

# Vue项目实战-小撩买菜

讲师：叶建华



老师答疑交流

项目整体目录结构

上海腾讯大厦

输入商品名称

新鲜毛豆1元换购

周五鲜蔬日

进入专区

蔬菜豆制品

肉禽蛋

水产海鲜

水果

乳品烘焙

米面粮油

方便速食

酒饮零食

宝宝餐

冰淇淋

限时抢购 01:40:37

更多

美国蛇果4个 720g

红扁豆 350g

湘村黑猪冷鲜肉馅 (3:7) 280g

蒙牛优 40g\*2

首页

分类

吃什么

购物车 5

我的

上海腾讯大厦

输入商品名称

猜你喜欢

恒都谷饲眼肉牛排T 150g

满59鸡大胸1元换

¥39.90

艺杏五香干 180g/袋

满18毛豆1元换

¥3.50

泰森 生鲜鸡

满18毛豆1元换

¥5.00

湘村黑猪冷鲜去皮五花肉

满18毛豆1元换

¥3.50

首页

分类

吃什么

购物车 5

我的

输入商品名称

安心蔬菜

全部

推荐

平价好菜

崇明蔬菜

豆制品

新鲜水果

肉禽蛋

海鲜水产

乳品烘焙

营养早餐

叮咚心选

米面粮油

推荐

甜玉米 约800g

甜到心里 每一粒都不想错过

满18毛豆1元换

¥6.50

普罗旺斯番茄 约350g

网红番茄 果肉沙软甜而不腻

满59鸡大胸1元换

¥8.90

鲜莲蓬 约230g

莲米饱满 夏季降火必备~

满18毛豆1元换

¥8.90

六鳌红蜜薯 750g/袋

网红红薯 细腻无丝 非你莫“..."

限时 满18毛豆1元换

¥8.29

首页

分类

吃什么

购物车 5

我的



项目整体目录结构

上海市

Q 输入您的收货地址

附近地址

绿地.蔷薇九里别墅区

松江区广富林路与龙马路交叉口东北200米

广富林路1599弄36幢

松江区松江区

绿地·蔷薇九里

松江区广富林路1518弄527-620号

松江区三湘四季幼儿园

松江区广富林路1599弄19号

上海猫心一号蔷薇别墅

松江区广富林路1518弄373号

绿地蔷薇派对别墅

松江区广富林路1518弄1-526号

广富林路1599弄31幢

松江区松江区

广富林路1599弄32B幢

松江区松江区

+ 添加新地址

<

商品详情

...

📷

艺杏五香干 180g/袋

咸鲜五香干 吃了忘不掉

¥3.50

🕒 最快17:45-18:15送达

促销

换购 满18.00元, 享超值换购;

推荐做法

🛒 5

加入购物车

选择收货地址 >

共3件 >

支付方式

微信

优惠券

0张可用 >

备注

选填, 告诉小哥你的特殊要求

商品金额

¥ 29.30

配送费

¥ 0.00

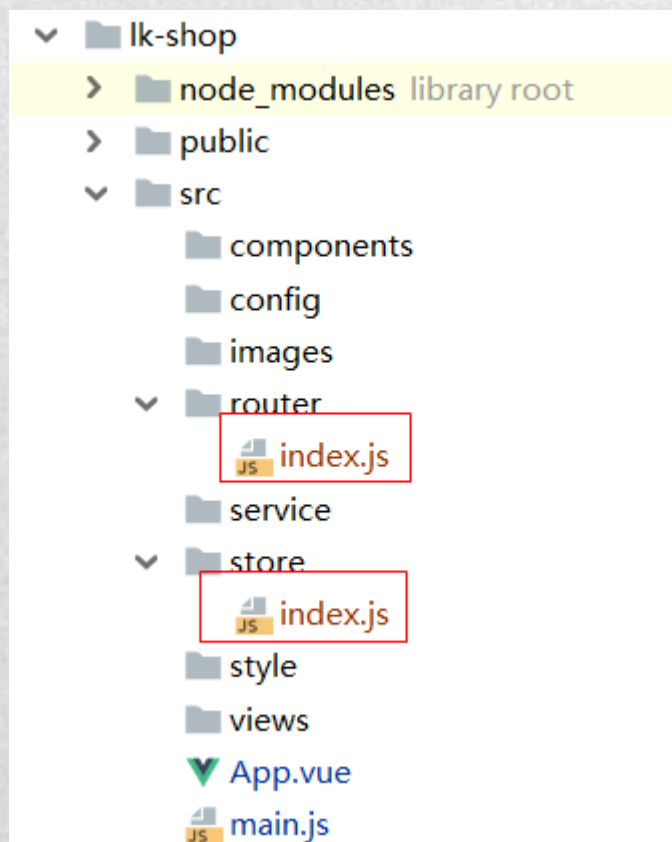
实付 ¥29.30

提交订单

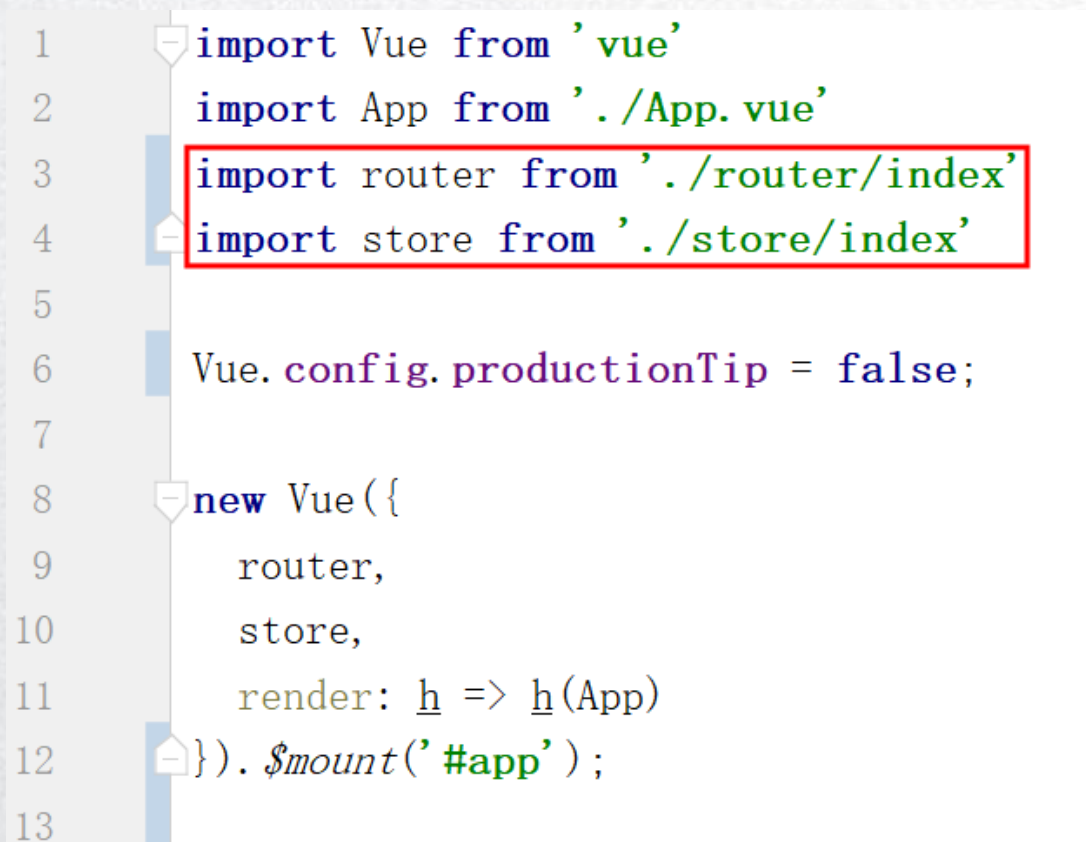
## • 项目整体目录结构



## • 项目整体目录结构



项目目录结构



main.js中配置



# • 配置FastClick

## □ 什么是FastClick?

- 1> FastClick 是一个简单易用的库，它消除了移动端浏览器上的物理点击和触发一个 click 事件之间的 300ms 的延迟；
- 2> 目的就是在不干扰你目前的逻辑的同时，让你的应用感觉不到延迟，反应更加灵敏；
- 3> 实现原理: 在检测到touchend事件的时候，会通过DOM自定义事件立即出发模拟一个click事件，并把浏览器在300ms之后真正的click事件阻止掉。

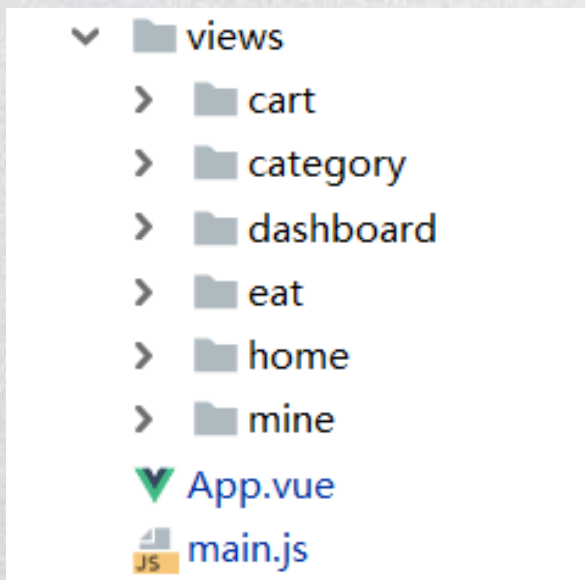
## □ 为什么会存在延迟？

- 1> 移动端浏览器会从你点击按钮之后，等待大约 300ms 才会触发点击事件
- 2> 原因是浏览器会等待看你的行为是否会双击

## □ 如何使用？

```
// 小擦提示: 引入FastClick
import FastClick from 'fastclick'
if ('addEventListener' in document) {
  document.addEventListener('DOMContentLoaded', function() {
    FastClick.attach(document.body);
  }, false);
}
```

# • 配置一级主界面



一级主界面

```
4 // 引入一级界面
5 import DashBoard from '../views/dashboard/DashBoard.vue'
6 const Home = () => import('../views/home/Home.vue');
7 const Category = () => import('../views/category/Category.vue');
8 const Eat = () => import('../views/eat/Eating.vue');
9 const Cart = () => import('../views/cart/Cart.vue');
10 const Mine = () => import('../views/mine/Mine.vue');

14 export default new Router({
15   routes: [
16     { path: '/', redirect: '/dashboard' },
17     {
18       path: '/dashboard',
19       name: 'dashboard',
20       component: DashBoard,
21       children: [
22         { path: '/dashboard', redirect: '/dashboard/home' },
23         { path: 'home', name: 'home', component: Home },
24         { path: 'category', name: 'category', component: Category },
25         { path: 'eat', name: 'eat', component: Eat },
26         { path: 'cart', name: 'cart', component: Cart },
27         { path: 'mine', name: 'mine', component: Mine }
28       ]
29     }
30   ]
31 })
```

路由配置



# • 集成UI组件库Vant

## □ 安装vant?

```
npm i vant -S
```

```
npm i babel-plugin-import -D
```

## □ babel.config.js中配置

```
1  module.exports = {  
2    presets: [  
3      '@vue/app'  
4    ],  
5    plugins: [  
6      ['import', {  
7        libraryName: 'vant',  
8        libraryDirectory: 'es',  
9        style: true  
10     }], 'vant']  
11  ]  
12  };
```



# • 集成Vant底部的Tabbar 标签栏

## ▣ 核心代码

```
<van-tabbar active="{{ active }}" bind:change="onChange">
  <van-tabbar-item info="3">
    <image slot="icon" src="{{ icon.normal }}" mode="aspectFit" />
    <image slot="icon-active" src="{{ icon.active }}" mode="aspectFit" />
    自定义
  </van-tabbar-item>
  <van-tabbar-item icon="search">标签</van-tabbar-item>
  <van-tabbar-item icon="setting-o">标签</van-tabbar-item>
</van-tabbar>
```

## ▣ 核心参数

active	当前选中标签的索引	Number	
active-color	选中标签的颜色	String	#1989fa
inactive-color	未选中标签的颜色	String	#7d7e80

# • 首页相关

## ▣ 数据接口

<http://demo.itlike.com/web/xlmc/api/homeApi>

<http://demo.itlike.com/web/xlmc/api/homeApi/categories>

<http://demo.itlike.com/web/xlmc/api/homeApi/categoriesdetail>

（注意：为了防止缓存，可在接口尾部拼接时间戳，详细代码看server文件夹中的index.js）

## ▣ 首页轮播图

```
npm install vue-awesome-swiper --save
```



# • 用户中心相关

## ▣ 数据接口

### 1) 获取短信验证码

[http://demo.itlike.com/web/xlmc/api/send\\_code](http://demo.itlike.com/web/xlmc/api/send_code)

### 2) 手机验证码登录

请求方式: POST 参数: {phone: String, code:String}

[http://demo.itlike.com/web/xlmc/api/login\\_code](http://demo.itlike.com/web/xlmc/api/login_code)

### 3) 用户名/密码登录

请求方式: POST 参数: {user\_name: String, user\_pwd : String, captcha : String}

[http://demo.itlike.com/web/xlmc/api/login\\_pwd](http://demo.itlike.com/web/xlmc/api/login_pwd)

### 4) 自动登录

<http://demo.itlike.com/web/xlmc/api/userinfo>

### 5) 退出登录

<http://demo.itlike.com/web/xlmc/api/logout>

# • 购物车相关

## □ 数据接口

### 1) 添加商品

请求方式: POST 参数: {user\_id: String, goods\_id: String, goods\_name: String, goods\_price: Number, small\_image: String}

<http://demo.itlike.com/web/xlmc/api/cart/add>

### 2) 获取购物车数据

请求方式: GET

<http://demo.itlike.com/web/xlmc/api/cart/search>

### 3) 修改购物车商品数量

请求方式: POST 参数: {user\_id: String, goods\_id: String, type: String<add, reduce>}

<http://demo.itlike.com/web/xlmc/api/cart/num>

### 4) 删除购物车数据

<http://demo.itlike.com/web/xlmc/api/cart/clear>



# • 购物车相关

## □ 数据接口

### 1) 单个商品的选中和取消选中

请求方式: POST

参数: {user\_id: String, goods\_id: String}

[http://demo.itlike.com/web/xlmc/api/cart/singer\\_select](http://demo.itlike.com/web/xlmc/api/cart/singer_select)

### 2) 所有商品的选中和取消选中

请求方式: POST 参数: {user\_id: String, flag: Boolean}

[http://demo.itlike.com/web/xlmc/api/cart/all\\_select](http://demo.itlike.com/web/xlmc/api/cart/all_select)

### 3) 查询所有已经被选中的商品

请求方式: GET

[http://demo.itlike.com/web/xlmc/api/cart/selected/:user\\_id](http://demo.itlike.com/web/xlmc/api/cart/selected/:user_id)

### 4) 删除已经生成订单的商品

[http://demo.itlike.com/web/xlmc/api/cart/del\\_checked/:user\\_id](http://demo.itlike.com/web/xlmc/api/cart/del_checked/:user_id)

# • 地址相关

## □ 数据接口

### 1) 获取当前用户收货地址

[http://demo.itlike.com/web/xlmc/api/address/search/:user\\_id](http://demo.itlike.com/web/xlmc/api/address/search/:user_id)

### 2) 添加用户收货地址

请求方式: POST 参数: {user\_id, address\_name, address\_phone, address\_area, address\_area\_detail, address\_post\_code, address\_tag, province, city, county, areaCode}

<http://demo.itlike.com/web/xlmc/api/address/add>

### 3) 获取修改用户信息

请求方式: POST 参数: {user\_id, address\_id}

<http://demo.itlike.com/web/xlmc/api/address/one>

(注意: 为了防止缓存, 可在接口尾部拼接时间戳, 详细代码看server文件夹中的index.js)



# • 地址相关

## □ 数据接口

### 1) 修改用户地址

请求方式: POST 参数: {address\_id,user\_id, address\_name, address\_phone, address\_area, address\_area\_detail, address\_post\_code, address\_tag, province, city, county, areaCode}

<http://demo.itlike.com/web/xlmc/api/address/edit>

### 2) 删除用户地址

请求方式: GET

[http://demo.itlike.com/web/xlmc/api/address/del/:address\\_id](http://demo.itlike.com/web/xlmc/api/address/del/:address_id)

(注意: 为了防止缓存, 可在接口尾部拼接时间戳, 详细代码看server文件夹中的index.js)

# • 订单相关

## ▣ 数据接口

### 1) 提交订单

请求方式: POST 参数: {user\_id, address\_id, arrive\_time, cart\_shop, notice, shop\_price, dis\_price}

<http://demo.itlike.com/web/xlmc/api/order/post>

### 2) 订单支付成功

请求方式: POST 参数: {user\_id, order\_id}

[http://demo.itlike.com/web/xlmc/api/order/change\\_status](http://demo.itlike.com/web/xlmc/api/order/change_status)

### 3) 查询订单

请求方式: POST 参数: {user\_id, status}

<http://demo.itlike.com/web/xlmc/api/order/get>



# • 微信支付

## ▣ 数据接口

### 1) 生成微信支付二维码

请求方式: GET 参数: {outTradeNo: String, totalFee: Number}

<http://47.98.157.152/WXPayProject/pay/createNative.do>

### 2) 查询是否支付成功

请求方式: GET 参数: {out\_trade\_no:String}

<http://47.98.157.152/WXPayProject/pay/queryPayStatus.do>

(注意: 已经开放白名单, 无需跨域)

### 3) vue-qrious插件使用

npm install vue-qrious -S

import VueQrious from 'vue-qrious';

Vue.use(VueQrious);

# • 定位集成

## ▣ 腾讯地图定位

### 1) 引入定位的JS文件

<https://3gimg.qq.com/lightmap/components/geolocation/geolocation.min.js>

### 2) Key

3PXBZ-DOQK6-FQISF-M2BXT-MOETT-L6FIM

### 3) 使用

```
let geolocation = new qq.maps.Geolocation("3PXBZ-DOQK6-FQISF-M2BXT-MOETT-L6FIM", "小撩买菜");  
geolocation.getIpLocation(this.showPosition, this.showErr);
```



# 喜欢IT, 就来撩课!

[www.itlike.com](http://www.itlike.com)



老师答疑交流