

相关文章

- [手摸手，带你用vue撸后台 系列一\(基础篇\)](#)
- [手摸手，带你用vue撸后台 系列二\(登录权限篇\)](#)
- [手摸手，带你用vue撸后台 系列三\(实战篇\)](#)
- [手摸手，带你用vue撸后台 系列四\(vueAdmin 一个极简的后台基础模板\)](#)
- [手摸手，带你封装一个vue component](#)

Donate

If you find this project useful, you can buy me a cup of coffee

Wechat



Alipay



个人圈子

[前端圈](#)

详细使用文档

wiki只列出了几个常见问题，更多信息请参考 [使用文档](#)

windows运行须知

1.windows环境下node-sass下载安装不成功问题

- [参考1](#)
- [参考2](#)

2.NODE_ENV不是内部或外部命令 已使用cross-env解决兼容性问题

- [参考1](#)
- [参考2](#)

mock

本项目目前所有接口使用mock.js模拟，如使用后端api，可在main.js移除

```
import './mock/index.js'; // 使用api请求时请将此行删除
```

跨域问题

Access-Control-Allow-Origin 报错，还是怎么解决跨域问题，这类问题请自行google。跨域问题涉及到的东西比较多，而且大部分时后端问题，所有请自行解决。

babel-polyfill

本项目暂时没有兼容性需求，如有兼容性需求可自行使用babel-polyfill。在Node/Browserify/webpack中使用

```
npm install --save babel-polyfill //下载依赖
```

在入口文件中引入

```
import 'babel-polyfill';  
// 或者  
require('babel-polyfill');//es6
```

在webpack.config.js中加入babel-polyfill到你的入口数组：

```
module.exports = {  
  entry: ["babel-polyfill", "./app/js"]  
}
```

具体可参考 [link](#)

或者更简单暴力 [polyfill.io](#) 使用它给的一个cdn地址，引入这段js之后它会自动判断浏览器，加载缺少的那部分polyfill，但国内速度肯能不行，大家可以自己搭 cdn。 [更多相关内容](#)

background url()问题

<https://github.com/vuejs/vue-loader/issues/481>

<https://github.com/vuejs/vue-cli/issues/112>

资源路径相关问题

该项目目前使用的是history的路由模式，当部署到服务器上时，请自行根据自己需求调整。 目录 config/index.js [相关问题](#)

vue中使用第三方库

在诸多 Vue.js 应用中, Lodash, Moment, Axios, Async等都是些非常有用的 JavaScript 库. 但随着项目越来越复杂, 可能会采取组件化和模块化的方式来组织代码, 还可能要使应用支持不同环境下的服务端渲染. 除非你找到了一个简单而又健壮的方式来引入这些库供不同的组件和模块使用, 不然, 这些第三方库的管理会给你带来一些麻烦. 详情见该[blog](#).

富文本tinymce须知

改项目目前使用的富文本tinymce目前没有使用import方式加载，使用script方式引入。如需不要可在index.html页面移除tinymce的引入，并删除static目录下tinymce整个文件夹。

这里在简述一下推荐使用tinymce的原因：tinymce是一家老牌做富文本的公司(这里也推荐ckeditor，也是一家一直做富文本的公司，也不错)，它的产品经受了市场的认可，不管是bug还是配置的自由度都很好。在使用富文本的时候有一点也很关键就是复制格式化，之前在用一款韩国人做的富文本summernote被它的格式化坑的死去活来，但tinymce的去格式化相当的好，它还有一个增值项目就是powerpaste,那是无比的强大，支持从word里面复制各种东西，都不就有问题。富文本还有一点也很关键，就是拓展性。楼主用tinymce写了好几个插件，学习成本和容易度都不错，很方便拓展。最后一点就是文档很完善，基本你想得到的配置项，它都有。tinymce也支持按需加载，你可以通过它官方的build页定制自己需要的plugins。

vendor过大问题

建议使用gzip，使用之后体积会只有原先1/3左右。还可以使用懒加载或者Code Splitting 建议参考[这篇文章](#)。打出来的 app.js 过大，查看一下是不是Uglify配置不正确或者sourceMap没弄对。优化相关请看该[文章](#)

将项目打包后发布到apache的www下的vue子目录

见[文档](#)

Webstorm IDEA cannot support webpack alias

<https://github.com/umijs/umi/issues/1109>