##购物车逻辑

###向购物车添加商品

1.商品页点击“加入购物车”按钮，触发事件

2.事件调用购物车“增加商品”的Js程序（函数、对象方法）

3.向Js程序传递传递“商品id”、“商品数量”等数据

4.存储“商品id”、“商品数量”到浏览器的localStorage中

###展示购物车中的商品

1.打开购物车页面

2.从localStorage中取出“商品Id”、“商品数量”等信息。

3.调用服务器端“获得商品详情”的接口得到购物车中的商品信息（参数为商品Id）

4.将获得的商品信息显示在购物车页面。

###完成购物车中商品的购买

1.用户对购物车中的商品完成购买流程，产生购物订单

2.清除localStorage中存储的已经购买的商品信息

\*\*备注1\*\*：购物车中商品存储的数据除了“商品id”、“商品数量”之外，根据产品要求还可以有其他的信息，例如完整的商品详情（这样就不用掉服务器接口获得详情了）、购物车商品的过期时间，超过时间的购物车商品在下次打开网站或者购物车页面时被清除。

\*\*备注2\*\*：购物车商品除了存储在localStorage中，根据产品的需求不同，也可以存储在sessionStorage、cookie、session中，或者直接向服务器接口发起请求存储在服务器上。何种情况使用哪种方式存储、有啥区别请自己分析。

##支付流程

1.购物车页面点击“去结算”

2.选择收货人、配送方式等信息。

3.选择支付方式，微信支付、支付宝、货到付款等等。

4.点击“付款"，根据选择的支付方式不同，进入不同的支付页面。

5.在不同的支付页面中，按要求“输入密码”或者“扫描二维码”等操作后发起支付请求。

6.完成支付，跳转到指定的页面，例如：订单页面等

\*\*备注\*\*：不同的支付方式、要求填写的信息不同。请自己尝试。

##附近的人

1.打开附近人的页面

2.获取当前设备的地理位置信息（例如：经纬度）

3.向服务器请求“获得附近的人信息”的接口（提交响应的参数、例如经纬度、搜索范围）

4.将从接口获得的“附近的人的”信息展示在页面。

\*\*备注1\*\*：获得当前设备的地理位置信息，通常是指调用设备的原生的接口访问Gps感应器的数据。Html5也有获得地理位置信息的api navigator.geolocation.getCurrentPosition

\*\*备注2\*\*：除了附近的人呢，附近的车、附近的店原理都非常类似。