

# 一、项目的创建和基本配置

## 1、创建项目

将vue-admin-template-master重命名为guli-admin

## 2、修改项目信息

package.json

```
1 {  
2   "name": "guli-admin",  
3   .....  
4   "description": "谷粒学院后台管理系统",  
5   "author": "Helen <55317332@qq.com>",  
6   .....  
7 }
```

## 3、如果需要修改端口号

config/index.js中修改

```
1 port: 9528
```

## 4、项目的目录结构

```
1 .  
2 |— build // 构建脚本  
3 |— config // 全局配置  
4 |— node_modules // 项目依赖模块  
5 |— src //项目源代码  
6 |— static // 静态资源  
7 |— package.jspon // 项目信息和依赖配置
```

```
1  src
2  |─ api // 各种接口
3  |─ assets // 图片等资源
4  |─ components // 各种公共组件，非公共组件在各自view下维护
5  |─ icons //svg icon
6  |─ router // 路由表
7  |─ store // 存储
8  |─ styles // 各种样式
9  |─ utils // 公共工具，非公共工具，在各自view下维护
10 |─ views // 各种layout
11 |─ App.vue //***项目顶层组件***
12 |─ main.js //***项目入口文件***
13 |─ permission.js //认证入口
```

## 5、运行项目

```
1  npm run dev
```

## 二、登录页修改

src/views/login/index.vue ( 登录组件 )

4行

```
1  <h3 class="title">谷粒学院后台管理系统</h3>
```

28行

```
1  <el-button :loading="loading" type="primary" style="width:100%;"
2    @click.native.prevent="handleLogin">
3    登录
4  </el-button>
```

## 三、页面零星修改

### 1、标题

index.html ( 项目的html入口 )

```
1 <title>谷粒学院后台管理系统</title>
```

修改后热部署功能，浏览器自动刷新

### 2、国际化设置

打开 src/main.js ( 项目的js入口 )，第7行，修改语言为 **zh-CN**，使用中文语言环境，例如：日期时间组件

```
1 import locale from 'element-ui/lib/locale/lang/zh-CN' // lang i18n
```

### 3、icon

复制 favicon.ico 到根目录

### 4、导航栏文字

src/views/layout/components ( 当前项目的布局组件 )

src/views/layout/components/Navbar.vue

13行

```
1 <el-dropdown-item>
2   首页
3 </el-dropdown-item>
```

17行

```
1 <span style="display:block;" @click="logout">退出</span>
```

## 5、面包屑文字

src/components ( 可以在很多项目中复用的通用组件 )

src/components/Breadcrumb/index.vue

38行

```
1 meta: { title: '首页' }
```

## 四、Eslint语法规范型检查

### 1、ESLint简介

JavaScript 是一个动态的弱类型语言，在开发中比较容易出错。因为没有编译程序，为了寻找 JavaScript 代码错误通常需要在执行过程中不断调适。

ESLint 是一个语法规则和代码风格的检查工具，可以用来保证写出语法正确、风格统一的代码。让程序员在编码的过程中发现问题而不是在执行的过程中。

### 2、语法规则

ESLint 内置了一些规则，也可以在使用过程中自定义规则。

本项目的语法规则包括：两个字符缩进，必须使用单引号，不能使用双引号，语句后不可以写分号，代码段之间必须有一个空行等。

### 3、确认开启语法检查


打开 config/index.js，配置是否开启语法检查

```
1 useEslint: true,
```

可以关闭语法检查，建议开启，养成良好的编程习惯。

### 4、ESLint插件安装

vs code的ESLint插件，能帮助我们自动整理代码格式



ESLint

dbaeumer.vscode-eslint

Dirk Baeumer

14,539,012

★★★★★

存储库

许可证

Integrates ESLint JavaScript into VS Code.

重载进行更新

禁用

卸载

根据您最近打开的文件推荐此扩展。

Ignore Recommendation

详细信息

发布内容

更改日志

## 5、插件的扩展设置

选择vs code左下角的“设置”，打开 VSCode 配置文件,添加如下配置

用户设置 工作区设置 文件夹设置

常用设置

文本编辑器

工作台

窗口

功能

应用程序

扩展

Auto Rename Tag co...

CSS

CSS Peek extension c...

CSS Settings

Emmet

ESLint

Git

Grunt

Gulp

HTML

Htmltagwrap configu...

Jake

JSON

LESS

Live Server Config

Markdown

Node debug

Npm

ESLint

Eslint: **\_legacy Module Resolve**

Uses the legacy module resolving.

Eslint: **Always Show Status**

Always show the ESLint status bar item.

Eslint: **Auto Fix On Save**

Turns auto fix on save on or off.

Eslint: **Enable**

Controls whether eslint is enabled for JavaScript files or not.

Eslint: **Node Path**

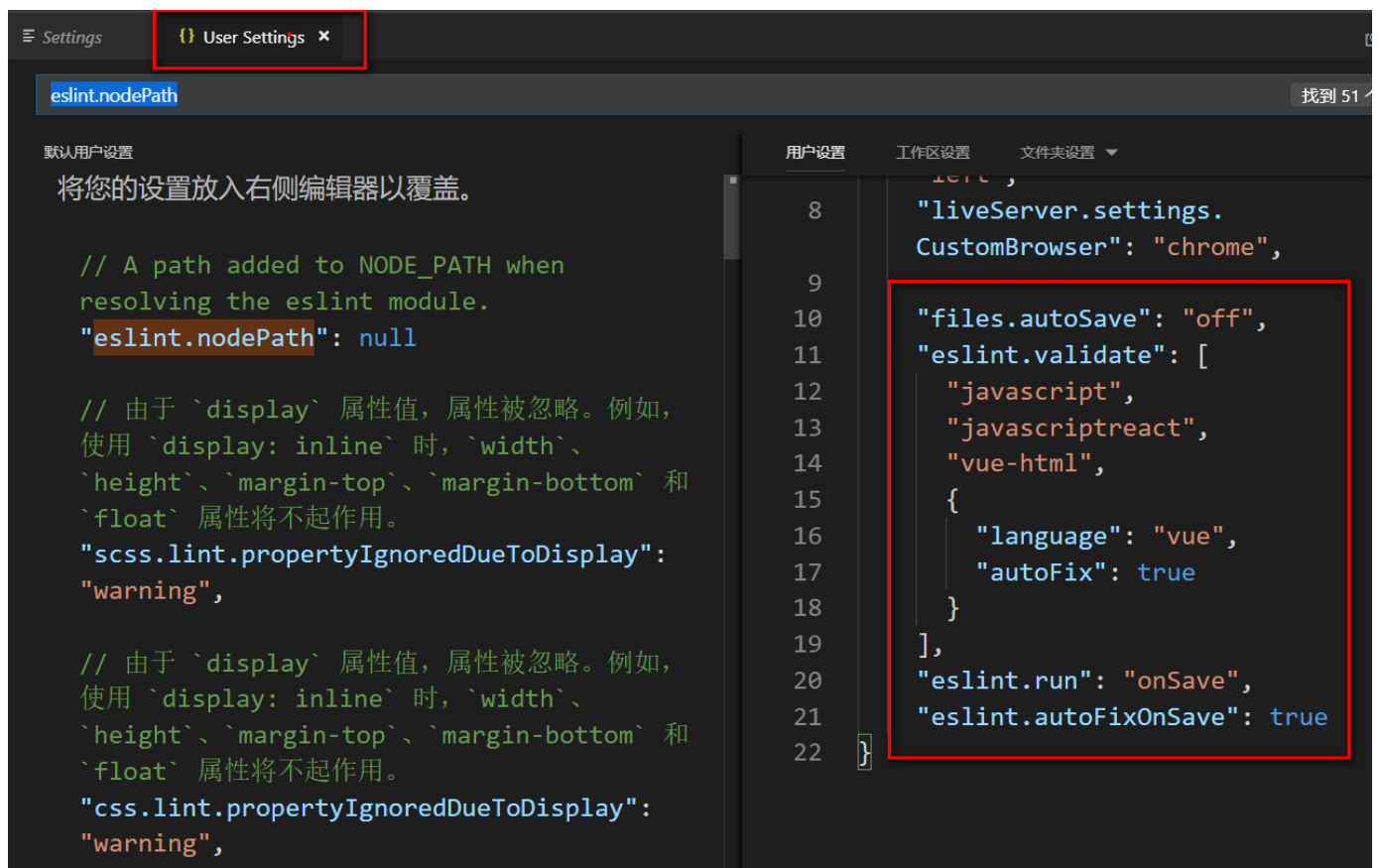
A path added to NODE\_PATH when resolving the eslint module.

在 settings.json 中编辑

Eslint: **Options**

The eslint options object to provide args normally passed to eslint when execu

http://eslint.org/docs/developer-guide/nodejs-api#cliengine).



```
1 "files.autoSave": "off",  
2 "eslint.validate": [  
3   "javascript",  
4   "javascriptreact",  
5   "vue-html",  
6   {  
7     "language": "vue",  
8     "autoFix": true  
9   }  
10 ],  
11 "eslint.run": "onSave",  
12 "eslint.autoFixOnSave": true
```