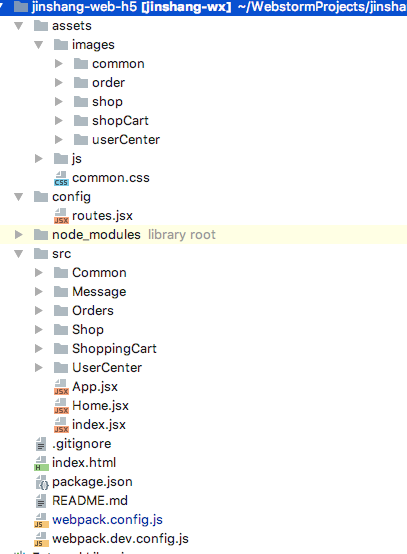
# 项目说明

1.项目概要

本项目采用npm+webpack+babel+react+weui进行项目搭建，用npm进行包管理，webpack进行打包，react框架进行应用开发，以适用微信的weui作为主样式库，并用git进行代码更新与版本控制。

2.文件目录与功能说明

项目目录如下图所示：



assets：用来存放公用的css、js和项目里所用图片。其中图片根据各自模块放在同名文件夹中。另外，打包出来的bundle.js文件也会生成在此目录下。

Config：此文件夹里的routes文件用来配置页面路由，需确保名称和路径的对应关系便于维护。

Node\_modules：依赖包存放位置。

Src：各功能模块的代码文件，Common里是公用的组件，Message、Shop、ShoppingCart、UserCenter分别对应咨询、商城、购物车和个人中心四大模块（根据各自英文名命名）。另外因为个人中心有一个隐形的大模块——订单模块，因此也将其当作一个单独目录便于长期的项目维护。

App.jsx：项目的主程序文件，所有的功能页面都作为子页面在此页面进行渲染。在此申明了历史栈HistoryManager，用于返回历史的控制。

Home.Jsx：首页文件，主要用于四大模块的切换显示。

Index.html：标准html文件，在此进行meta配置和外部文件引入。需要注意的这里的尺寸算法（1rem＝10px），用于基本的页面适配。

PackageJson：定义了这个项目所需要的各种模块，以及项目的配置信息（比如名称、版本、许可证等元数据）。npm install命令根据这个配置文件，自动下载所需的模块，也就是配置项目所需的运行和开发环境。

Webpackconfig：打包的配置文件，具体内容已在文件中注释，这里不多作说明。

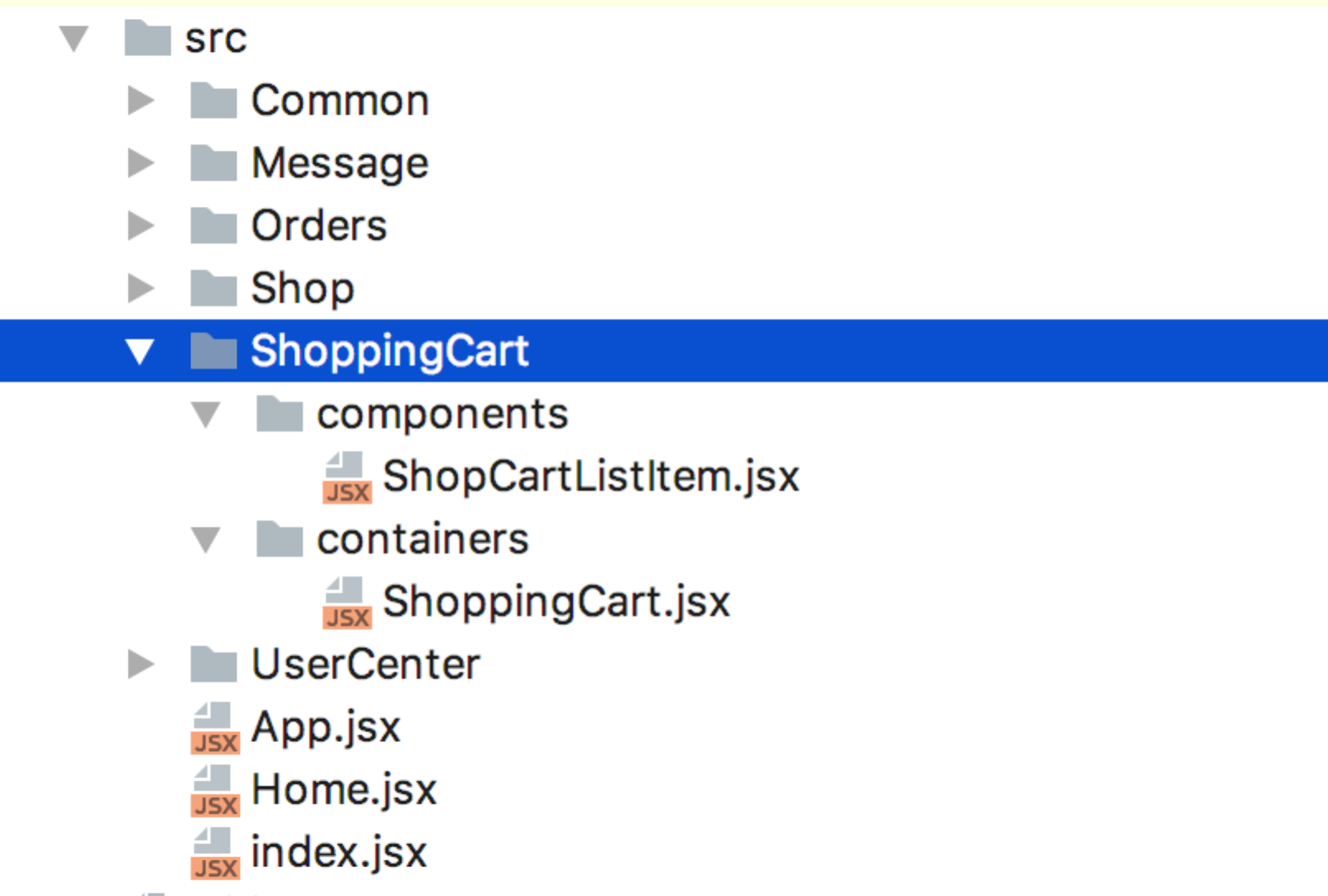
3.项目启动与打包：

启动：进入终端，在当前项目文件路径下如果第一次启动先输入npm install命令，执行完后再输入npm start命令，执行完成后在浏览器中打开地址localhost：8090即可开发。

打包：同样在当前路径下执行npm build命令。

4.代码说明

以购物车页面举例：



components文件夹存放的是组件，containers文件夹存放的是页面。



这是页面引入的依赖和各项组件，红框内的是weui的部分（具体可至react-weui官网了解），也可申明自定义的style和变量。



在react生命周期的constructor中进行state初始声明和触发事件的绑定（绑定当前this）。在componentWillMount或是componentDidMount里进行页面载入和数据载入的处理。



在render中的return方法里写入页面布局的代码，在页面初始载入时和state改变时会进行渲染，故尽量少在render里进行数据处理。

在熟悉jsx语法的情况下，可以将css写在文件内部，若不熟悉也可以用css引入的方式。

图中的map方法执行动态布局，ShopCartListItem作为组件根据接口返回的数据逐个渲染，每个Item都是购物车列表中的一项商品，组件里所需属性从外部传入，再在组建内部进行处理，同样也可加入回调事件。

页面代码使用setState（）更改state从而控制页面，然而属性props是无法内部更改的，因此组件无法进行数据处理，需要跟父组件通讯，在外部进行重新渲染。

另外，可以在defaultProps中申明各个属性的初始值。如下图：

