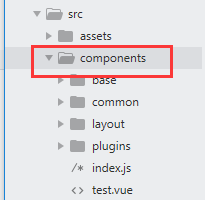
## 项目结构说明：

<https://github.com/moment3784/vue-temp-design>

项目用了vue-cli webpack 进行构建,使用前请阅读vue-cli 构建项目的一些教程。

## Src/components/ 说明



Components 组件

Base ===》 定义为基础组件，具有单一功能，比如 按钮 弹窗之类的。

Common ===》 定义为公用组件，复合组件，往往是多个base组件组合起来的。解决一个小模块的功能的，比如轮播图，图片查看，预览等

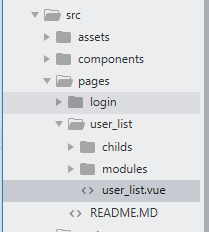
Layout ===》 项目外框架，一个比较大的项目结构定义

Plugins ===》 一些js公共插件，比如 vue 的自定义事件

Index.js ===》 对以上组件进行构建跟管理

Test.vue ===》 组件测试用的

## src/pages/



Login 跟User\_list 都是一个比较大的页面

User\_list ===》 主要内容的页面

/modules ===》 页面划分模块写

/childs ===》 跳转到子页面

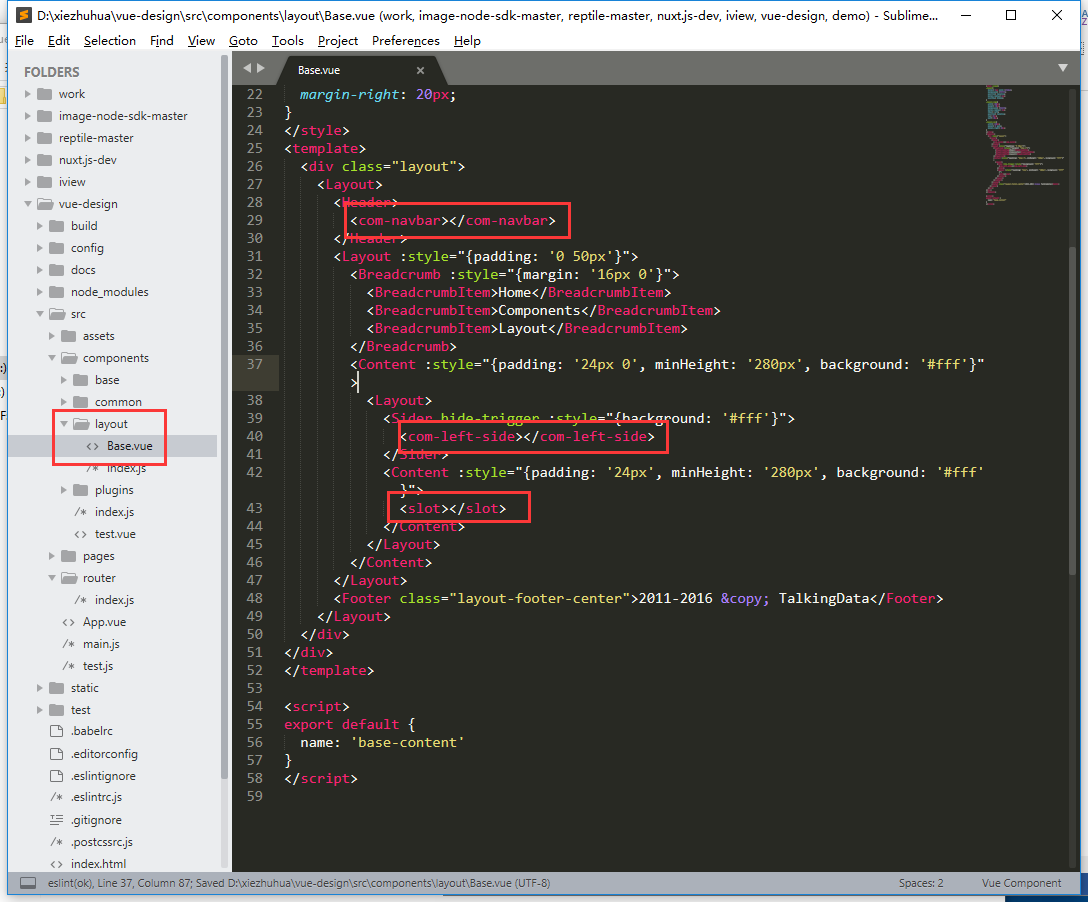
## 项目结构详细说明

### Components

Base 暂由 iview组件代替

#### Layout/base

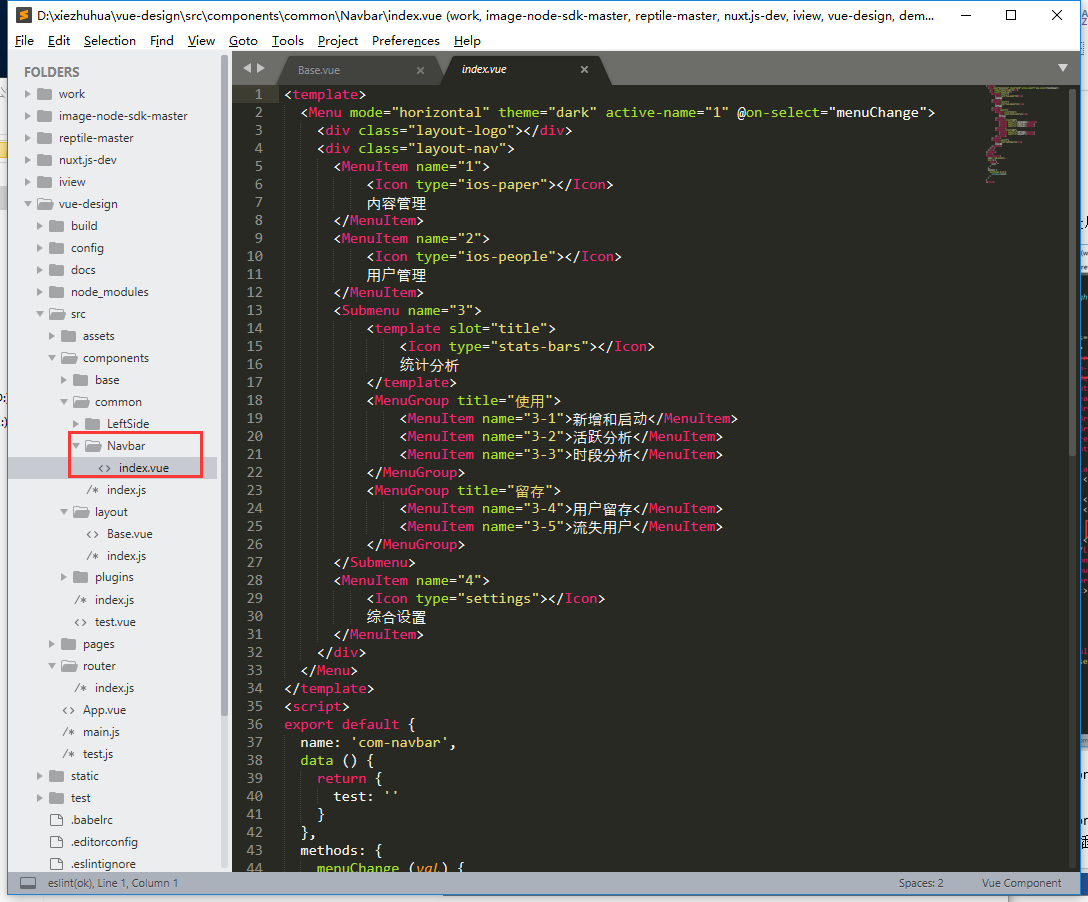
目标在于建立基本的外框！既是后台管理页面常见的菜单栏，左栏，右栏主内容



#### <com-navbar> 公共组件

Common/navbar

目前数据内容固定写死

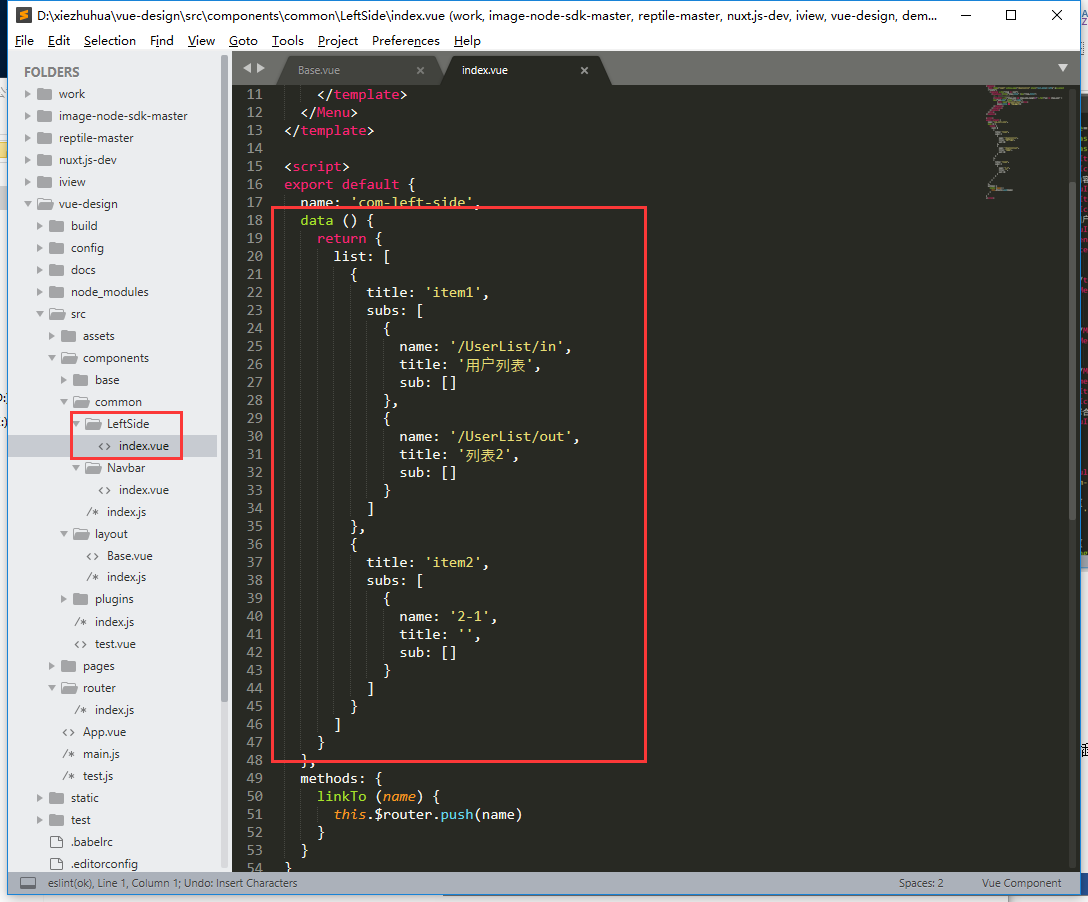


#### <com-left-side>

Common/leftside

左栏菜单

Data 里面填写的是模拟数据，这部分可设计成由父组件传参组建



#### <slot>

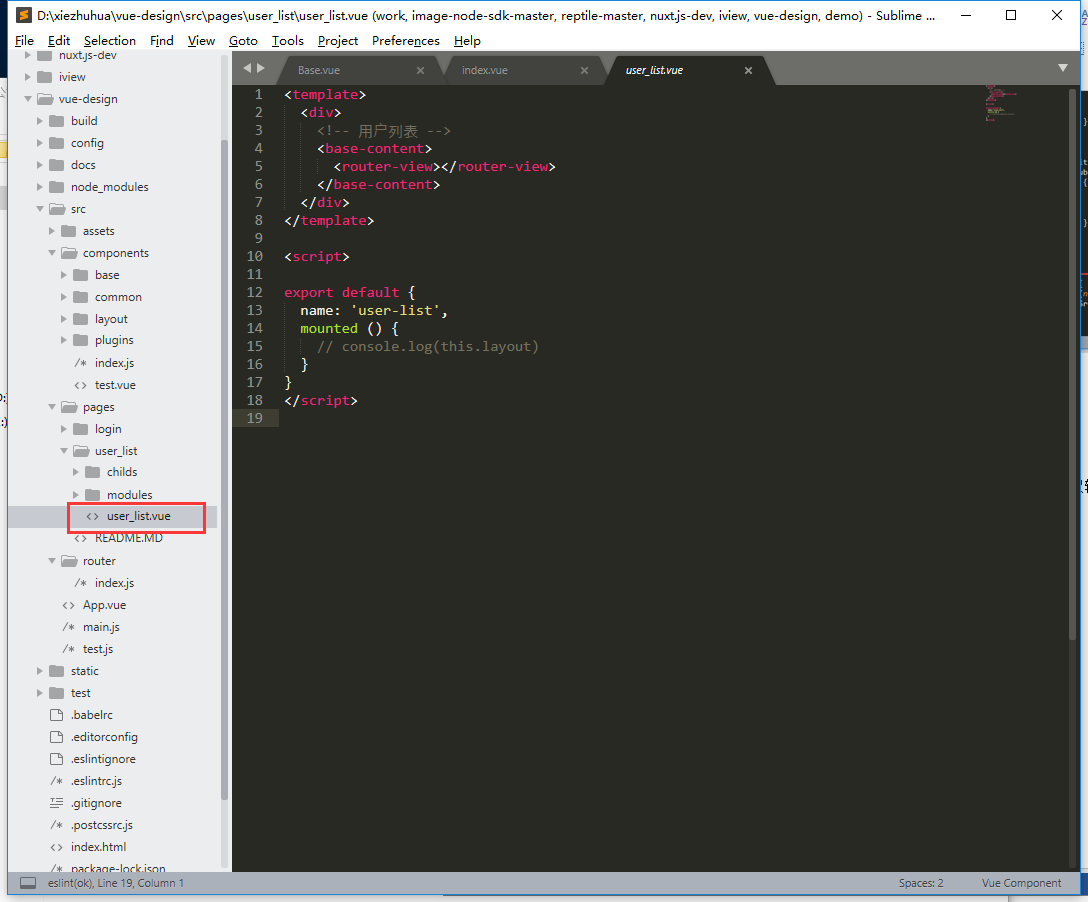
用于引用方插入内容的。

将将要编写的主体内容的填写权转移出去。

### Pages

#### /user\_list/

路由跳转过来的一个主页面



#### <base-content>

/layout/base/

这是组件创建出来的layout组件

#### <router-view></router-view>

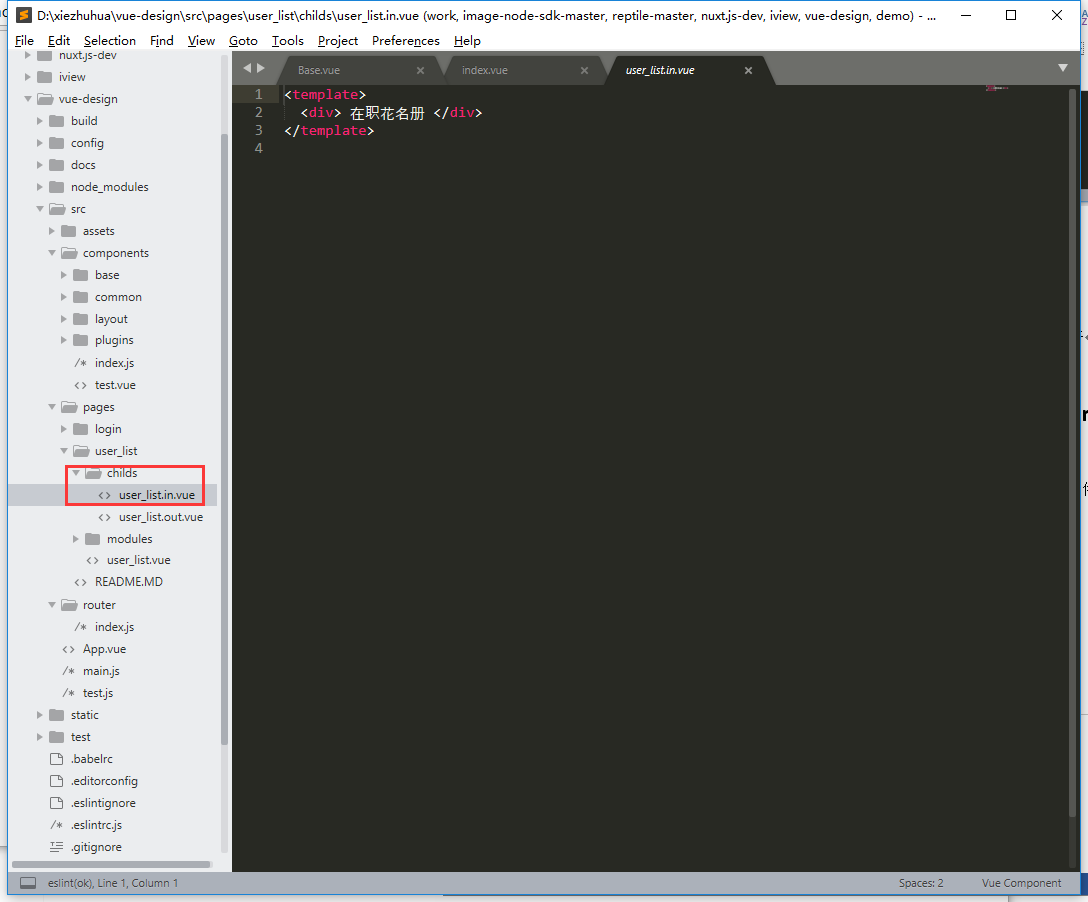
子路由

目的是展现childs目录下的子组件

这部分配合路由使用

其中一个子组件内容如下

#### /user\_list/childs/



/user\_list/childs/

### Components/business.js

定义为业务组件

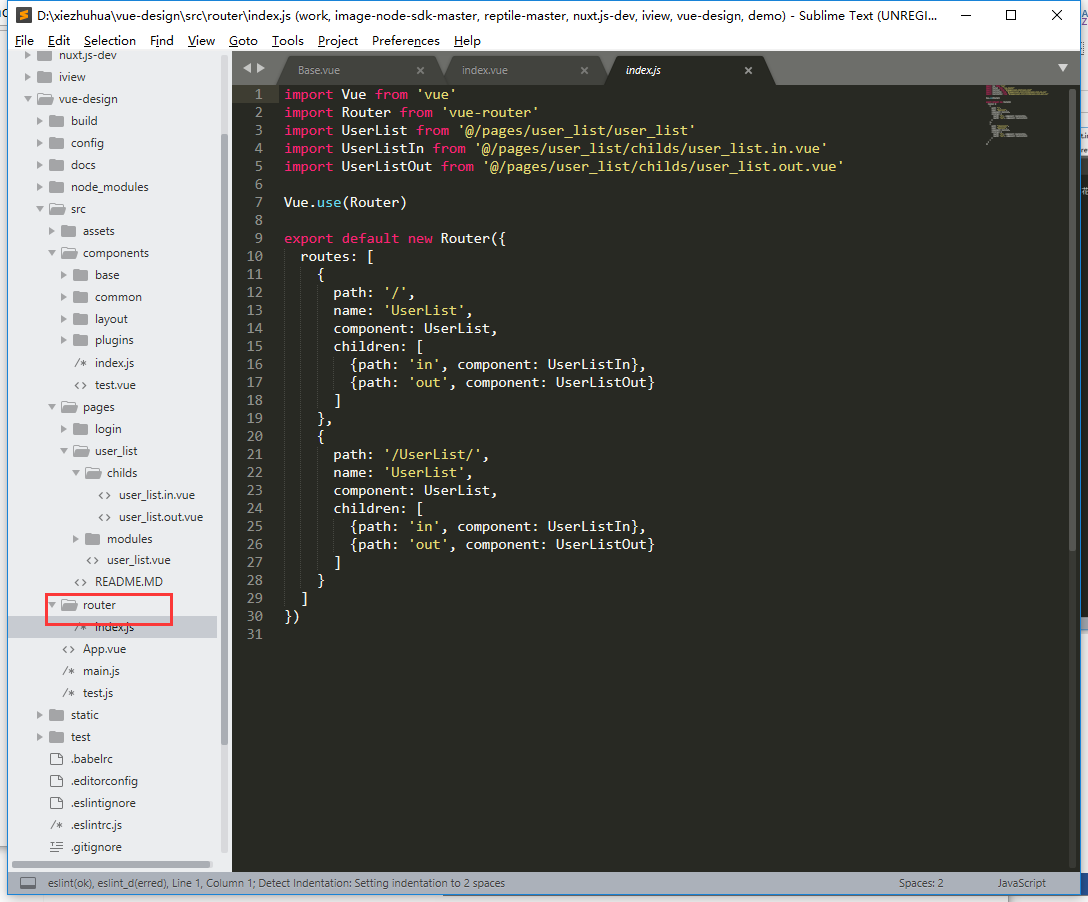
如果在pages页面里面形成了一个通用的模块，可以在这里升级为全局调用。

## Router

Import 引入组件

设置路由标识,

设置子路由



## 最后形成的效果。

组件间可以灵活插入

