# 第十二周总结

## 大致总结

### 本周学习内容：

准备ppt：

1. 总结ssm后台用户信息管理demo;
2. 总结Vue外卖项目
3. 总结微信小程序-天气信息项目
4. 总结公司项目-充电桩和订单中心

### （二）遇到的问题及如何解决

1．问题： 更新状态数据, 对应的界面不变化？

原因: 一般方法给一个已有绑定的对象中添加一个新的属性, 这个属性没有数据绑定

解决:

Vue.set(obj, 'xxx', value)才有数据绑定

this.$set(obj, 'xxx', value)才有数据绑定

2.问题：在利用better-scroll实现列表滚动时，无法滚动？

原因：页面内容的高度超过视口高度的时候，才出现纵向滚动

解决：在创建滚动对象前，计算并赋予页面内容真实高度

3. 问题：在回顾以前写的ssmDemo时，发现有些惨不忍睹——前端验证不友好或不完善，三层混乱嵌套太多(对参数的验证不足、业务逻辑写在了控制层、入参出参范围写得太小)，控制流程没提示等等。

原因：以前对项目开发没概念，只求功能实现即可。

解决：熟悉项目开发流程，熟悉公司业务

4.在学习Vue后再回去看Jsp时，简直感觉Vue简直太好用了。个人使用时认为

Vue好处有以下几点：

1. 前后端分离，开发更有针对性。在写后台时不需考虑如何如何跳转，如何怕请求失效什么；
2. 减轻数据库的负担、请求次数大大减少。Vue可以存数据，使得不用反复请求数据；
3. 页面控制更灵活；
4. 页面渲染快，体验更好。不用每次都去加载整个网页，当数据改变时，只需更新对应的数据绑定即可。
5. 面向对象开发思想更加明显。其模块化、组件化、工程化贯穿整个开发流程，使得开发流程更加清晰。

### （三）下周计划

学习微信小程序的开发。

## 学习笔记

### （一）Vue外卖笔记(续上周)

十二、ShopGoods组件

1). 动态展现列表数据

2). 基本滑动:

使用better-scroll

理解其基本原理

创建BScroll对象的时机

watch + $nextTick()

callback + $nextTick

3). 滑动右侧列表, 左侧同步更新

better-scroll禁用了原生的dom事件, 使用的是自定义事件

绑定监听: scroll/scrollEnd

滚动监听的类型: probeType

列表滑动的3种类型

手指触摸

惯性

编码

分析:

类名: current 标识当前分类

设计一个计算属性: currentIndex

根据哪些数据计算?

scrollY: 右侧滑动的Y轴坐标 (滑动过程时实时变化)

tops: 所有右侧分类li的top组成的数组 (列表第一次显示后就不再变化)

编码:

1). 在滑动过程中, 实时收集scrollY

2). 列表第一次显示后, 收集tops

3). 实现currentIndex的计算逻辑

4). 点击左侧列表项, 右侧滑动到对应位置

十三、CartControl组件

1). 问题: 更新状态数据, 对应的界面不变化

原因: 一般方法给一个已有绑定的对象中添加一个新的属性, 这个属性没有数据绑定

解决:

Vue.set(obj, 'xxx', value)才有数据绑定

this.$set(obj, 'xxx', value)才有数据绑定

十四、ShopCart组件

1). 使用vuex管理购物项数据: cartFoods

2). 解决几个功能性bug

十五、Food组件

1). 父子组件:

子组件调用父组件的方法: 通过props将方法传递给子组件

父组件调用子组件的方法: 通过ref找到子组件标签对象

十六、ShopRatings组件

1). 列表的过滤显示

2). 自定义过滤器

十七、 ShopInfo组件

1). 使用better-scroll实现两个方向的滑动

1). 通过JS动态操作样式

2). 解决当前路由刷新异常的bug

十八、 Search组件

1). 根据关键字来异步搜索显示匹配的商家列表

2). 如实实现没有搜索结果的提示显示

十九、 项目优化

1). 缓存路由组件对象

2). 路由组件懒加载

3). 图片司加载: vue-lazyload

4). 分析打包文件并优化