VUE开发注意事项

**PS:本文档配置说明针对vue-cli 2.9.x版本**

**1.webpack配置注意事项**

1.1 build文件夹下 utils.js 修改

ExtractTextPlugin.extract选项增加publicPath: '../../'属性值（解决打包后图片路径404问题）;

1.2 build文件夹下build.js修改

搜索‘process.env.NODE\_ENV = 'production'’并注释掉（指定node运行环境所用）

1.3 build文件夹下webpack.prod.conf.js修改

变量env值改为 config.build[process.env.ENV\_CONFIG+'Env']

Ps：作用--->根据环境进行打包操作(指定BASE\_URL)；可与原有设置进行比较分析其作用。

1.4 config文件夹下 index.js修改

1.4.1 dev选项配置修改

host值改为0.0.0.0(解决局域网移动端无法扫码测试访问);

1.4.2 build选项配置修改

1.4.2.1 assetsPublicPath值改为’./’（解决打包资源路径问题）；productionSourceMap值改为‘process.env.NODE\_ENV === 'production' ? false : true’（根据运行环境决定是否打包map文件，测试则打包方便调试。正式打包去除map文件减小文件体积，）；productionGzip值改为true（压缩打包文件，需要后台服务容器配置:nginx-->gzip on;gzip\_static on;）此项安装插件npm install --save-dev compression-webpack-plugin;

1.4.2.2 增加三个字段，如下图所示（稍后1.5会解释对应字段的用途）



dev

1.5 不同环境参数配置(以ajax请求BASE\_URL为例)

上述中讲到需要添加三个字段，现对其进行说明：

devEnv字段代表开发环境（本地及测试dev.env.js文件）配置如下图：



proEnv字段代表生产环境（prod.env.js文件），配置如下：



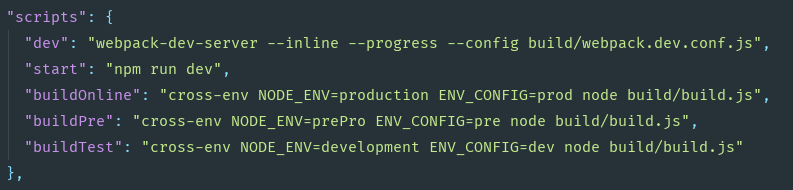
preEnv为我们手动创建的pre.env.js文件，此处为预发布环境配置：



**PS：NODE\_ENV为node运行环境； BASE\_URL为ajax请求基础url；ENV\_CONFIG为环境信息。（NODE\_ENV及ENV\_CONFIG仅为注释作用，无实际作用，起实际作用的为下属2中script打包脚本设置的值）**

1. **package.json文件修改**

把scripts启动脚本配置中的build改为多个，一个为测试打包，一个为生产环境打包，一个为预发布环境包，具体如下图：



其中 buildOnline为线上打包，buildTest为测试环境打包，buildPre为预发布环境打包（需要安装cross-env--->npm i cross-env -D）；运行方式 npm run buildTest

1. **ajax请求工具 axios的配置**

3.1 全局引用

安装依赖 npm install axios --save在main.js 引入import axios from 'axios'以及import qs from 'qs'

3.2 配置axios

// 处理返回数据 去除headers status等值只保留后台返回信息

axios.interceptors.response.use(

response => response.data

);

// 处理post请求时向后台传递数据格式问题

axios.defaults.transformRequest = [data => qs.stringify(data)];

// 设置请求头的格式

axios.defaults.headers = {

'Content-Type': 'application/x-www-form-urlencoded;charset=UTF-8',

'Access-Control-Allow-Credentials': true

};

// 设置请求链接的基础路径 --->根据运行环境确定baseUrl

axios.defaults.baseURL=process.env.BASE\_URL;

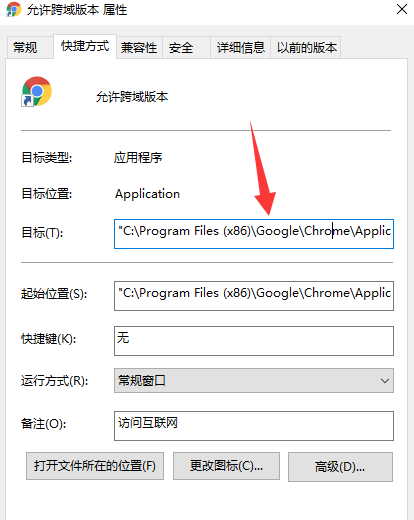
// 设置发送请 cookie认证

axios.defaults.withCredentials = true

//设置全局使用axios ---> this.$axios.post().....

Vue.prototype.$axios = axios

1. **Chrome浏览器解除跨域限制**



新复制一个Chrome快捷方式，然后鼠标右键属性打开如上图所示，在目标位置后面添加

--args --disable-web-security --user-data-dir （注意--args前面有一空格），然后保存退出，利用修改后的快捷方式打开chrome即可去除跨域问题。打开后在状态栏会有一警告，此警告即是跨域限制去除的标识：



PS:打开之前请关闭其他的所有chrome窗口