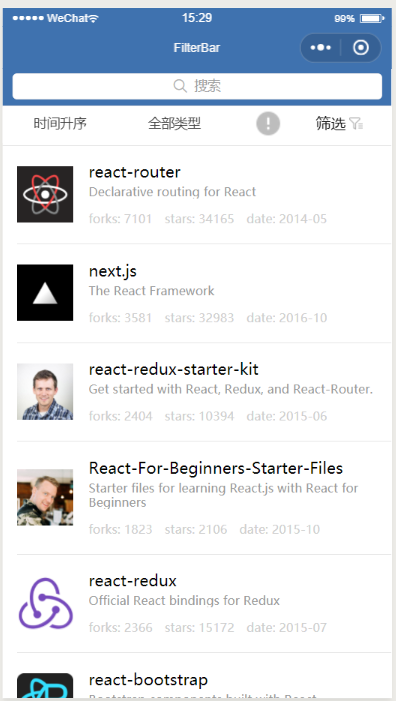
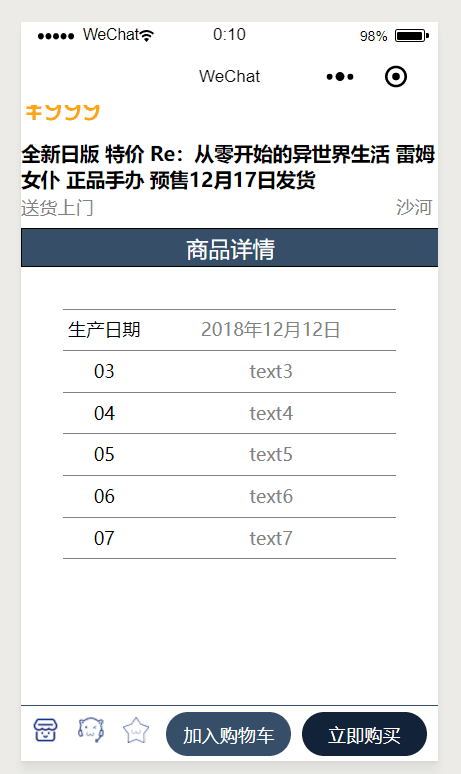
后端实现功能的解释和实现方法：

1.搜索页面：



实现云开发库内，**各类型商品的展示（集合储存数据，利用类型标签进行分类）**，并且对**时间排序**，**价格排序**，**排列方式改变**，**筛选等功能**全部实现（集合内每一个物品都设定有自己的特定标签属性，在进行排序时利用微信的搜索功能即可实现），保证用户能直接找到自己想要的物品。

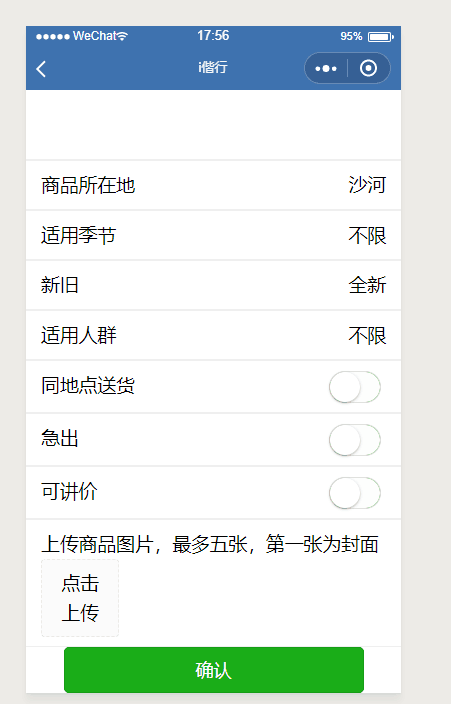
2.物品详细界面



保证在**搜索页面点击后**能进入到页面后能**显示出对应的内容（搜索页面点击后对应到集合内的那组数据，然后在页面的不同位置进行展示即可）**，注意下方**收藏**和**加入购物车功能**的实现（创建“收藏夹”、“购物车”集合（会自动和使用者的open\_id进行对应），在点击后将物品的特定id导入即可）。

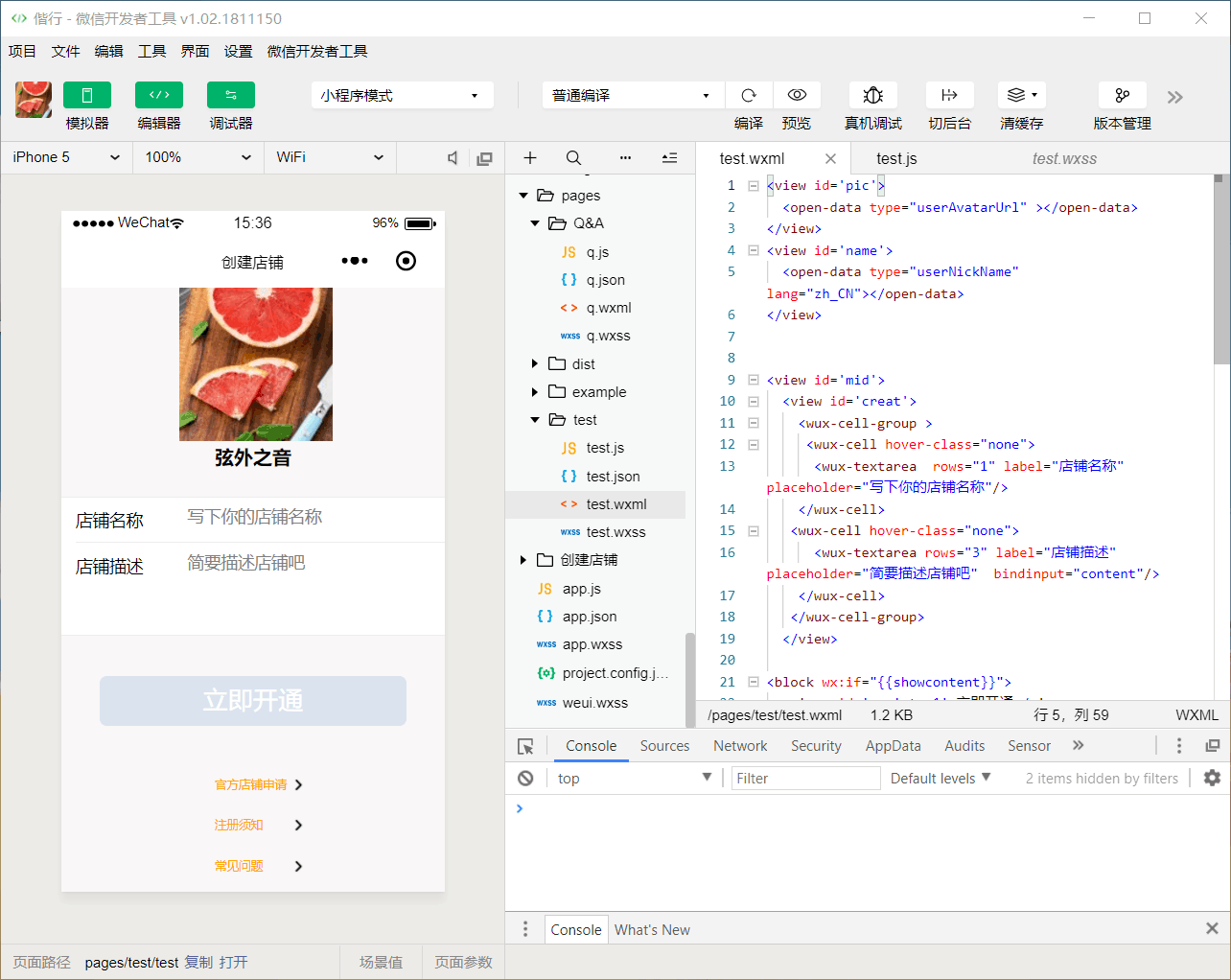
对应的收藏夹、购物车页面也是相同的实现原理，在数据库内搜索与用户的open\_id对应的收藏夹或购物车，然后根据物品的id展示信息即可（先实现文字的展示，图片不用添加）。

3.上传商品页面



注意各信息**和商品储存的联系（上传商品=》将上面选择的各类选项当做物品的标签导入到集合中即可）。**注意图片和商品的联系（图片上传到云数据库中，然后和物品创建后的id实现对应即可）

4.店铺页面



在点击开通后**将数据进行储存（建立新集合包括填写的名称和描述）**，注意**两者之间的联系**，并且**和当前的用户进行连接**（和用户id进行连接）。

同时注意如果当前用户**已经有了店铺（搜索当前用户的open\_id如果已有集合便返回即可）**，再点击**开通时进行提醒并且不上传数据**。并且跳转到我的店铺，展示的数据便是自己上传的各类商品信息（在店铺集合内根据open\_Id搜索物品然后展示即可）

5.个人信息页面



首先**每个人**在云开发库中需要设定一个**包含所有信息的集合**。

内容就包括之前提到的**能修改的各种东西**：**收货地址，店铺名称，店铺描述，（作为标签储存数据即可）我的订单（注意进行是否收货标签的设置）**。

并且在**弹出栏**能**对收货地址进行修改**：先**选择大地址**：沙河，学院路；**然后自己填具体**的：几公寓几室（对应个人集合的地址标签的修改）。**点击我的店铺**后能跳转到一个新的页面，上方标题为我的店铺，下面有**店铺名**称和**描述**以及自己出售的商品（调用open\_id对应的商品进行展示），修改信息功能暂时不实现。

6.查看订单页面



根据open\_id对应的购物车内的数据进行搜索，然后在页面内进行展示即可。