# 摘 要

随着生活品质的改变，音乐往往成了人们烦恼时心灵的归宿。音乐播放器便是播放音乐的工具，音乐播放器是一种用于播放各种音乐文件的多媒体播放软件，而音乐播放系统便是存在于网页的在线音乐播放器。它涵盖了各种格式的播放工具，比如：MP3播放器，MP4播放器等。本文介绍了基于HTML的音乐播放系统，其运用了CSS装饰页面，以及BootStrap框架来实现页面响应式布局，用JavaScript实现页面动态效果，Node.JS编写后台。用户信息则使用了MySql数据库，建立user表，来存储。

**关键词：**BootStrap框架、音乐播放系统、NodeJS、MySql数据库

目录

[摘 要 1](#_Toc526945262)

[一 绪论 1](#_Toc526945263)

[**1.1目的** 1](#_Toc526945264)

[1.2内容 1](#_Toc526945265)

[1.3技术领域 1](#_Toc526945266)

[二 需求分析 2](#_Toc526945267)

[**2.1 系统需求概述** 2](#_Toc526945268)

[**2.2 功能需求分析** 2](#_Toc526945269)

[2.2.1 用户管理模块需求分析 2](#_Toc526945270)

[2.2.2歌曲推荐模块需求分析 2](#_Toc526945271)

[2.2.3歌曲査询模块需求分析 3](#_Toc526945272)

[三 系统架构设计 4](#_Toc526945273)

[3.1系统体系架构设计 4](#_Toc526945274)

[3.2 bootstrap框架介绍 4](#_Toc526945275)

[3.3 vue.js框架介绍 4](#_Toc526945276)

[3.4系统功能架构设计 4](#_Toc526945277)

[四 系统详细设计 6](#_Toc526945278)

[4.1音乐播放系统的整体结构 6](#_Toc526945279)

[4.2系统各功能模块的详细设计 6](#_Toc526945280)

[4.2.1 banner图实现 6](#_Toc526945281)

[4.2.2歌单动态加载实现 7](#_Toc526945282)

[4.2.3歌单详细界面实现 7](#_Toc526945283)

[4.2.4 音乐播放窗口实现 8](#_Toc526945284)

[4.2.5 MV播放界面设计 8](#_Toc526945285)

[4.2.6歌曲搜索实现 10](#_Toc526945286)

[4.2.7用户收藏及登录设计 10](#_Toc526945287)

[五 实现遇到的问题及解决方法 11](#_Toc526945288)

[5.1查询问题 11](#_Toc526945289)

[结论 12](#_Toc526945290)

[致 谢 13](#_Toc526945291)

[参考文献 14](#_Toc526945292)

# 一 绪论

1.1目的

随着生活品质的改变，音乐往往成了人们烦恼时心灵的归宿。在这车水马龙的城市生活中，歌声倾诉着我们的心声。音乐播放系统可以达到人们日常对音乐欣赏的要求。

## 1.2内容

音乐播放系统拥有着有限的歌曲，MV库存，可获取有限的歌曲和MV。

对用户而言则添加了用户对歌曲及MV的收藏功能，用户可登录后获取自己收藏的歌曲及MV的信息。

## 1.3技术领域

Bootstrap提供了优雅的HTML和CSS规范，它是由动态CSS语言Less写成。

JavaScript是一种动态类型、弱类型、基于原型的语言，内置支持类型。它的[解释器](https://baike.so.com/doc/5904938.html)被称为JavaScript引擎，为[浏览器](https://baike.so.com/doc/2920715.html" \t "https://baike.so.com/doc/_blank)的一部分，用来给HTML网页增加动态功能。

Node.JS是一个Javascript运行环境，由Ryan Dahl开发，实质是对Chrome V8引擎进行了封装。Node.js 不是一个 JavaScript [框架](https://baike.so.com/doc/1863840-1971314.html)，不同于CakePHP、[Django](https://baike.so.com/doc/5344143-8197406.html" \t "https://baike.so.com/doc/_blank)、[Rails](https://baike.so.com/doc/5499393-5736829.html" \t "https://baike.so.com/doc/_blank)。Node.js 更不是浏览器端的库，不能与 jQuery、ExtJS 相提并论。Node.js 是一个让 JavaScript 运行在[服务端](https://baike.so.com/doc/5988965-6201932.html" \t "https://baike.so.com/doc/_blank)的开发平台，它让 JavaScript 成为与[PHP](https://baike.so.com/doc/5336503-5571942.html" \t "https://baike.so.com/doc/_blank)、[Python](https://baike.so.com/doc/1790119-1892991.html" \t "https://baike.so.com/doc/_blank)、[Perl](https://baike.so.com/doc/5106225-5334848.html" \t "https://baike.so.com/doc/_blank)、[Ruby](https://baike.so.com/doc/4990463-5214197.html" \t "https://baike.so.com/doc/_blank) 等服务端语言平起平坐的[脚本语言](https://baike.so.com/doc/2874347-3033293.html" \t "https://baike.so.com/doc/_blank)。

Node.js对一些特殊用例进行优化，提供替代的[API](https://baike.so.com/doc/24846846-25783482.html" \t "https://baike.so.com/doc/_blank)，使得V8在非浏览器环境下运行得更好。V8引擎执行Javascript的速度非常快，性能非常好。 Node.js是一个基于Chrome JavaScript运行时建立的平台， 用于方便地搭建响应速度快、易于扩展的网络应用。Node.js 使用[事件驱动](https://baike.so.com/doc/357634-378877.html" \t "https://baike.so.com/doc/_blank)， 非阻塞[I/O](https://baike.so.com/doc/3034590-3199433.html" \t "https://baike.so.com/doc/_blank) 模型而得以轻量和高效，非常适合在分布式设备上运行数据密集型的实时应用。

# 二 需求分析

本章节主要是对音乐播放系统的功能性需求做出分析,并且多角度的分析系统的非功能性需求,在需求分析的最后部分,利用流程图来更加清楚的说明系统的需求。

2.1 系统需求概述

本文所设计的音乐播放系统是一个网上音乐在线播放系统。作为一个音乐播放系统,其需要对歌曲推荐界面管理，以及对面向用户的歌单界面的管理。

2.2 功能需求分析

### 2.2.1 用户管理模块需求分析

该部分主要是对音乐播放系统的用户登录模块进行具体的需求分析。用户登录管理模块的需求分析主要包括用户注册功能的需求分析,用户登录功能的需求分析等。下面是该模块中的每一个功能。如图2-1所示。

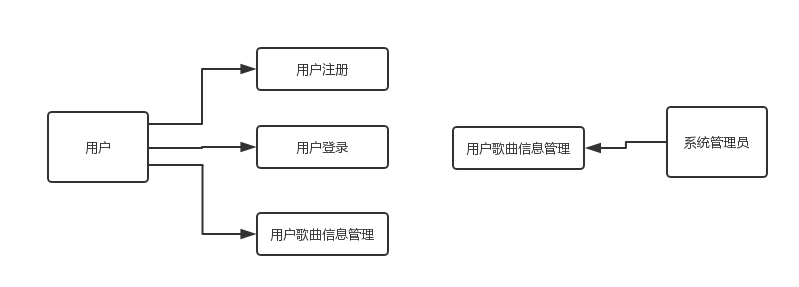


图2-1用户管理需求分析图

### 2.2.2歌曲推荐模块需求分析

在现在主流的一些国内的音乐播放网站,如网易云,QQ音乐等,都会有热门推荐，新品推荐，排行榜及与其相类似的功能模块,它们方便用户了解网站内的各类歌曲,了解大众的需求,帮助他们判断自己的需求。推荐模块如图2-2所示:

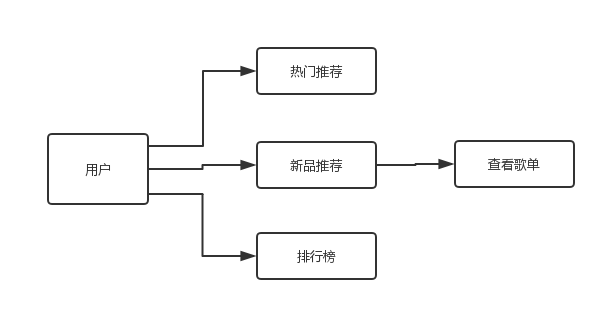


图2-2歌曲推荐需求分析图

### 2.2.3歌曲査询模块需求分析

如果用户是有目的性的查找某一首歌或者某一个MV,则用户会需要网站提供一个自主查询的功能。商品查询模块如图2-3所示。

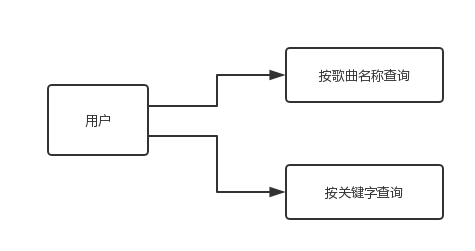


图2-3歌曲查询需求分析

三 系统架构设计

## 3.1系统体系架构设计

音乐播放系统采用的是MVC模式进行开发的。MVC模式将整个系统架构分为H层,模型层,视图层,控制层。同时使用stmts框架来清晰地区分业务逻辑和视图,从而简化开发的过程。

## 3.2 bootstrap框架介绍

Bootstrap是美国[Twitter](https://baike.so.com/doc/3861657-4054241.html" \t "https://baike.so.com/doc/_blank)公司的设计师Mark Otto和Jacob Thornton合作基于HTML、CSS、[JavaScript](https://baike.so.com/doc/5344215-5579659.html" \t "https://baike.so.com/doc/_blank) 开发的简洁、直观、强悍的[前端](https://baike.so.com/doc/6484003-6697709.html" \t "https://baike.so.com/doc/_blank)开发框架，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。Bootstrap一经推出后颇受欢迎，一直是[GitHub](https://baike.so.com/doc/5414728-5652870.html" \t "https://baike.so.com/doc/_blank)上的热门开源项目，包括NASA的MSNBC(微软全国广播公司)的Breaking News都使用了该项目。 国内一些移动开发者较为熟悉的框架，如[WeX5](https://baike.so.com/doc/24300723-25713213.html)前端开源框架等，也是基于Bootstrap源码进行性能优化而来。

## 3.3 vue.js框架介绍

Vue.js是一套构建用户界面的渐进式框架。与其他重量级框架不同的是，Vue 采用自底向上增量开发的设计。Vue 的核心库只关注视图层，并且非常容易学习，非常容易与其它库或已有项目整合。另一方面，Vue 完全有能力驱动采用单文件组件和Vue生态系统支持的库开发的复杂单页应用。

Vue.js 的目标是通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件 。

Vue.js 自身不是一个全能框架--它只聚焦于视图层。因此它非常容易学习，非常容易与其它库或已有项目整合。另一方面，在与相关工具和支持库一起使用时，Vue.js 也能完美地驱动复杂的单页应用。

## 3.4系统功能架构设计

本文设计的购物网站的主要功能架构设计如下图3-1所示：

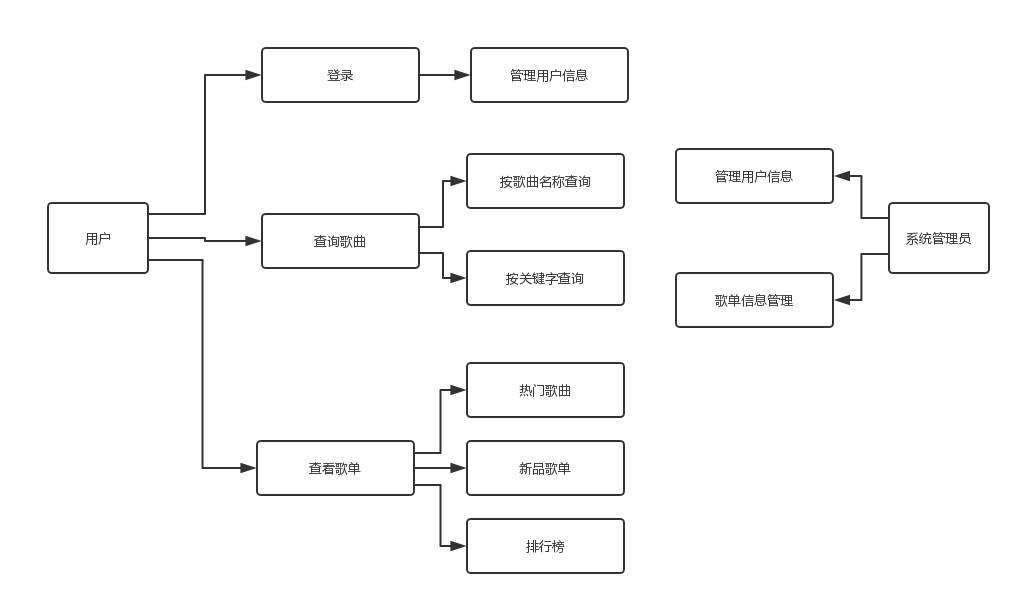


图3-1系统功能架构图

该购物网站主要分为用户和系统管理员两部分。系统管理又分为用户信息管理和歌单信息管理H大模块。用户管理分为用户登录管理，查询歌曲，查看歌单这几部分。用户登录后则可以查看自己收藏的歌曲，歌单信息或是修改信息，添加或删除歌曲，查询歌曲则分为按歌曲名或者按关键字搜索。歌单分为热门歌单，新品歌单，排行榜。用户可根据需求使用不同功能，且其他功能无需登录，可以更方便游客欣赏音乐。而系统管理也就是后台管理需要对用户更改的信息进行整理，对歌单推荐进行实时整理。

# 四 系统详细设计

## 4.1音乐播放系统的整体结构

为了更加准确的对音乐播放系统进行设计,我们有必要对网站的整体结构的先有所了解。该网站主要有几个核心的网页,分别为index.html,songlist.html,list.html,mv.html,login.js。

Index.html 内容为网站的主页面设计，内包含了header类，nav类等。

Songlist,list,mv等html为index中的内联框架iframe,从而实现仅在本页面加载iframe。

songlist内容为歌单内容。

list内容为歌单详细内容，内包含歌曲，mv等。

MV内容为歌曲的相关mv内容。

Login.js为登录提交的后台js文件。

## 4.2系统各功能模块的详细设计

## 4.2.1 banner图实现

Banner设计其实是我们接触互联网工作后，最常见最普及的一种工作。  
它就像主食一样，支撑着一个项目，一个公司，一个网站的运转。  
也是多数新产品，新事物，各种优惠活动呈现给用户和消费者的主要第一途径。

什么是Banner？

Banner的核心使命其实是吸引用户关注，然后被点击。所以它会是主题性明确，突出关键内容并有效抓住用户眼球的一种广告。  
通常穿插在页面中。当然在有效吸引用户的前提下，能跟不同页面融为一体是最好的境界。

本网页的banner大致如图4-1所示：



图4-1banner实图

## 4.2.2歌单动态加载实现

什么是API？

API就是操作系统留给应用程序的一个调用接口，应用程序通过调用操作系统的 API 而使操作系统去执行应用程序的命令。

由于歌曲获取资源问题，我使用了网易云提供的api，使用ajax向接口发送请求，返回我需要的歌单的相关数据且歌单的获取是打乱的，从而实现动态加载歌单。实图如图4-2所示：

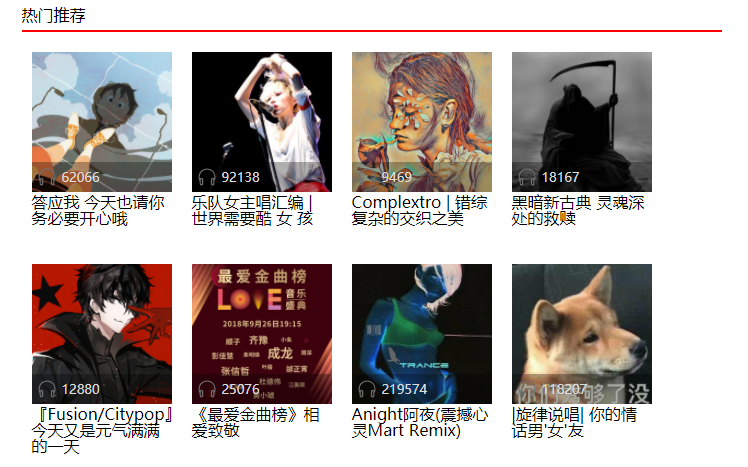


图4-2歌单实图

## 4.2.3歌单详细界面实现

歌单的详细列表同样使用了网易云提供的api，获取相应歌单内的歌曲详细列表。实图大致如图4-3所示：



图4-3歌单详细列表实图

## 4.2.4 音乐播放窗口实现

本网页的音乐播放窗口参考了网易云的音乐窗口，因考虑到用户播放音乐时，可能要浏览其他信息，所以当歌曲点击播放时不能干扰到其他内容，我选择将其固定到浏览器窗口底部，当用户点击播放歌曲时，player类div显示。实图大致如图4-4所示：

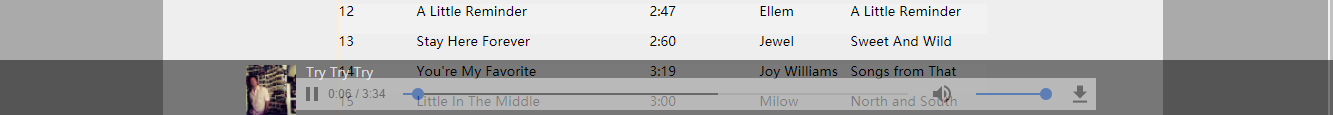


图4-4音乐播放窗口实现

## 4.2.5 MV播放界面设计

为了满足用户需求，本网页添加了mv播放界面，当用户点击歌曲对应列表中的mv时，iframe跳转至歌曲对应mv.实图如图4-5所示：



图4-5MV播放实图

## 4.2.6歌曲搜索实现

如果用户早已抱有目的的查找歌曲，则可以通过seacher入口按照歌曲名或关键字进行查询，查询结果包含对应歌曲信息及其mv信息。实图如图4-6所示。

## 4.2.7用户收藏及登录设计

每个歌单和每首歌曲都有一个收藏功能，用户可以通过登录后使用功能，登录后，用户可以对歌单，歌曲，mv进行收藏，收藏后可以在我的收藏中查看信息，以及对收藏的信息进行处理。这里使用的是MySql数据库，用户注册登录，通过mysql查询验证后，用户可以对该用户在数据库中的信息进行管理，包括对用户收藏的增，删，改，查等功能。

# 五 实现遇到的问题及解决方法

## 5.1查询问题

按关键字或歌曲名查询歌曲时，因为url中传递的参数为中文，所以传递时会出现乱码。

解决方案：传递参数前使用encodeURL（）进行参数编码，传递参数后使用decodeURL(decodeURL())对url参数进行双解码。

用例证明：当用户按关键字查询，如输入“海阔天空”关键字时，控制台url参数及查询结果如图4-6所示：

VI7PQ4%T)BUOMQ~[F`RJ4CI

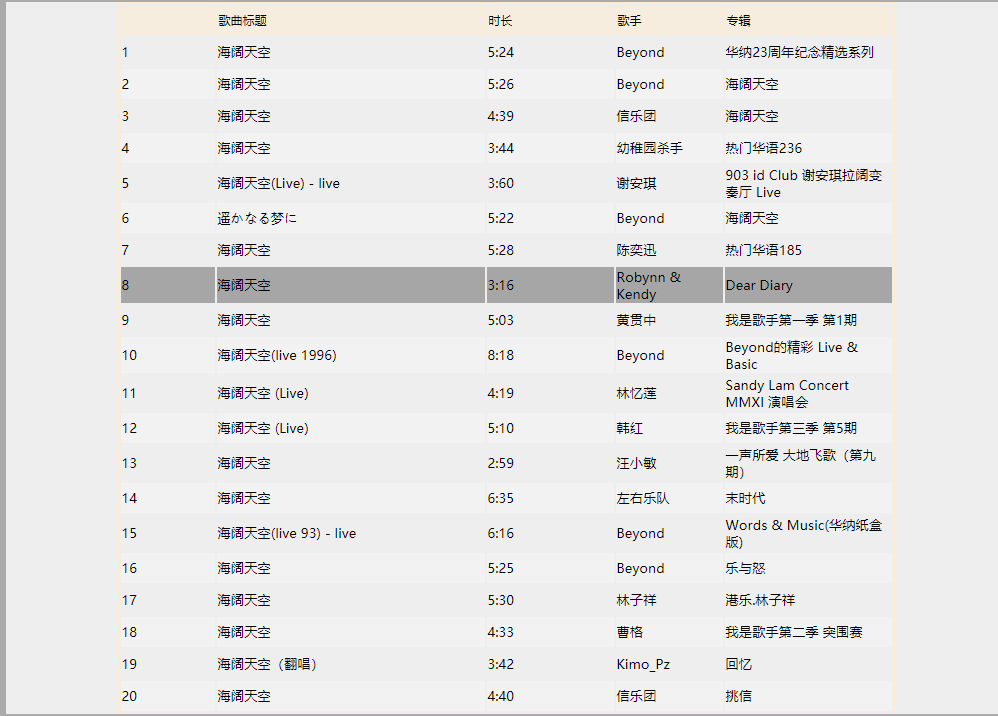


图4-6歌曲查询结果实图

# 结论

到此,毕业设计己经基本完成。在完成毕业设计的这段时间,我学到了很多。虽然之前学校给我们开设过许多的专业课程,包括java,数据库,数据结构等,也有一些相关的课程实验,但我们真正掌握的大多数都是偏理论的知识，在完整的设计开发音乐播放系统的过程中,由于知识掌握不足,出现过很多的问题,通过不停的网上查找资料,图书馆借阅相关书籍,和同学、指导老师讨论等多种途径,到此,该网站的开发终于基本完成。

# 致 谢

感谢学院给我提供了一个展现自己的平台，给了我一次煅炼机会，提高我的动手能力和并把知识付诸于实践。本设计的完成是在老师的细心指导下进行的，每次设计遇到问题时老师不辞辛苦的讲解才使得我的设计顺利的进行。从设计的选题到资料的搜集直至最后设计的修改的整个过程中，花费了老师很多的宝贵时间和精力，在此向老师表示衷心地感谢！老师严谨的治学态度，开拓进取的精神和高度的责任心都将使我受益终生！还要感谢我的几位同学，是你们在我平时设计中和我一起探讨问题，并指出我设计上的误区，使我能及时的发现问题把设计顺利的进行下去，没有你们的帮助我不可能这样顺利地结稿，在此表示深深的谢意。

感谢所有授我以业的老师，没有这些年知识的积淀，我没有这么大的动力和信心完成这篇设计。感恩之余，诚恳地请各位老师对我的设计多加批评指正，使我及时完善设计的不足之处。

# 参考文献

1. 唐晓莉.音乐类自媒体发展的新思路——以网易云音乐为例[J].新闻战线,2017(06):147-148.
2. 李鹏.Node.js及MongoDb应用研究[J].天津科技,2015(06):36-38+41.
3. 刘爱云,党军.浅析HTML5&CSS3的新特性[J].经贸实践,2015(08):358-359.
4. 旷志光,纪婷婷,吴小丽.基于Vue.js的后台单页应用管理系统的研究与实现[J].现代计算机(专业版),2017(30):53-57.
5. 兰旭辉,熊家军,邓刚.基于MySQL的应用程序设计[J].计算机工程与设计,2004(03):124-125+150.