

npm install style-loader stylus stylus-loader --save-dev

{  
 test: /\.css$/,  
 loader: 'style-loader!css-loader!stylus-loader',  
 include: [*resolve*('src'), *resolve*('test'), *resolve*('node\_modules/webpack-dev-server/client')]  
}

npm install fastclick babel-runtime babel-polyfill --save-dev

npm install jsonp --save-dev

npm install better-scroll --save-dev

npm install axios --save-dev

**最新的vue没有dev-server.js文件，如何进行后台数据模拟？**

最新的vue里dev-server.js被替换成了webpack-dev-conf.js

在模拟后台数据的时候直接在webpack-dev-conf.js文件中修改

第一步，在const portfinder = require(‘portfinder’)后添加

//第一步

const axios = require('axios');  
const express = require('express')  
const app = express()//请求server  
var apiRoutes = express.Router()  
app.use('/api', apiRoutes)//通过路由请求数据

第二步：找到devServer,在里面加上before（）方法

devServer: {

clientLogLevel: 'warning',

historyApiFallback: true,

hot: true,

compress: true,

host: HOST || config.dev.host,

port: PORT || config.dev.port,

open: config.dev.autoOpenBrowser,

overlay: config.dev.errorOverlay

? { warnings: false, errors: true }

: false,

publicPath: config.dev.assetsPublicPath,

proxy: config.dev.proxyTable,

quiet: true, // necessary for FriendlyErrorsPlugin

watchOptions: {

poll: config.dev.poll,

},

//第二步找到devServer,在里面添加

//通过axios代理请求

before(app) {  
 app.get('/api/getDiscList', (req, res) => {  
 const url = 'https://c.y.qq.com/splcloud/fcgi-bin/fcg\_get\_diss\_by\_tag.fcg';  
  
 axios.get(url, {  
 headers: {  
 referer: 'https://c.y.qq.com/',  
 host: 'c.y.qq.com'  
 },  
 params: req.query  
 }).then((response) => {  
 res.json(response.data);  
 }).catch((e) => {  
 console.log('axios err' + e);  
 });  
 });  
 }

请求报500错误 ，http请求头中含有 host referer ,需要反向代理设置

图片懒加载

npm install vue-lazyload --save-dev

npm install vuex --save-dev

store文件夹

index.js 入口文件

import Vue from 'vue';  
import Vuex from 'vuex';  
import \* as actions from './actions';  
import \* as getters from './getters';  
import state from './state';  
import mutations from './mutations';  
// 每次通过mutation修改state时打印日志  
import *createLogger* from 'vuex/dist/logger';  
  
Vue.use(Vuex);  
  
const debug = process.env.NODE\_ENV !== 'production';  
  
export default new Vuex.*Store*({  
 actions,  
 getters,  
 state,  
 mutations,  
 strict: debug,  
 plugins: debug ? [createLogger()] : []  
});

state.js 状态

const state = {  
 singer: {}  
};  
  
export default state;

getter.js 获取state 数据

export const *singer* = state => state.singer;

mutation-types.js mutation相关字符串常量

export const SET\_SINGER = 'SET\_SINGER';

mutations.js

import \* as types from './mutation-types';  
  
const mutations = {  
 [types.SET\_SINGER](state, singer) {  
 state.singer = singer;  
 }  
};  
  
export default mutations;

action.js 异步修改

singer.vue 设置数据

import {mapMutations} from 'vuex';

methods: {  
 selectSinger (singer) {  
 this.$router.push({  
 path: `/singer/${singer.id}`  
 });  
 this.setSinger(singer); // 实现了对mutation的提交

},  
 ...mapMutations({  
 setSinger: 'SET\_SINGER' // mutation的修改映射成方法  
 })  
},

singer-detail.vue 获取数据

import {mapGetters} from 'vuex';

computed: {  
 ...mapGetters([  
 'singer'  
 ])  
},

动画 Js创建 css3动画

npm install create-keyframe-animation --save-dev

歌词base64解码

npm install js-base64 --save-dev

歌词字符串解析

npm install lyric-parser

对sessionStorage 和 localStorage 封装，使用相同的api

npm install good-storage --save-dev

编译打包

编写服务器代码

prod.server.js

var express = require('express')  
var config = require('./config/index')  
var axios = require('axios')  
  
var port = process.env.PORT || config.build.port  
  
var app = express()  
  
app.get('/api/getDiscList', (req, res) => {  
 const url = 'https://c.y.qq.com/splcloud/fcgi-bin/fcg\_get\_diss\_by\_tag.fcg';  
  
 axios.get(url, {  
 headers: {  
 referer: 'https://c.y.qq.com/',  
 host: 'c.y.qq.com'  
 },  
 params: req.query  
 }).then((response) => {  
 res.json(response.data);  
 }).catch((e) => {  
 console.log('axios err' + e);  
 });  
});  
  
app.get('/api/lyric', (req, res) => {  
 const url = 'https://c.y.qq.com/lyric/fcgi-bin/fcg\_query\_lyric\_new.fcg';  
  
 axios.get(url, {  
 headers: {  
 referer: 'https://c.y.qq.com/',  
 host: 'c.y.qq.com'  
 },  
 params: req.query  
 }).then((response) => {  
 var ret = response.data;  
 if (typeof ret === 'string') {  
 var reg = /^\w+\(({[^()]+})\)$/;  
 var matches = ret.match(reg);  
 if (matches) {  
 ret = ***JSON***.parse(matches[1]);  
 }  
 }  
 res.json(ret);  
 }).catch((e) => {  
 console.log('axios err' + e);  
 });  
});  
  
app.get('/api/getSongVkey', (req, res) => {  
 const url = 'https://c.y.qq.com/base/fcgi-bin/fcg\_music\_express\_mobile3.fcg';  
  
 axios.get(url, {  
 headers: {  
 referer: 'https://c.y.qq.com/',  
 host: 'c.y.qq.com'  
 },  
 params: req.query  
 }).then((response) => {  
 res.json(response.data);  
 }).catch((e) => {  
 console.log('axios err' + e);  
 });  
});  
  
app.get('/api/getSongList', (req, res) => {  
 const url = 'https://c.y.qq.com/qzone/fcg-bin/fcg\_ucc\_getcdinfo\_byids\_cp.fcg';  
  
 axios.get(url, {  
 headers: {  
 referer: 'https://c.y.qq.com/'  
 },  
 params: req.query  
 }).then((response) => {  
 res.json(response.data);  
 }).catch((e) => {  
 console.log('axios err' + e);  
 });  
});  
  
app.use(express.static('./dist'))  
  
module.exports = app.listen(port, function (err) {  
 if (err) {  
 console.log(err)  
 return  
 }  
 console.log('Listening at http://localhost:' + port + '\n')  
})

npm build

node prod.server.js

按需加载组件，路由组件异步加载减小app.js的体积

const *Recommend* = () => import('@/components/recommend/recommend');  
const *Singer* = () => import('@/components/singer/singer');  
const *Rank* = () => import('@/components/rank/rank');  
const *Search* = () => import('@/components/search/search');  
const *SingerDetail* = () => import('@/components/singer-detail/singer-detail');  
const *Disc* = () => import('@/components/disc/disc');  
const *TopList* = () => import('@/components/top-list/top-list');  
const *UserCenter* = () => import('@/components/user-center/user-center');

npm build

node prod.server.js

移动端调试查看日志

npm install vconsole --save-dev

main.js

import Vconsole from 'vconsole';  
*/\* eslint-disable no-unused-vars \*/*let vconsole = new Vconsole();  
console.log('test');

 