# 第八周总结

## 大致总结

### （一）本周学习内容：

目的：做一个前后端分离的SPA外卖应用demo，学习vue开发方式,插件、库，样式布局。

功能模块:

a．首页 分类导航，商家列表

b搜索

c订单

d．个人中心 登录管理

主要技术：

a.Vue技术栈(vue,vuex ,router)

b.前后端交互：ajax库,axios,mockjs，postman

c.ES6+babel

d.stylus

r. webpack,vue-cli,eslint

项目开发模式:

模块化，组件化，工程化(项目构建——webpack)

### （二）遇到的问题及如何解决

1．拆分静态组件时，首页页面布局有差错？

原因：css拆分出错。Stylus格式被打乱

解决：注意排查对应的style格式，利用检查工具定位元素

2．制作首页分类时，swiper滑动效果失败？

原因：swiper的创建必须在数据更新到页面之后，仅仅数据更新没用。

解决：此时，使用this.$nextTick(()=>{}),界面更新就立即调用,

并且使用watach来监视数据的改变。

3.计时器可不断被点击，时间被重置？

原因：计时器启动条件错误

解决：应当设定：如果当前没有计时才开启计时器

4. 向后端发送请求是，端口号修改与后端一致也不能请求？

原因：请求端口号被浏览修改

解决：配置代理实现跨域ajax请求。

5．初始显示异常. Cannot read property 'xxx' of undefined"？

原因：初始值是空对象, 内部没有数据, 而模块中直接显示3层表达式(a.b.c),2层表达式不会报错。

解决: 使用v-if指令(不能用v-show，控制显示的方式不同，一个销毁，一个disable)。

6.发送请求总是失败？

原因:调用接口时传入参数类型错误

解决：使用接口函数时，注意参数类型一致。多个参数使用大括号{}包裹为对象

### （三）下周计划

继续完善demo,争取下周完成demo,对vue复盘

## 学习笔记

\*\*\*\*Vue外卖项目\*\*\*\*

一.开发准备

1.项目描述(4个方面)

什么项目，做什么？

一个前后端分离的SPA应用

功能模块

首页 分类导航，商家列表

搜索

订单

个人中心 登录管理

主要技术

项目开发模式

模块化，组件化，工程化(项目构建——webpack)

2.技术选型(分类处理)

vue技术栈/全家桶——即vue以及相关插件

前后台交互——ajax库

测试接口 postman

模拟数据 mockjs

ES6+eslint

es6语法：

babel:把最新版的JavaScript编译成当前可执行的版本，

promise：进一步封装ajax请求函数，直接获取data,而不是resposne）)

css转换器：stylus

3.API接口

包含4个信息

url,data,get/post,type(返回数据类型，json)

二、开启项目开发

创建项目

安装依赖

开发环境运行

生产环境打包与发布

三、搭建项目整体界面结构

stylus的理解与使用

结构化 嵌套，与html层次结构一致

可使用变量，函数

minxin(混合)

vue-router的理解与使用

标签：router-link,router-view,keep-alive

router：路由器对象，包含操作路由的功能函数，实现编程式导航(跳转路由)

route:当前路由对象，当前路由信息数据管理容器，属性：path,meta，query,params

项目路由拆分

底部导航组件：FooterGuide

导航路由组件：Msite,Serach,Order,Profile

四、抽取组件

抽取头部组件HeadTop

slot 传递标签

props 传递一般数据

商家列表组件Shops

五、登录路由组件

静态组件

FooterGuide的显示与隐藏：通过路由的meta

六、后台项目

启动后台项目：理解前后端分离

测试后台接口：使用postman

修正接口文档

七、前后端交互

ajax请求库：axios

加强版：ajax请求函数封装：axios + promise（为了直接获取到data,而不是resposne）

接口请求函数封装：每个后台接口

八、异步数据

封装ajax:

promise+axios封装ajax请求的函数

封装每个接口对应的请求函数(能根据接口定义ajax请求函数)

解决ajax的跨越域问题: 配置代理, 对代理的理解

vuex编码

创建所有相关的模块: store/index|state|mutations|actions|getters|mutation-types

设计state: 从后台获取的数据

实现actions:

定义异步action: async/await

流程:　发ajax获取数据, commit给mutation

实现mutations: 给状态赋值

实现index: 创建store对象

main.js: 配置store

组件异步显示数据

在mounted()通过$store.dispatch('actionName')来异步获取后台数据到state中

mapState(['xxx'])读取state中数据到组件中

在模板中显示xxx的数据

模板中显示数据的来源

data: 自身的数据(内部改变)

props: 外部传入的数据(外部改变)

computed: 根据data/props/别的compute/state/getters

异步显示轮播图

通过vuex获取foodCategorys数组(发请求, 读取)

对数据进行整合计算(一维变为特定的二维数组)

使用Swiper显示轮播, 如何在界面更新之后创建Swiper对象?

1). 使用回调+$nextTick()

2). 使用watch+$nextTick()

九、 前后台交互相关问题

1). 如何查看你的应用是否发送某个ajax请求?

浏览器的network

2). 发ajax请求404

请求的路径的对

代理是否生效(配置和重启)

服务器应用是否运行

3). 后台返回了数据, 但页面没有显示?

vuex中是否有

组件中是否读取

十、登陆/注册: 界面相关效果

a. 切换登陆方式(v-if/else, )

b. 手机号合法检查(正则, 计算属性)

c. 倒计时效果(循环定时器, )

d. 切换显示或隐藏密码(text/password(2个输入框切换), 动态class(on/off))

g. 前台验证提示(v-model搜集，弹窗组件的使用)

十一、登陆/注册：交互功能

a.短信登录/密码登录

b.登录成功后页面更新

c.登陆的基本流程

表单前台验证, 如果不通过, 提示

发送ajax请求, 得到返回的结果

根据结果的标识(code)来判断登陆请求是否成功

1: 不成功, 显示提示

0. 成功, 保存用户信息, 返回到上次路由

d.注意:

使用network查看请求(路径/参数/请求方式/响应数据)

使用vue的chrome插件查看vuex中的state和组件中的数据

使用debugger语句调试代码

实参类型与形参类型的匹配问题

十二、搭建商家整体界面

1). 拆分界面路由

2). 路由的定义/配置|使用