

基于 Vue 的音乐平台设计与实现

黄风云 黄媛媛

四川大学锦城学院：计算机与软件学院 四川 成都 611731

【摘要】本文简述了一个基于 Vue 框架计划开发的音乐平台。基于“热爱生活，享受音乐”的理念，参考学习其他当下热门音乐平台的优点，摒弃他们的缺点，构建了一款操作简单，页面简洁的在线音乐平台。该项目采取了 Vue 框架来构建前端页面和实现功能。在本平台中，用户可以搜索音乐与歌手、播放音乐，查看相关的 MV 视频与相关的歌单信息，查看歌词与评论歌曲。

关键词：Vue；音乐平台；前端

1 引言

随着时代的发展，科技在随之飞速发展，人们的压力越来越大，很多人没有时间去放松心情，为了让人们能够在闲暇时间调整心情，让内心得到相应的放松，所以在空余时间听会音乐是一个不错的选择，人们可以从音乐中快速汲取能量。本音乐平台提供了搜索，音乐与视频播放，歌单管理等功能。本文将对选用的首要技术，可行性，用户需求以及功能模块几大部分归纳介绍本平台的实现流程。

2 主要技术选型介绍

2.1 Vue.js

Vue.js 是一套目前前端开发项目中一个重要的框架。和其他框架对比，Vue 采取自底向上增量开发的设计。而且其核心库只关注视图层，比较容易上手，也容易与其他库或已有的项目合并。Vue.js 还可以利用 MVVM 数据绑定和一个能够组合的组件系统，其具备简单、灵活的 API，因而可以经过简单的 API 实现响应式的数据绑定。我们也可以说 Vue.js 是一套响应式的系统，这会让状态管理变得非常简单且直观，可实现数据的双向绑定。

2.2 Element UI 组件库介绍

Element UI 是一个 UI 库，它不依赖于 Vue，但能配合 Vue 做项目开发，是一个比较好的框架。它提供了非常多的组件，能够帮助开发者快速构建页面。而且这个库使用较为简单，只需安装好相应环境，在官网寻找想要使用的样式导入即可。

3 可行性分析

3.1 开发模式可行性

本音乐平台主要采用了 Vue.js 和 Element UI。一个页面相当于一个单独的组件，不同的页面的样式与功能都写在对应的 Vue 页面中。这种单页面开发的模式能够节省用户的时间，对服务器请求较少，减少了服务器的压力，只需要第一次打开时加载，增强了用户的体验感。而且这种开发模式下代码复用度很大，后续进行代码的维护成本也会很低。

3.2 技术可行性

Vue 现在是前端开发中主流的三个框架之一，首先

它是一个轻量级的框架，而且相较于其他框架更加易于理解和学习，它也能进行双向数据绑定，页面组件化。Vue 更是操纵虚拟的 Dom，大大提升系统运行速率。

3.3 经济可行性

Vue 简单易学，花费时间少，降低了学习成本。开发出来的项目运行后直接能在谷歌浏览器的开发者工具栏中和 cmd 控制台中就能看见输出结果和纠正错误，大大降低开发成本。

4 需求分析

随着时代的进步，科技的发展，人们生活的压力也随之越来越大，而在细碎的时间片里能够让身心得到放松，音乐就成为了第一种选择。音乐本身的音质，旋律，编曲和抒情是各方面的艺术综合，是人类特有的体验，这种体验能影响生理感觉，当耳机里开始播放音乐，我们就沉浸在自己的世界里，不同的歌曲也能满足我们对不同精神世界的追求，所以音乐平台就随之孕育而生。本音乐平台的设计开发，参考了当下热门同类型平台的设计，将他们的优点采纳，让界面更加简洁美观，用户使用起来心情也会变得愉悦。同时将用户功能需求大抵分为：用户登录/注册，歌曲搜索，视频搜索，歌单搜索，歌单详情，歌曲评论等。

登陆/注册

满足用户登录属于自己本身账号的需求，账号里有专属的身份信息与歌曲，用户登录账号用于解锁更多平台功能。

搜索

满足用户搜索自己感兴趣的歌曲，视频与歌单的需求。

歌曲评论

满足用户对于不同的人对一首歌看法的需求，了解其他人的想法并且能够发表自己的想法，能够更好的使用户进行交流与沟通。

歌单详情

满足用户对相似歌曲的大量需求以及用户能通过歌单详情分析出是否是自己所喜欢的歌单。

评论点赞收藏

满足用户对自己认可的评论进行点赞与收藏的需求，同时也能激发用户进行歌曲评论的欲望。

每日推荐

满足用户希望能够让系统自动推送符合本身喜好的音乐的需求,同时满足增加音乐范围的需求。

5 系统设计与实现

5.1 功能模块拆分

本音乐平台功能模块主要分为用户管理,搜索管理,评论管理,歌单管理与歌曲管理等进行功能模块开发。致力于为用户打造一个美观简洁且高效的音乐平台。当用户未登录账号时,不能使用评论功能与歌曲收藏功能,其他所有的功能不会受到影响都能够照常使用。具体的模块设计如图1所示:

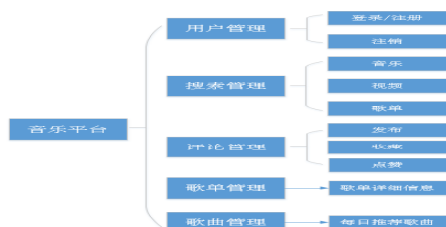


图1 功能模块设计

5.2 功能模块描述

5.2.1 用户管理

注册功能只针对新用户,新用户可以直接在页面注册账号并进入平台使用所有功能。用户使用现有的账号进行登录时,系统后台会进行账号密码的核对,登录成功就能使用当前用户账号,密码错误系统会对用户进行提示。歌曲的评论与点赞功能需要用户成功登录才能解锁使用。如果用户不想登录账号也可以使用搜索与播放的功能。

5.2.2 搜索管理

用户在搜索框输入自己想要搜索的内容后回车,系统会跳转到相应的页面并且获取输入框的内容,通过api接口地址取得与输入框相关的内容并且在页面显示相关内容的数量而且显示相关的搜索结果,用户能查看搜索到的音乐,视频,歌单,并且显示的音乐与视频都能播放。

5.2.2.1 具体实现

用户在全局页面头部的搜索框中输入想要搜索的内容并且按回车就会触发页面跳转的事件,系统会将搜索框里面的内容传递到另一个页面。另一个页面会在created的axios的params中接受到这个参数并且同时调用api,将想要搜索的内容与api接口里的数据进行对比并且将数据信息渲染到页面相应的位置。当用户点击想要播放的音乐,就会触发一个单击事件,这个事件同样会用axios调用一个接口地址,将想要播放音乐的id作为一个关键字去获取到其播放地址。然后通过this.\$parent将这个地址传递到父页面的底部audio的src属性中,并且audio拥有一个autoplay属性,这样就能实现搜索功能与单击歌曲播放音乐功能。

当用户分别点击音乐,视频还有歌单选项卡时,系统会在watch下使用switch判断其name属性,这样系统就知道用户点击到了哪一个部分,而且音乐,视频,歌单中都有一个type属性并且相互都不同,接着系统使用一个if对type进行条件判断,一个type对应一个部

分,在对应的if下获取到相应的数据,系统将这些数据渲染到对应的el-tab-pane组件里去就能实现搜索音乐,视频,歌单三个部分的切换。当用户点击一个视频,系统会跳转播放视频的页面并且传递一个id,这个页面同样获取到传递的id后调用视频的api接口,将视频的url添加到这个页面中video的src属性中去。用户点击想要了解的歌单,也同样是传递歌单的id到一个新的页面,然后新的页面通过这个id去调用api接口获取到这个id对应的歌单信息,将歌单信息渲染到歌单的页面上,然后将这些歌单中的歌曲添加一个播放事件,与之前的播放事件相对应就又可以播放歌曲,这样就实现了歌曲的详情页。

部分关键页面跳转传递参数与播放音乐的代码如下:

```
toMV(id) {  
  this.$router.push(`/MV?s=${id}`);  
},  
toPlayList(id) {  
  this.$router.push(`/PlayList?s=${id}`);  
},  
playMusic(id) {  
  axios({  
    url: <https://autumnfish.cn/song/  
url>,<br>  
    method: <get>,<br>  
    params: {<br>      id: id,<br>    },<br>  
  }).then((res) => {<br>    let url = res.data.data[0].url;<br>    this.$parent.musicUrl = url;<br>  });<br>},</pre>
```

5.2.3 评论管理

用户双击歌曲能进入歌曲的详情页,系统会调用相应api接口获取歌曲相关的评论显示在页面底下,用户能够对这些评论进行收藏点赞,并且自己也能够发布对这首歌的评论。

5.2.4 歌单管理

用户点击歌单,系统会进入到相应的页面并且显示歌单的创建者,创建歌单的时间日期与歌单包含的歌曲等相关信息,系统在用户点击歌单时获取歌单的id并且将id与相应接口地址里的id所匹配,将匹配到的信息显示在页面对应的位置。

5.2.5 歌曲管理

歌曲列表显示:打开歌曲页面,该页面打开时调用接口从后端获取歌曲信息(包括:ID、名称、作者、时长、歌曲对应图片地址等),后端获取消息,返回符合条件的歌曲信息,前端收到此信息,更新数据,界面根据绑定关系,自动更新页面歌曲列表。

6 结束语

本文简述了基于Vue.js框架开发一个音乐播放平台的实现过程,选取Vue.js和Element UI技术搭建了所有的页面并且实现了整体功能。经过测试,该网站已经

实现预定功能，是一个操作方便、用户体验良好的在线音乐平台。目前该音乐平台仍有需要改进完善的地方，在后续过程中将添加一些新的用户交互功能，让用户能够在音乐社区进行沟通与交流，而且后续还会将根据用户的使用情况不断进行优化升级与更新，致力于为广大用户提供一个简洁，高效音乐平台。

【参考文献】

- [1] 基于 web 的音乐播放器的设计与实现 [J]. 邓海文. 电脑知识与技术. 2018, 14(29)
- [2] 基于 Vue. js 的开发平台的设计与实现 [J]. 王志任. 广东工业大学. 2018