# 项目结构说明

1. 功能拆分
2. 按照原型稿，pc和mobile端的稿子对比，发现图片介绍详情和尺寸选择及添加到购物车可以划分为一个大的模块，在pc端和图片并列显示，在mobile端纵向显示，所以把这两块划分好之后，通过css的媒体查询去适配设备的屏幕尺寸。Pc端以横向为主轴，mobile端以纵向为主轴。
3. 接下来是查分其他细节的模块，首先是header部分，差分出来成为一个单独的header模块，其次是展示的大图拆分成一个独立的模块，再其次是把详情介绍划分为一个模块尺寸选择和添加至购物车分为一个模块，最后就是点击购物车出现的dialog分为一个独立模块。
4. 由此梳理目录结构里的component文件夹就有carttoast组件就是点击弹出的dialog，description组件即为详情介绍和价格标明的组件，header则是封装的头部组件，imagebox是大图展示的一个组件，sizebox是尺寸选择和添加至购物车的功能封装的组件。
5. Asset文件夹里存放静态图片资源
6. Store里存放redux状态管理的相关文件
7. 本项目知识要点
8. 关于媒体查询

用了css’提供的@media方法，本项目主要查询了屏幕尺寸最宽为414px，即iPhone x之类的手机端，还有最大宽度为768px，即ipad之类的平板电脑设备，基本样式是针对pc端的

1. 布局类型主要是用到了display:flex的布局

Flex-direction设置主轴是row同行排列还是colum纵向排列

Align-item设置垂直居中/水平居中（根据主轴定）

Justify-content设置水平居中/垂直居中（根据主轴定）

1. 关于react

主要采用无状态组件，只有一个sizebox是一个class声明的有状态的控制组件。

1. 关于react-redux

用class声明的组件用到的是connec高阶函数的方式去绑定状态树和dispatch一个action的，

无状态的组件如header用到了hooks，使用useSelector和use Dispatch两个方法去绑定状态树和处理dispatch的action，更为简洁高效

1. 添加了购物车的内容都存进了local storage里，刷新或关闭后打开依旧存在