豆瓣

项目简介

豆瓣是一个高品质社区，核心用户群是具有良好教育背景的都市青年，包括白领及大学生。他们热爱生活，除了阅读、看电影、听音乐，更活跃于[豆瓣小组](https://baike.baidu.com/item/%E8%B1%86%E7%93%A3%E5%B0%8F%E7%BB%84" \t "https://baike.baidu.com/item/%E8%B1%86%E7%93%A3%E7%BD%91/_blank)、小站，对吃、穿、住、用、行等进行热烈的讨论，本项目即是基于豆瓣社区提供的信息进行进一步整合筛选，压缩豆瓣社区的冗余信息，整合并提供电影/音乐/书籍/图片等信息的分类显示，将优质资源集合在手机/移动客户端进行展示，相比较于豆瓣官方app，推送的内容信息更加纯净，适合电影/音乐/书籍/图片爱好者使用。

功能介绍

本项目中的功能主要有以下几种

1. 电影

本区域主要展示与电影相关的信息，包括电影海报缩略图，电影名称，分类，发行时间，豆瓣评分等，用户可以通过此板块了解电影相关的信息和动态，从而可以在短时间内筛选出自己需要的电影，内容分为以下三个区域：

* 1. Top250 展示当前评分在250名以内的优质电影资源。
  2. 正在热映 展示近期上映的好评电影。
  3. 即将上映 展示近期即将上映的电影资源。

1. 音乐

依托豆瓣的fm等社区功能推荐筛选出优质的音乐资源并整合至不同分类，供使用者快速遴选自己喜欢的音乐。主要板块如下

* 1. 飙升榜
  2. 新歌榜
  3. 原创榜
  4. Ktv麦榜
  5. 华语金曲榜
  6. 随便听听/个性推荐

1. 书籍

依托豆瓣读书和豆瓣阅读的高质量的书籍服务，从中选择优质书籍资源在该版块展示和轮播。用户可以在此板块快速获得需要的图书资料。

1. 图片

该功能拟接入豆瓣社区，并筛选豆瓣中的优质图片资源并进行分类，并以相册的形式进行展示。当前功能具有平面展示和大图详情等。

技术概览

App在实现过程中涉及/使用了以下几种语言/技术

* + 1. HTML
    2. CSS
    3. JavaScript
    4. Jquery
    5. Vue.js

简单的讲，其中html，超文本标记语言，超级文本标记语言是[标准通用标记语言](https://baike.baidu.com/item/%E6%A0%87%E5%87%86%E9%80%9A%E7%94%A8%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80" \t "https://baike.baidu.com/item/HTML/_blank)下的一个应用，也是一种规范，一种[标准](https://baike.baidu.com/item/%E6%A0%87%E5%87%86" \t "https://baike.baidu.com/item/HTML/_blank)，它通过标记符号来标记要显示的网页中的各个部分，网页[文件](https://baike.baidu.com/item/%E6%96%87%E4%BB%B6" \t "https://baike.baidu.com/item/HTML/_blank)本身是一种文本文件，通过在文本文件中添加标记符，可以告诉[浏览器](https://baike.baidu.com/item/%E6%B5%8F%E8%A7%88%E5%99%A8" \t "https://baike.baidu.com/item/HTML/_blank)如何显示其中的内容，主要负责搭建项目的结构，将所需要的内容信息一一展现在每一个板块中；

css，层叠样式表，(英文全称：Cascading Style Sheets)是一种用来表现[HTML](https://baike.baidu.com/item/HTML" \t "https://baike.baidu.com/item/CSS/_blank)（[标准通用标记语言](https://baike.baidu.com/item/%E6%A0%87%E5%87%86%E9%80%9A%E7%94%A8%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80" \t "https://baike.baidu.com/item/CSS/_blank)的一个应用）或[XML](https://baike.baidu.com/item/XML" \t "https://baike.baidu.com/item/CSS/_blank)（标准通用标记语言的一个子集）等文件样式的计算机语言，静态的修饰网页，本项目中主要将内容信息进行像素级布局和渲染美化，使界面更美观和简洁便于阅读；

javascript，一种直译式脚本语言，是一种动态类型、弱类型、基于原型的语言，内置支持类型。它的解释器被称为JavaScript引擎，为浏览器的一部分，广泛用于客户端的脚本语言，在本项目中则是实现界面和用户进行交互的技术，通过该脚本语言，用户可以实现本项目预设的功能与需求；

Jquery，一个快速、简洁的JavaScript代码库（或JavaScript框架），设计的宗旨是“write Less，Do More”，它封装JavaScript常用的功能代码，提供一种简便的JavaScript设计模式，优化HTML文档操作、事件处理、动画设计和Ajax交互。核心特性为：具有独特的链式语法和短小清晰的多功能接口；具有高效灵活的css选择器，并且可对CSS选择器进行扩展；拥有便捷的插件扩展机制和丰富的插件，通过该类库，可以快捷/简洁/高效的实现项目功能，同时可以缩短项目工期；

vue.js，一个构建数据驱动的 web 界面的渐进式框架，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，它采用自底向上增量开发的设计，核心库只关注视图层，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。通过该框架，本项目可以实现前端界面与后台数据库之间的完美通信，并且可以实现在用户端的渲染，高效简洁。

工作环境搭建：

因为本项目使用到了vue和vue-cli等，所以需要安装node环境和webpack等，安装流程如下：

1. 安装node.js,从node.js官网下载并安装node，按照提示一直下一步即可。

安装完成之后，打开命令行工具(win+r，然后输入cmd)，输入 node -v，

如果出现相应的版本号，则说明安装成功。

2.安装webpack，打开命令行工具输入：npm install webpack -g，

安装完成之后输入 webpack -v，

如果出现相应的版本号，则说明安装成功。

3.安装vue-cli脚手架构建工具，打开命令行工具输入：npm install vue-cli -g，

安装完成之后输入 vue -V（注意这里是大写的“V”）

如果出现相应的版本号，则说明安装成功。

4.在硬盘上找一个文件夹放工程用的,cd 目录路径

5.安装vue脚手架输入：vue init webpack projectname ，

注意这里的“projectname” 是项目的名称可以说是随便的起名，

但是需要主要的是“不能用中文”。

注意：安装项目过程中，会出现下面的选项，

? Use ESLint to lint your code? (Y/n) n

---------------------是否启用eslint检测规则，这里个人建议选no

6.cd 命令进入创建的工程目录，首先cd projectname（这里是自己建工程的名字）；

7.安装项目依赖：npm install（时间比较长）

8.启动项目，输入：npm run dev。

项目起步（本项目命名为mineobj）

首先通过html和css进行布局，通过将界面的每个位置进行定位和固定元素占位，初步展现出界面全貌。



如上图所示，整体分为上中下三部分，所以在本项目中实施组件和路由进行搭建，然后再添加功能。

第一步：首先对环境生成的显示界面进行调整，在其基础上进行搭建，快捷高效。

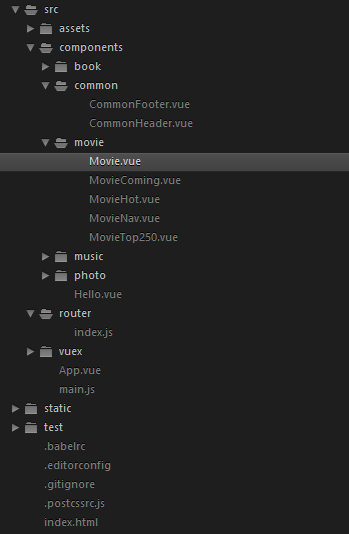
因为是适用于手机等移动端的app，所以在index.html这个主显示界面添加<meta name="viewport"

content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">，以适应不同移动设备的屏幕显示。

第二步，修改 mineobj\src\对应的app.vue的样式设置。删除（或修改router\index.js）对应的路由，将添加的组件修改为自己的组件movie。

此时环境搭建和基础修改已完成。接下来需要进一步完善功能。下面将主要以movie\top为例进行讲解。（本项目涉及了ECMAScript6的部分语法，如需要请自行补充）

搭建movie，需要在src\component文件夹下自行建立两个文件夹，common文件夹，用于存放公共的头部和脚部文件。 movie文件夹，用于存放movie本身和MovieComing，MovieHot，MovieNav，MovieTop250等文件。



整个文档那个结构是index.html 为整个加载渲染界面,main.js将App.vue 挂载到index.html 中。

<div id="app">

<router-view></router-view>//此处是路由占位。

</div>

Movie.vue 则通过路由文件index.js加载到App.vue的预留位置处。通过vue的渲染和动态加载，将数据动态的加载到界面中。值得注意的一点是，在每个组件中只允许有一个根目录，如下。

<template>

<div id="app">//唯一一个子目录

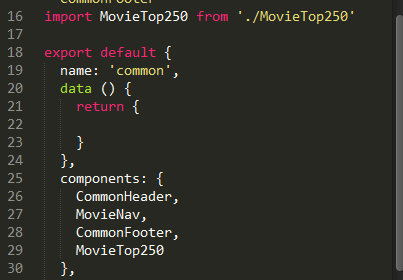
<router-view></router-view>

</div>

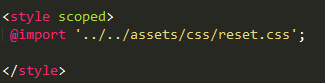
</template>

Movie.vue为主文件，作为其他子组件的加载区域模块使用。

ECMAScript6的部分语法 import 资源名称 from '资源路径' --->资源引入。要注意的是在组件中资源名称要保持一致，如下图



第16和29行资源名称保持一致。



Scoped表示此样式只在当前文件下有效。