**IOS APP说明文档**



**题目 团购Demo**

**作者姓名 陈受恩**

**作者学号 21551106**

**学科专业 移动互联网与游戏开发**

**所在学院 软件学院**

**简言**

近年来，由于团购所带来的价格优惠，吸引越来越多的用户使用团购软件进行购物消费。市场上也涌现了许多优秀的团购软件，如：美团、大众点评、百度糯米等等。我仿照美团开发了一个团购demo，下面对其进行详细的说明。

**主要功能解析**

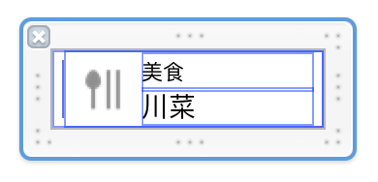
**（1）自定义导航栏**

下图是导航栏效果：

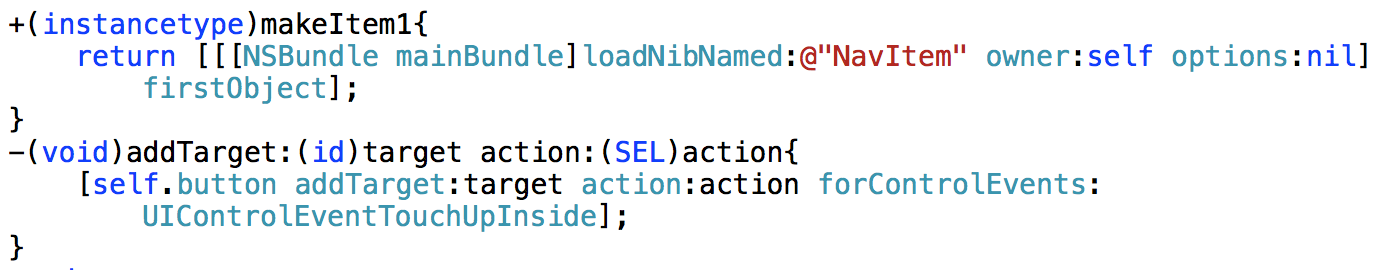
 导航栏由logo图片、3个自定义UIBarButtonItem、自定义图片按钮组成。下面主要介绍自定义UIBarButtonItem的开发详情，以第一个Item为例，

首先创建继承自UIView的NavItem类，再创建名称为NavItem的View的xib（xib名和类名一致），最后关联xib和类。

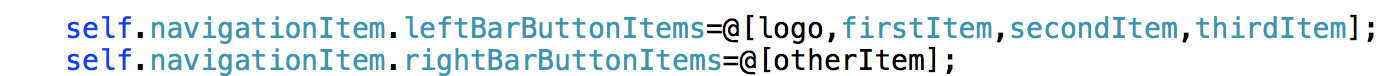
Xib布局如下：



将标题label、子标题label和button和类进行关联，并在button上添加action，NavItem类中设置创建xib的item函数，部分关键代码如下所示：



同理，后面2个item也是这么依样画葫芦，具体见源代码，这里就不再赘述。最后，把这些item添加到导航栏：



这样就完成了自定义导航栏的UI布局

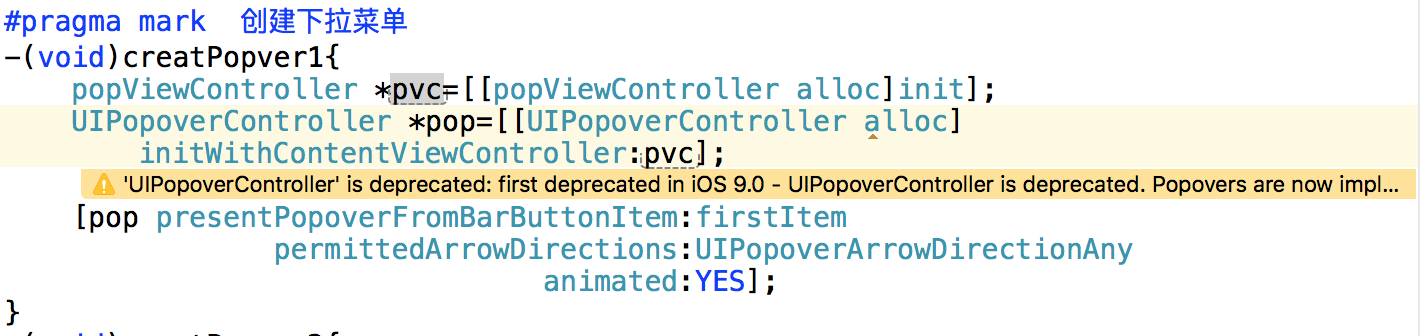
**（2） 下拉菜单**

当点击导航栏里的3个item时，会弹出相对应的下拉菜单，并且3个item所对应的下拉菜单各异，因此做法也不同，下面对3个下拉菜单进行详细的讲解。

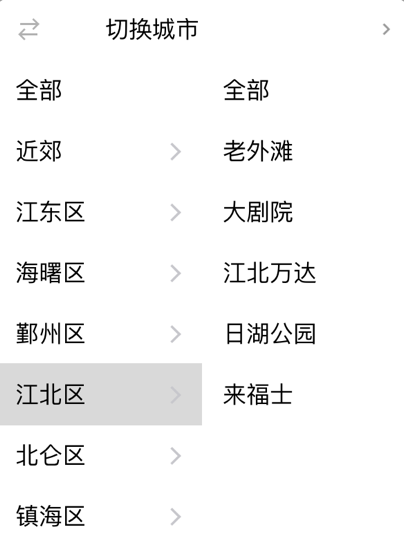
**2.1第一个下拉二级菜单UI布局如下所示：**



这是由2个tableview并排所组成，在第一个tableview中每个cell由图片＋文字所组成，第二个tableview展示的是第一个选中cell的附属信息。所以先准备菜单，和自定义Item类似，加载xib文件即可，再创建下拉菜单，ipad官方提供了UIPopoverController，直接调用函数即可，部分关键代码如下所示：



**2.2第二个下拉菜单是3个下拉菜单中最复杂的一个，UI布局如下所示：**

****

结构上主要可以分成2个部分：切换城市按钮＋类似第一个菜单的下拉二级菜单。

其中，下拉二级菜单部分可以套用第一个菜单的方法，但是这样会略显累赘，所以，我的做法是对第一个popview进行了封装，制定相关协议，第二个菜单通过遵行协议，实现方法，来复用popview。

点击切换城市按钮，弹出切换城市页面，可以通过搜索城市来切换更改当前城市，这部分详解见搜索城市部分。

**2.3第三个下拉菜单比前2个菜单简单，UI布局如下：**



这部分因为比较简单，就没有使用xib，而是用代码写的，主要就是构造了7个button，并且button上添加action，当点击其中一个button时，发送相应通知，进行传值处理。

**（3） 搜索城市**

搜索城市功能UI布局如下所示：



主要有这么几部分组成：搜索框、城市名称列表。这里主要是遵行tableview协议和searchbar协议，实现相应方法来做的。当用户输入一个城市名称，相应方法开始搜索数据中包含该拼音的城市，并将其显示到结果tableview。当用户切换城市时，相应协议发送通知，这样痛过通知就可以及时对用户的操作进行反馈。部分关键代码如下所示：



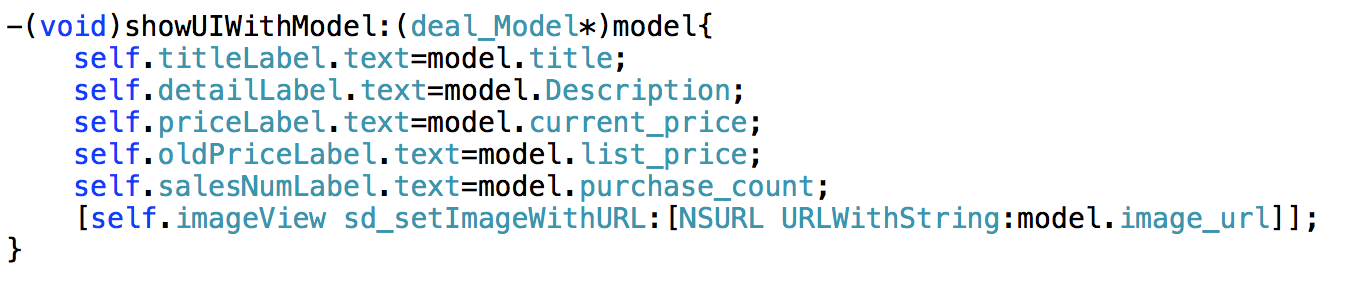
**（4）商品展示**

商品展示部分是采用自定义UI CollectionCell实现的，它的UI布局效果如下所示：



结构上分别是UIImageView+标题Label+详情Label＋现价label＋自定义的原价label＋销售量label。

其实，这一部分主要是对网络请求的数据进行解析后，再将数据显示到对应的控件中，数据模型转换的关键部分代码如下所示：



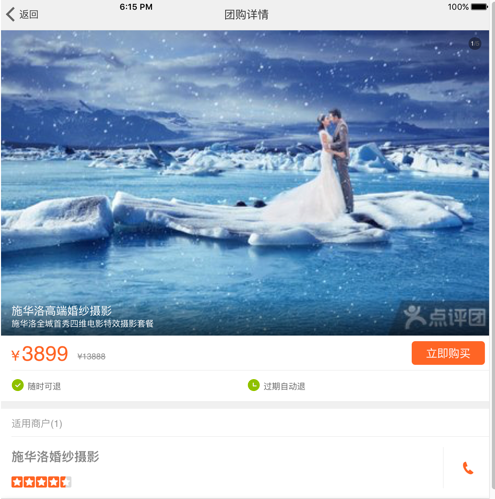
**（5）商品详情**

当点击一个商品展示cell时，跳转到商品详情页面，详情页面是是一个自定义的分栏控制器，这也是ipad与iphone开发不同点之一，左右两侧的view分别是自定义view和webview。左侧view具体UI布局如下所示：



从结构上来说有：UIImageview、标题Label，详情Label，购买button，4个带图片的button。

右侧view是个简单的webview，访问的大众点评API，分析数据模型得到url后，web加载url所得，效果如下所示：



**（6）新闻链接**

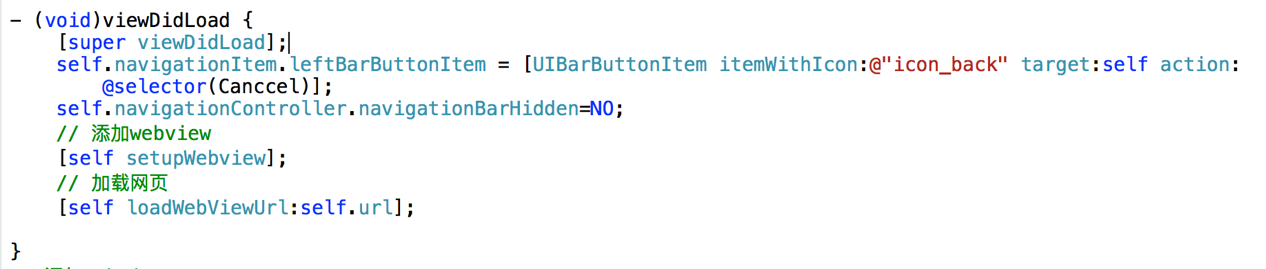
导航栏右侧item是更多的链接，本来想完成一个生活类的APP，受时间限制，这里只实现了新闻访问的功能，使用了第三方的menu库，效果如下图所示：



这一部分的关键代码如下所示：



这里展示的百度新闻链接按钮，其他也类似，得到url后，加载相应web页面，这部分代码如下所示：



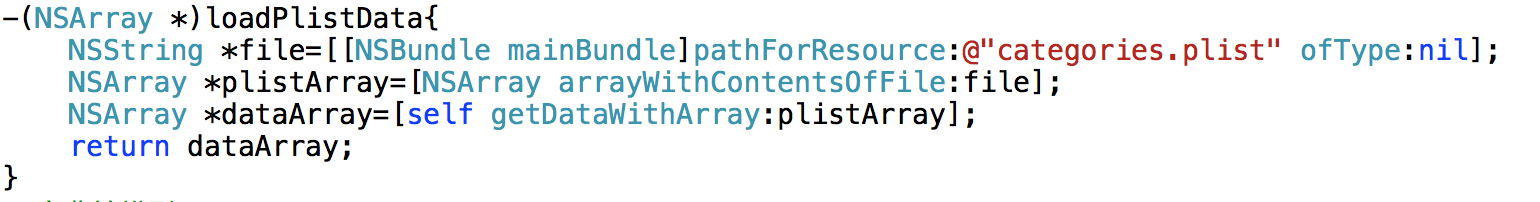
**重难点解析**

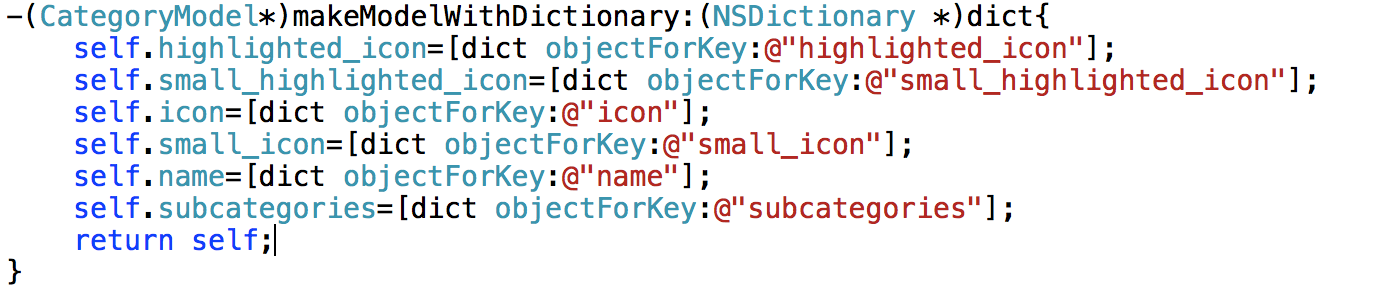
**（1）读取plist文件数据，转换模型**

这个主要运用在3个下拉菜单中，每个下拉菜单都是加载plist文件中的数据，这里以第一个分类item为例进行说明。

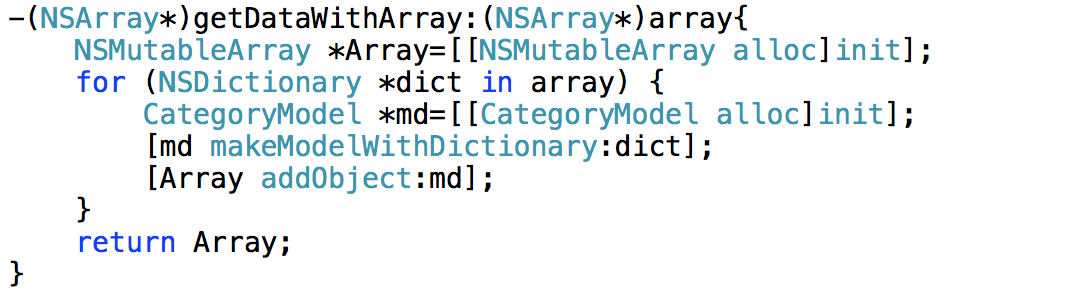
首先，观察plist文件可知，它里面的元素是dictionary，所以决定在读取plist文件时，是将其装到Array中，而Array中的元素是dictionary。主要分3步：

**NO.1加载plist文件**

 **NO.2字典转数据模型**



**NO.3解析数据**



经过这三步，就可以加载plist数据数组，解析数组里面的字典模型。

**（2）使用通知中心传值**

虽然设置代理和协议也可以监听，但是这会比较麻烦，通知中心就会方便得多。通知中心传值，分成3步，这里以排序item为例：

**NO.1添加观察者，绑定方法**



这里绑定的方法是sortChange：，接受的通知名字是@“sortDidChanged”

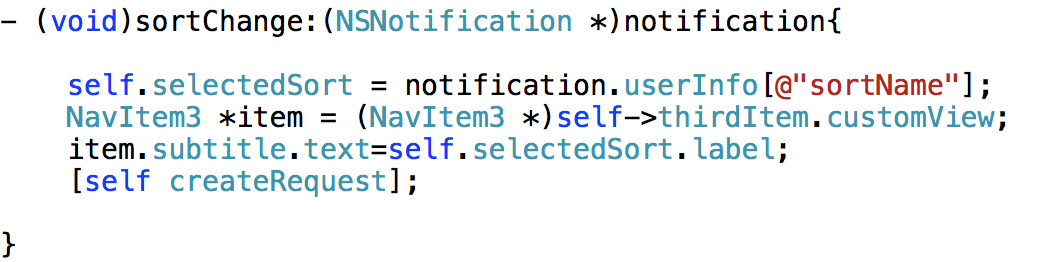
**NO.2发送通知与传值**

当用户点击sort按钮时，发送通知并传值。



这里发送的通知名称是@“sortDidChanged”，传递的key是@“sortName”，value是buton.sort。

**NO.3接受信息，实现方法**



这部分代码是接受信息，并且刷新item上的标题与发送网络请求，网络请求部分在下一部分讲解。

**（3）发送网络请求，并解析网络数据**

这里使用的是大众点评的API，网络请求就变得较容易，只需遵行<DPRequestDelegate>协议里面的方法，提供相应的参数，代码如下：



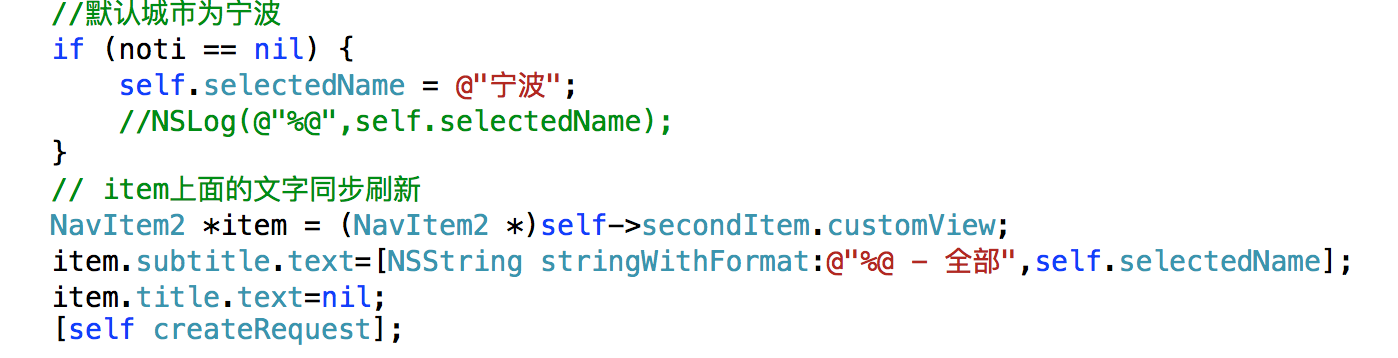
通过发送request后，便得到了网络json数据，下面就是解析这些请求到的数据，把数据转换成模型，结合这个应用选取需要的数据，下面是json数据转换成模型的方法，代码如下：  


通过上面的方法，就得到了网络数据模型的数组arr。而这个是整个应用的精髓所在，因为商品展示和商品详情里面的数据都是从这个arr里面获取的。

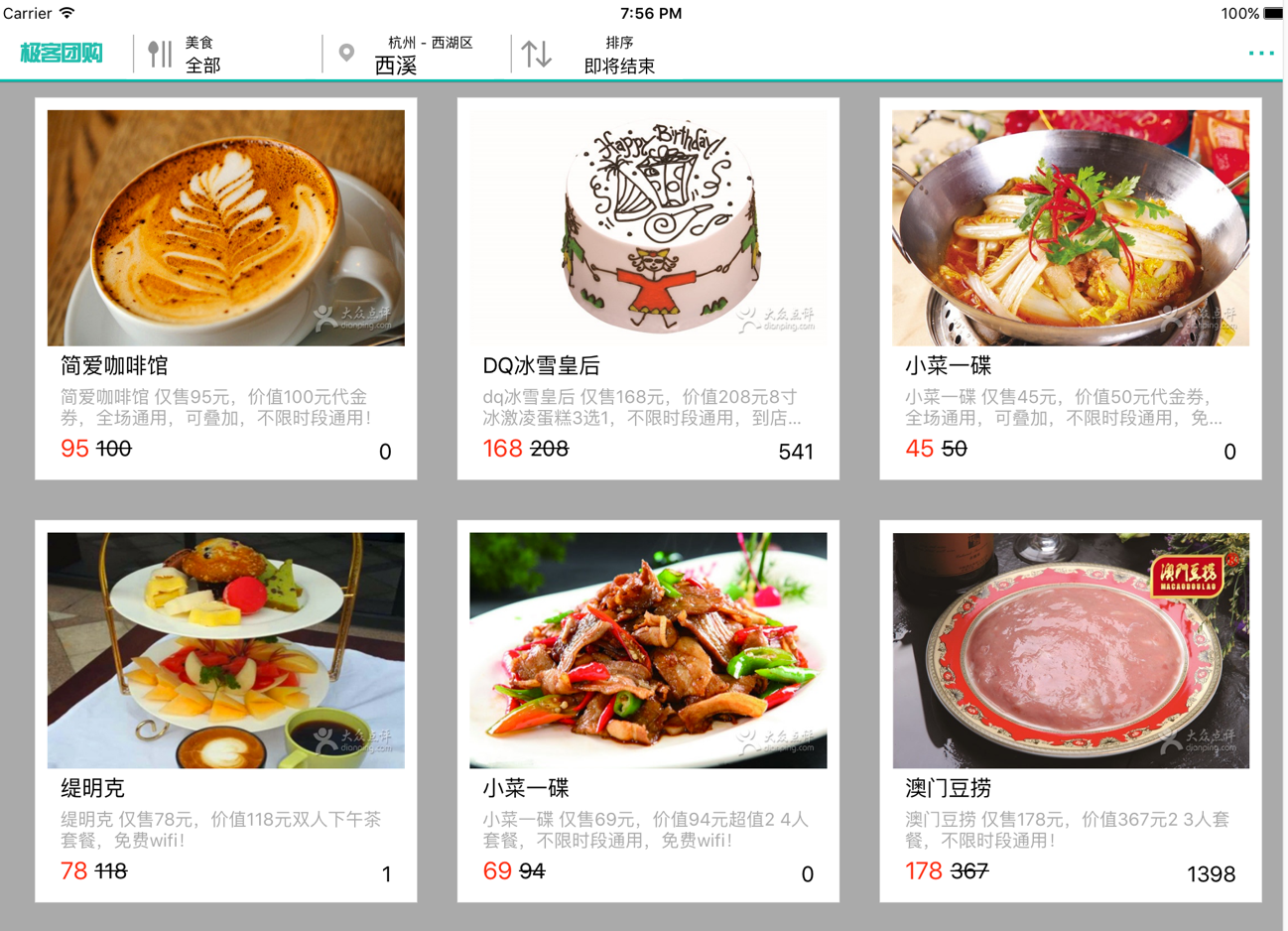
**一些细节**

（1）由于Storyboard不适用于ViewController的复用，在这个项目中，我使用代码+xib的方式来编写，值得一提的是，Xcode7版本要求必须设定rootViewController，所以初始化UIWindow与Xcode6时有所不同，具体部分代码如下：

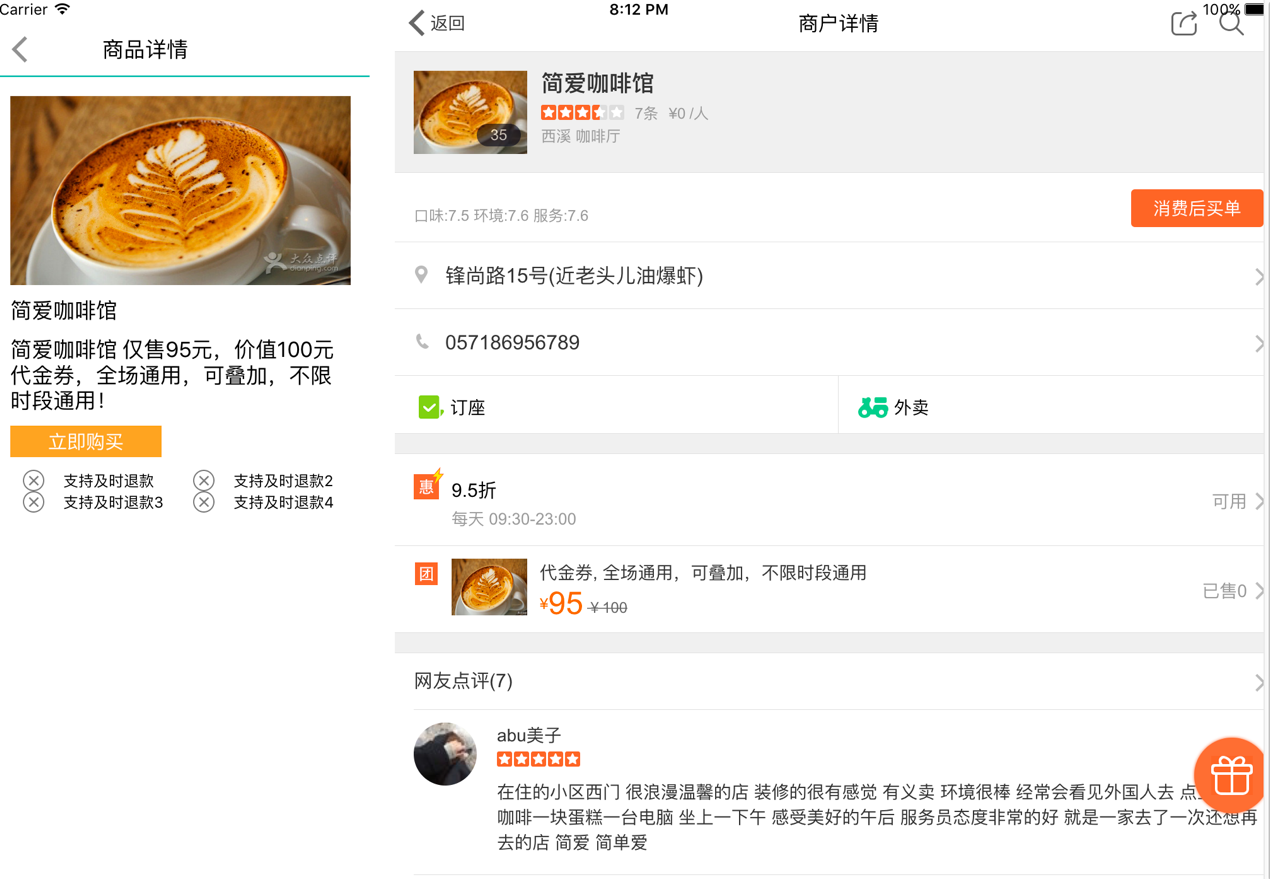


（2）当用户第一次打开应用时，并不希望看到空白的界面，所以这里需要设置默认的信息，我设置了城市为宁波地区，分类是全部，排序是默认的，这样一定程度上提升用户体验。代码如下：  


**附录**



主界面



商品详情界面