已测试通过 |
| 搜素音乐 | 已执行 | 已测试通过 |
| 分页显示歌曲 | 已执行 | 已测试通过 |
| 上传歌曲 | 已执行 | 已测试通过 |
| 修改歌曲 | 已执行 | 已测试通过 |
| 删除歌曲 | 已执行 | 已测试通过 |
| 权限管理 | 已执行 | 已测试通过 |

### 6.2.2 测试问题解决

表6-3 测试问题解决

|  |  |
| --- | --- |
| **错误或问题描述** | **错误或问题状态** |
| 用户登录失败时，提示信息没有显示 | 已解决 |
| 用户注册时单击提交后，未跳转至其它页面 | 已解决 |
| 后台登录界面显示信息不完整 | 已解决 |
| 搜索音乐出错 | 已解决 |

## 

## 7 总结与展望

在这几个月的开发过程，我从相关的书籍学到了许多关于开发音乐网站的知识，从导师的指导中学到了新的开发思路。对于开发音乐网站，我也有了一些自己的见解。我认为音乐网站的开发有三个重要的方面:(1)首页上的导航设计要清晰明了，让用户直接可以看明白怎样跳转其他的页面，不让用户浪费时间，甚至影响用户的情绪。(2)系统的可维护性。系统维护很重要，我们的国家日新月异，互联网也在不断的发展，用户对音乐网站的需求，也在不断变化。因此，系统应当随着用户的需求不断改进，以满足更多用户的需求。(3)系统的安全性，如果一个系统不安全，那么系统随时可能瘫痪。用户的信息随时可能泄露，所以说，安全性对于一个系统来说是至关重要的。

本系统已经初步实现，然而，仅仅用三个月的时间来分析、设计和开发一个完整的系统是远远不够的。该系统还有许多地方需要改进。例如，系统的主页面需要进一步的美观，系统的功能模块较少，需要增加新的功能。系统的内容也需要更加丰富和多样化，系统还可能存在一些未知问题。我会继续努力，逐步完善系统的各个功能模块。回顾这三个月的系统开发过程，我觉得一个网站从无到有的开发是比较辛苦的，但是成功后会有很大的成就感。努力的越多，成功后的成就感也就越强。

参考文献

[1]吴教育.WEB开发技术[M].北京:清华大学出版社,2018.

[2]李卫东.零基础Java入门到精通[M].广东:人民出版社,2017.

[3]张新.网页制作入门到精通[M].北京:中国青年出版社,2017.

[4]李建中.数据库系统原理[M].北京:电子工业出版社,2018.

[5]贺林林.音乐网站的开发[J].电子测试,2019.

[6]徐磊.Jquery实战[M].湖北:华中科技大学出版社,2017.

[7][徐郡明](https://book.jd.com/writer/%E5%BE%90%E9%83%A1%E6%98%8E_1.html" \t "https://item.jd.com/_blank).MyBatis技术[M].北京:[电子工业出版社](https://book.jd.com/publish/%E7%94%B5%E5%AD%90%E5%B7%A5%E4%B8%9A%E5%87%BA%E7%89%88%E7%A4%BE_1.html" \o "电子工业出版社" \t "https://item.jd.com/_blank),2017.

[8]吴为胜.SSM实战[M].北京:清华大学出版社,2019.

[9]张雄.重构改善既有代码的设计[M].广东:人民邮电出版社，2017.

[10]刘欣.码农翻身[M].北京:电子工业出版社，2018.

[11][王耀](https://book.jd.com/writer/%E7%8E%8B%E8%80%80_1.html" \t "https://item.jd.com/_blank).深入理解SpringMVC[M].北京:[中国水利水电出版社](https://book.jd.com/publish/%E4%B8%AD%E5%9B%BD%E6%B0%B4%E5%88%A9%E6%B0%B4%E7%94%B5%E5%87%BA%E7%89%88%E7%A4%BE_1.html" \o "中国水利水电出版社" \t "https://item.jd.com/_blank),2019.

[12]张浩.Mysql5.7从入门到精通[M].北京:清华大学出版社,2017.

[13][李春](https://book.jd.com/writer/%E6%9D%8E%E6%98%A5_1.html" \t "https://item.jd.com/_blank)等.Mysql性能优化法则[M].北京:电子工业出版社,2019.

[14][孙卫琴](https://book.jd.com/writer/%E5%AD%99%E5%8D%AB%E7%90%B4_1.html" \t "https://item.jd.com/_blank).Tomcat技术详解[M].北京:[电子工业出版社](https://book.jd.com/publish/%E7%94%B5%E5%AD%90%E5%B7%A5%E4%B8%9A%E5%87%BA%E7%89%88%E7%A4%BE_1.html" \o "电子工业出版社" \t "https://item.jd.com/_blank),2019.

[15][谭勇德](https://book.jd.com/writer/%E8%B0%AD%E5%8B%87%E5%BE%B7_1.html" \t "https://item.jd.com/_blank).设计模式就该这样学[M].北京:[电子工业出版社](https://book.jd.com/publish/%E7%94%B5%E5%AD%90%E5%B7%A5%E4%B8%9A%E5%87%BA%E7%89%88%E7%A4%BE_1.html" \o "电子工业出版社" \t "https://item.jd.com/_blank),2020.

[16][Flanagan](https://book.jd.com/writer/David Flanagan_1.html" \t "https://item.jd.com/_blank),D.JavaScript[M].SanDiego:Halstead,2020.

[17][Forta](https://book.jd.com/writer/Ben Forta_1.html" \t "https://item.jd.com/_blank),B.Mustknowand must know Mysql[M].New York:Macmillan,2020.

[18][Singh](https://book.jd.com/writer/Kamalmeet Singh_1.html" \t "https://item.jd.com/_blank),K.Java development[M].NewYork:Scholars,2019.

**致 谢**

我在设计和开发这个系统的过程中，老师多次给我指出存在的问题，帮助我改善系统各个模块的功能。老师认真的指导和独特的思维方式给予我很大的启发。使我的知识水平和解决实际问题的能力上有了很大的提高，这对于我以后的工作和学习来说也是一种很大的帮助。快毕业了，我的内心无法平静，从项目的开始到论文的顺利完成，有许多老师、同学、给予了我关心和帮助，在这里请接受我真诚的感谢!我还要感谢把我从小扶养长大的父母和在我背后永远支持我的亲人和朋友，感谢你们！感谢我的导师，也感谢各位老师四年来的培养和同学四年的照顾。正是因为你们的支持和帮助，我才能够解决掉一个一个的难题，直至本论文的顺利完成。