**重庆理工大学毕业设计（论文）**

文 献 综 述

学 院 计算机科学与工程

班 级 115030702

学生姓名 苏业纯 学 号 11503070233

**文献综述要求**

1、文献综述是要求学生对所进行的课题搜集大量资料后综合分析而写出的一种论文。其特点“综”是要求对文献资料进行综合分析、归纳整理，使材料更加精练明确、更有逻辑层次；“述”就是要求对综合整理后的文献进行比较专门的、全面的、深入的、系统的论述。

2、文献综述中引用的中外文资料，内容须与课题或专业方向紧密相关，理工类不得少于10篇，非理工类不少于12篇。

3、文献综述不少于2000字，其所附注释、参考文献格式要求同正文。

**文献综述评阅**

**评阅要求：**应根据学校“文献综述要求”，对学生的文献综述内容的相关性、阅读数量以及综述的文字表述情况等作具体的评价。

**指导教师评语：**

（要求手写，封面为第一页，此页为第二页，不得改变格式。）

指导教师：

年 月 日

**仿网易云音乐在线播放器的设计与实现文献综述**

1. 引言

随着信息化时代的发展，互联网、计算机等行业日趋繁荣。面对生活的快节奏，工作的漫长时间，人们对休闲时间的安排也越来越重视。人们在闲暇时间里会听音乐来释放自己的情感。目前中国的网民已经超过9亿，而在人们的日常生活中，使用率最高的设备就是智能手机。随着智能手机的服务越来越多，体量也变得越来越大。

1. 项目开发背景
2. 人们对音乐悠闲越来越重视了

不管是小孩、学生，还是社会上各种行业的人，他们对自己的精神世界的丰富也越来越重视了，并成为他们生活中的一部分了。音乐是情感的衍生品，它代表着人们的内心感受。音乐播放器是人们经常使用的平台之一，通过音乐平台人们可以放 松心情，感受生活的喜怒哀乐，得到精神上的升华[1]，大人们会给小孩子听儿歌，增加孩子们对这个世界的美好憧憬和向往，锻炼他们的想象能力。而大人们呢，在下班或者闲暇时打开他们自己的手机，戴上耳机，听一首自己喜欢的歌，或许是高兴的、悲伤的、平淡的，他们的感情都得到了释放。可以说，现在这个时代，人们已经在开始追求精神世界的富足了。这大大推动了各式各样音乐播放软件的产生。

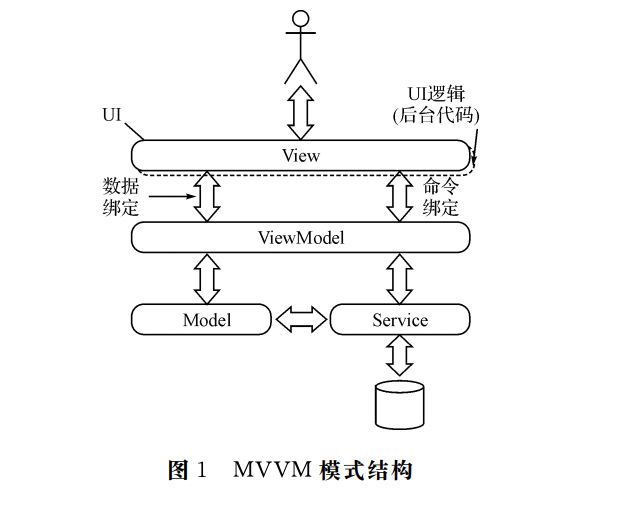
1. 音乐播放器APP的体积和大部分用户手机的硬件环境的冲突

近几年，随着科技的不断进步，各种高大上的设备或系统都一体化的出现在我们的视野中。这些高科技都伴随着信息的快速更新，所以很多服务提供商的服务也变得越发的花哨，他们不管用户到底喜欢与否，就为用户增加了很多不必要的功能，他们将APP做得越来越大。然而大部分用户的手机却没有升级，他们的手机根本无法承载这么大的APP，因此他们无法获得快速的体验良好的服务。此时，Web App的出现解决了这个难题。Web App可以实现跨平台的功能，可以无需考虑用户的手机系统到底是安卓还是iOS，并且，Web App非常的轻便，不会占用用户大量的手机内存空间，也是因为这个原因，可以提高用户手机的运行速度，提高了用户的使用体验。

1. 开发人员的开发和维护成本过高

现在的大部分软件服务提供商，都采取开发移动APP的方式为用户提供服务。手机的操作系统目前主要分为安卓和iOS，所以开发人员不得不开发两套完整的产品。但是安卓开发和iOS原生开发需要投入大量的人力，并且需要长时间的开发周期，而且后期的维护也要分为安卓和苹果两个类别，显得繁琐而且效率低下。此时，采用Web的开发方式就能解决这些问题。Web App可以实现跨平台，开发一套产品就能适应于所有手机操作系统，并且体积小，轻便，拥有良好的扩展性和维护性。现在很多主流的产品都开始转向采用Web的方式进行开发了，例如国内的WPS、国外的GitHub客户端等等，他们都是利用Web技术开发的，可见Web开发已经成为一种行业趋势。

1. 开发技术
   1. MVVM设计模式

随着程序功能的拓展与扩张，代码量也会相应地增加。程序也将会变得十分复杂。设计模式可以帮助解决用户界面不断变更的难题，并且能够使代码层次分明，易于阅读，条理清晰，并且便于后期 维护和改进。另一方面，如果开发人员更换，新的开发人员可以以更高的效率理解代码并参与后续的开发和维护。[2]那么什么是MVVM设计模式呢？

MVVM即 Model-View-ViewModel，即“模型-视图-视图模型”，是微软WPF和Silverlisht应用特有的一种界面设计模式。MVVM模式的核心，它是连接view和model的桥梁。它有两个方向(如图一)：一是将[模型]转化成[视图]，即将后端传递的数据转化成所看到的页面。实现的方式是：数据绑定。二是将[视图]转化成[模型]，即将所看到的页面转化成后端的数据。实现的方式是：DOM 事件监听。这两个方向都实现的，我们称之为数据的双向绑定。在MVVM的框架下视图和模型是不能直接通信的。它们通过ViewModel来通信，ViewModel通常要实现一个observer观察者，当数据发生变化，ViewModel能够监听到数据的这种变化，然后通知到对应的视图做自动更新，而当用户操作视图，ViewModel也能监听到视图的变化，然后通知数据做改动，这实际上就实现了数据的双向绑定。并且MVVM中的View 和 ViewModel可以互相通信。[12]

使用MVVM设计模式可以帮助我们分离业务逻辑，也就是实现前后端分离，显示逻辑和用户界面，使得我们的程序代码结构清晰，容易被阅读、测试、维护、替换、 扩展和改进。

* 1. 框架开发技术
     1. Vue框架

Vue是一套基于MVVM设计模式的Web框架，对比目前比较流行的 React、Angular、Ploymer 框架，与其他重量级框架不同的是，Vue 是一套构建用户界面的渐进式框架，采用自底向上增量开发的设计方式，是更加灵活、开放的解决方案，架构更加简单，适合开发人员快速掌握其全部特性并投入使用，还便于与第三方库或既有项目整合。结合 Vue生态系统支持库 Vuex、Vue-router，能够为复杂的应用程序提供驱动[4]。

Vue框架的核心思想是：数据驱动视图的变化。把一个普通 JavaScript对象传给 Vue实例的data选项，同时每个Vue 实例都有相应的 watcher 实例对象。如果 data的属性发生变化，会通知 watcher 重新计算，从而致使它关联的组件得以更新。Vue异步执行 DOM 更新。只要观察到数据变化，Vue 将开启一个队列，并缓冲在同一事件循环中发生的所有数据改变。如果同一个 watcher 被多次触发，就会依次推入到任务队列中，然后在缓冲去除了重复数据，避免了不必要的计算和 DOM 操作。Vue的响应为双向绑定数据，实时反映数据的真实变化，并映射到数据源上，避免了前端页面开发中 DOM 选择器繁杂的操作，简化了 Web 前端开发流程，降低了开发难度，提高了前端开发效率，缩短了开发成本和周期。

同时，Vue框架还是前后端分离的框架，前后端开发人员各司其职，提高前后端开发人员的效率。

* + 1. Spring Boot框架

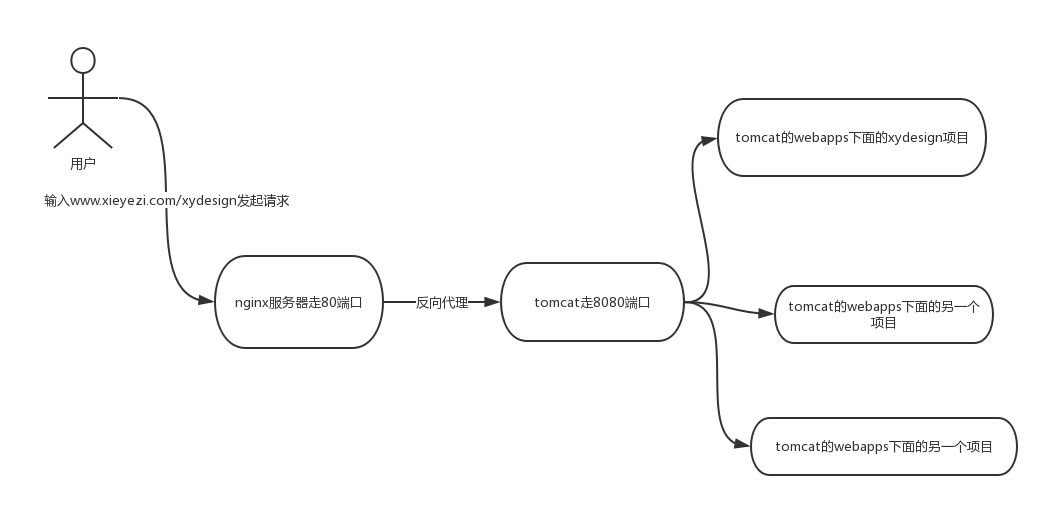
Spring Boot是由Pivotal团队提供的全新框架，其设计目的是用来简化新Spring应用的初始搭建以及开发过程。该框架使用了特定的方式来进行配置，从而使开发人员不再需要定义样板化的配置。

Spring Boot的核心功能有：

1. 独立运行spring项目。springboot可以以jar包的形式进行独立运行，使用java -jar xx.jar 就可以成功运行项目。
2. 内嵌servlet容器。内嵌容器，使得我们可以执行运行项目的主程序main函数，使得项目可以快速运行。
3. 提供starter简化Maven配置。springboot提供了一系列的start pol用来简化maven依赖。如：常用的spring-boot-starter-web、spring-boot-starter-tomcat、spring-boot-starter-actuator等
4. 自动配置spring。springboot会根据我们项目中类路径的jar包，为jar包的类进行自动装配bean。
5. 应用监控。springboot提供了基于HTTP、ssh、telnet对运行时的项目进行监控。springboot提供了actuator组件，只需要在配置中加入spring-boot-starter-actuator依赖，通过继承AbstractHealthIndicator这个抽象类，然后在doHealthCheck()方法中检测服务健康的方法，就可以实现一个简单的监控。
   1. Nginx实现反向代理
6. 什么是反向代理

反向代理（Reverse Proxy）方式是指以代理服务器来接受Internet上的连接请求，然后将请求转发给内部网络上的服务器；并将从服务器上得到的结果返回给Internet上请求连接的客户端，此时代理服务器对外就表现为一个服务器[10]。

1. 反向代理服务器的工作原理

反向代理服务器通常有两种模型，它可以作为内容服务器的替身，也可以作为内容服务器集群的负载均衡器。如下图：

客户端访问[www.xieyezi,com/xydesign](http://www.xieyezi,com/xydesign)这个网站，会先向Nginx服务器请求，Nginx服务器收到请求以后，根据反向代理规则，会找到目标网站所在的服务器和相应的端口，再将资源返回给客户端。

3.实现反向代理

Nginx 反向代理的指令不需要新增额外的模块，默认自带 proxy\_pass 指令，只需要修改配置文件就可以实现反向代理。只需要添加一个server，指向目标资源的域名和端口号即可。

* 1. JDBC
     1. JDBC连接数据库

注册数据库驱动:DriverManager.registerDriver(new Driver());

构建数据库连接URL:

DriverManager.getConnection("jdbc:mysql://localhost:3306/test", "root", "825825")【10】;

* + 1. **Mysql数据**库的基本操作

建表、增、删、改、查、主键、索引设置等操作。

* + 1. 数据库的特点。

数据库结构化：所谓整体的结构化是指数据库中的数据不在仅仅是针对某个应用，而是面向整个组织或企业；不仅数据库内部是结构化的，而且整体是结构化的，数据库之间是具有联系的。

数据库的共享性、冗余度低且易扩充：数据库系统从整体角度看待数据库和描述数据，数据不在面向某个应用而是面向整个系统。

数据独立性高：物理独立性和逻辑独立性。

数据由数据库管理系统统一管理和控制：数据库的安全性保护、完整性检查、并发控制、数据库恢复[8]。

* 1. 网络安全技术
     1. HTTP挟持

什么是HTTP劫持呢，大多数情况是运营商HTTP劫持，当我们使用HTTP请求请求一个网站页面的时候，网络运营商会在正常的数据流中插入精心设计的网络数据报文，让客户端（通常是浏览器）展示“错误”的数据，通常是一些弹窗，宣传性广告或者直接显示某网站的内容。

原理：在客户端与目的[服务器](https://www.baidu.com/s?wd=%E6%9C%8D%E5%8A%A1%E5%99%A8&tn=24004469_oem_dg&rsv_dl=gh_pl_sl_csd)所建立的专门的数据传输通道中，监视特定数据信息，如果满足条件，就会把客户端的http请求截断了，然后解包，植入一段js脚本，目的是让客户端程序解释“错误”的数据，进而展示宣传性内容。

解决方法：对付这种劫持最好的办法就是HTTPS,HTTPS能够加密网页传输内容，不仅防止可以强插广告，还可以防盗号。它可以防止窃听，还可以防止他人修改传输内容。还有一种方式就是向ISP运营商投诉。

参考文献

1. 邓海文. 基于web的音乐播放器的设计与实现[J].电脑知识与技术.2018(29)
2. 李嘉.赵凯强.李长云. Web前端开发技术的演化与MVVM设计模式研究[J]. 电脑知识与技术,2018.02
3. 李响. Web前端研究及优化[J]. 科技展望,2016,26(24):15.
4. 朱二华. 基于Vue.js的Web前端应用研究[J].科技与创新.2017(20)
5. 易剑波. 基于MVVM模式的WEB前端框架的研究[J].信息与电脑(理论版).2016(19)
6. 刘云浩.李沛. 基于Spring Boot的后台服务器开发[J].江西财经大学.中国科技信息.2018.17
7. 张丽.基于java语言访问的MySQL数据库[J]. 电子技术与软件工程，2016,（22）：165-166.
8. 王珊,萨师煊. 数据库系统概论[J]. 高等教育出版社. 2014.9
9. 乔亮,肖明华,李琳. 计算机网络工程安全问题及其对策[J]. 科技展望.2014(22)
10. 郝淑惠. 基于Nginx的Web服务器负载均衡策略改进与实现[J].电子技术与软件工程. 2019(02)
11. Model-View-ViewModel (MVVM) Design Pattern using Windows Presentation Foundation (WPF) Technology. The Maersk Mc-Kinney Moller Institute Campusvej 55, DK-5230 Odense M, Denmark
12. Spring Boot Reference Guide. Phillip Webb , Dave Syer , Josh Long , Stéphane Nicoll , Rob Winch , Andy Wilkinson , Marcel Overdijk , Christian Dupuis , Sébastien Deleuze , Michael Simons 1.5.4.RELEASE.
13. Developing a Web Application on NodeJS and MongoDB using ES6 and Beyond. Metropolia University of Applied Sciences Bachelor of Engineering Information Technology Bachelor’s Thesis. 25 January 2019