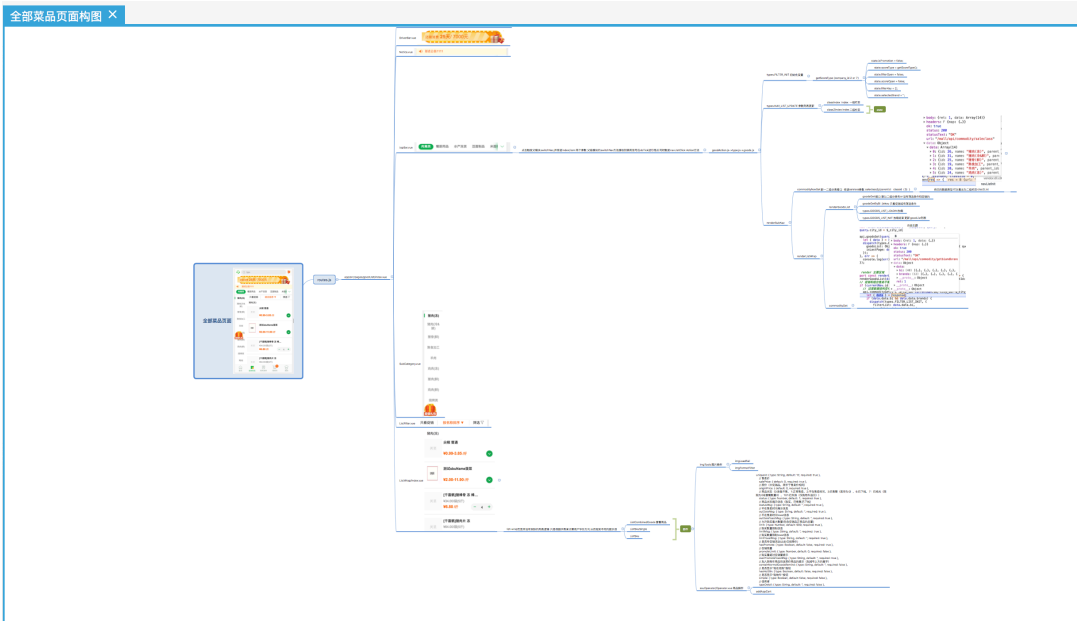


学习记录

1. 学习商城全部菜品页面源码，制作流程图



2. 学习Navigator对象属性方法

Navigator 对象属性

属性	说明
appCodeName	返回浏览器的代码名
appName	返回浏览器的名称
appVersion	返回浏览器的平台和版本信息
cookieEnabled	返回指明浏览器中是否启用 cookie 的布尔值
platform	返回运行浏览器的操作系统平台
userAgent	返回由客户机发送服务器的user-agent 头部的值

Navigator 对象方法

方法	描述
javaEnabled()	指定是否在浏览器中启用Java
taintEnabled()	规定浏览器是否启用数据污点(data tainting)

已解决

1. v-tap 学习自定义指令directive

指令directive 分为局部指令 和全局指令

```
directive: {  
  focus:{  
    inserted:function(el){  
      el.focus();  
    }  
  }  
}
```

```
Vue.directive('focus',{  
  inserted: function(el){  
    el.focus();  
  }  
})
```

钩子函数

一个指令定义对象可以提供如下几个钩子函数 (均为可选):

- `bind` : 只调用一次, 指令第一次绑定到元素时调用。在这里可以进行一次性的初始化设置。
- `inserted` : 被绑定元素插入父节点时调用 (仅保证父节点存在, 但不一定已被插入文档中)。
- `update` : 所在组件的 VNode 更新时调用, 但是可能发生在其子 VNode 更新之前。指令的值可能发生了改变, 也可能没有。但是你可以通过比较更新前后的值来忽略不必要的模板更新 (详细的钩子函数参数见下)。
- `componentUpdated` : 指令所在组件的 VNode 及其子 VNode 全部更新后调用。
- `unbind` : 只调用一次, 指令与元素解绑时调用。

接下来我们来看一下钩子函数的参数 (即 `el`、`binding`、`vnode` 和 `oldVnode`)。

钩子函数参数

指令钩子函数会被传入以下参数：

- `el`：指令所绑定的元素，可以用来直接操作 DOM。
- `binding`：一个对象，包含以下属性：
 - `name`：指令名，不包括 `v-` 前缀。
 - `value`：指令的绑定值，例如：`v-my-directive="1 + 1"` 中，绑定值为 `2`。
 - `oldValue`：指令绑定的前一个值，仅在 `update` 和 `componentUpdated` 钩子中可用。无论值是否改变都可用。
 - `expression`：字符串形式的指令表达式。例如 `v-my-directive="1 + 1"` 中，表达式为 `"1 + 1"`。
 - `arg`：传给指令的参数，可选。例如 `v-my-directive:foo` 中，参数为 `"foo"`。
 - `modifiers`：一个包含修饰符的对象。例如：`v-my-directive.foo.bar` 中，修饰符对象为 `{ foo: true, bar: true }`。
- `vnode`：Vue 编译生成的虚拟节点。移步 [VNode API](#) 来了解更多详情。
- `oldVnode`：上一个虚拟节点，仅在 `update` 和 `componentUpdated` 钩子中可用。



除了 `el` 之外，其它参数都应该是只读的，切勿进行修改。如果需要在钩子之间共享数据，建议通过元素的 `dataset` 来进行。

未解决