

电商系统是如何设计的？

在需求还不太明确的情况下，比较可行的方式就是，先把那些不太会变化的核心系统搭建出来，尽量简单地实现出一个最小化的系统，然后再逐步迭代完善。

电商系统的核心流程：

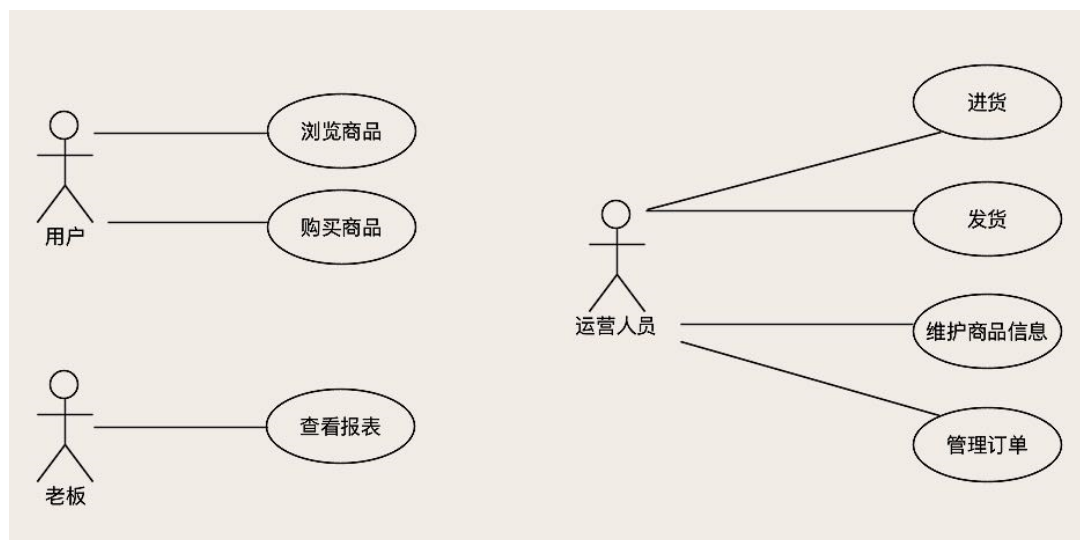
1.需求分析

不要一上来就设计功能，而是先要回答下面这两个问题：

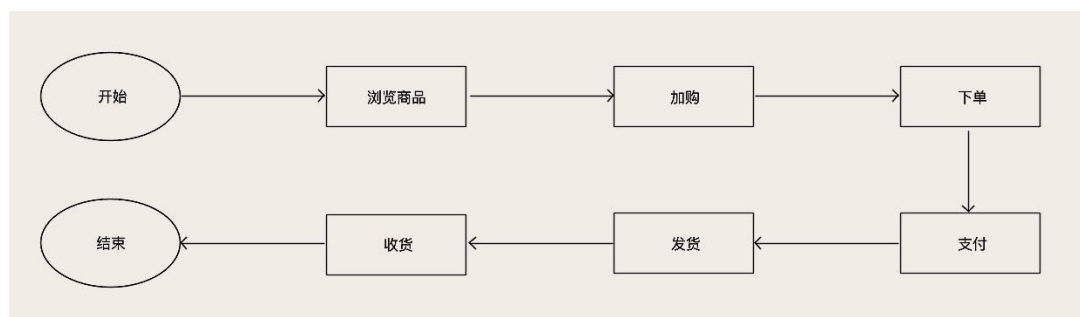
这个系统（或者是功能）是给哪些人用的？

这些人使用这个系统来解决什么问题？

用例图：



然后我们分析电商系统的流程。显然，一个电商系统最主要的业务流程，一定是购物这个流程。你应该很容易就能把这个流程分析出来，它的流程图是这样的：



1.如何根据流程来规划功能模块：

接下来，我们把这个业务流程再细化，看一下电商系统如何实现这个流程？我把细化之后的流程，绘制成了下面这个时序图（Sequence Diagram）：

我们一起看下这个时序图中的每个步骤。

用户开始浏览商品，需要有一个商品模块来支撑，给用户展示商品的介绍、价格等等这些信息。

用户把选好的商品加入购物车，这个步骤，也需要一个购物车模块来维护用户购物车中的商品。

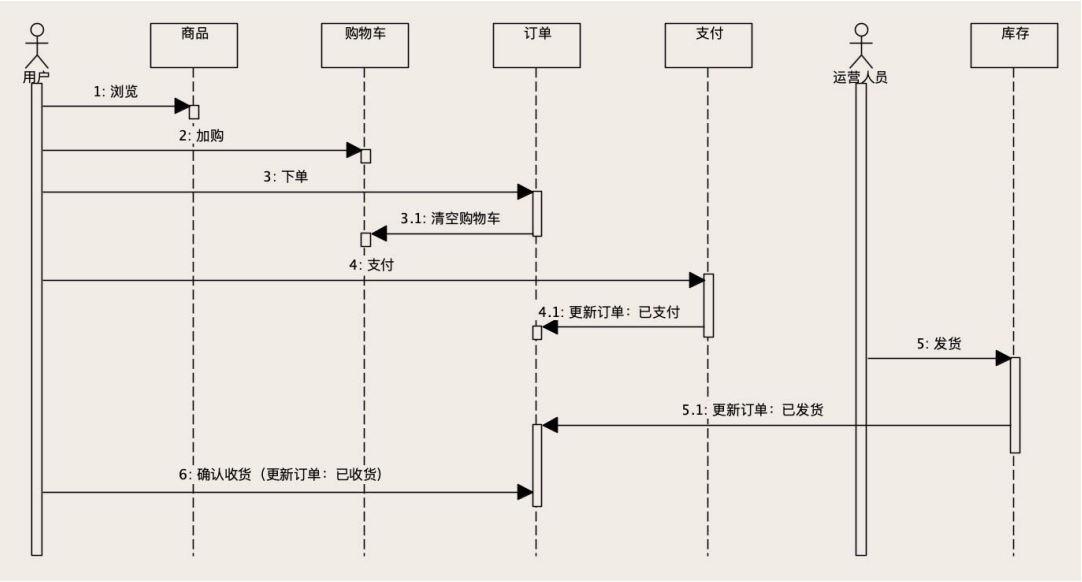
用户下单肯定需要一个订单模块来创建这个新订单。订单创建好了之后，需要把订单中的商品从购物车中删除掉。

订单创建完成后，需要引导用户付款，也就是发起支付流程，这里需要有一个支付模块来实现支付功能，用户成功完成支付之后，需要把订单的状态变更为“已支付”。

之后运营人员就可以发货了，在系统中，发货这个步骤，需要扣减对应商品的库存数量，这个功能需要库存模块来实现，发货完成后，还需要把订单状态变更为“已发货”。

最后，用户收货之后，在系统中确认收货，系统把订单状态变更为“已收货”，流程结束。

这个流程涉及到的功能模块有：商品、购物车、订单、支付和库存，这几个模块就是一个电商系统中的核心功能模块。



相比购物这个流程，剩下的几个用例和流程都没那么复杂，用同样的方法就可以把其他功能模块分析出来。在这里我们就省略分析过程，直接给出我们电商系统的功能模块划分：

上面这个图，我使用的是 UML 中的包图（Package Diagram）来表示。整个系统按照功能，划分为十个模块，除了购物流程中涉及到的：商品、订单、购物车、支付、库存五个模块以外，还补充了促销、用户、账户、搜索推荐和报表这几个模块，这些都是构建一个电商系统必不可少的功能。我们一个一个来说每个模块需要实现的功能。

商品：维护和展示商品信息和价格。

订单：维护订单信息和订单状态，计算订单金额。

购物车：维护用户购物车中的商品。

支付：负责与系统内外部的支付渠道对接，实现支付功能。

库存：维护商品的库存数量和库存信息。

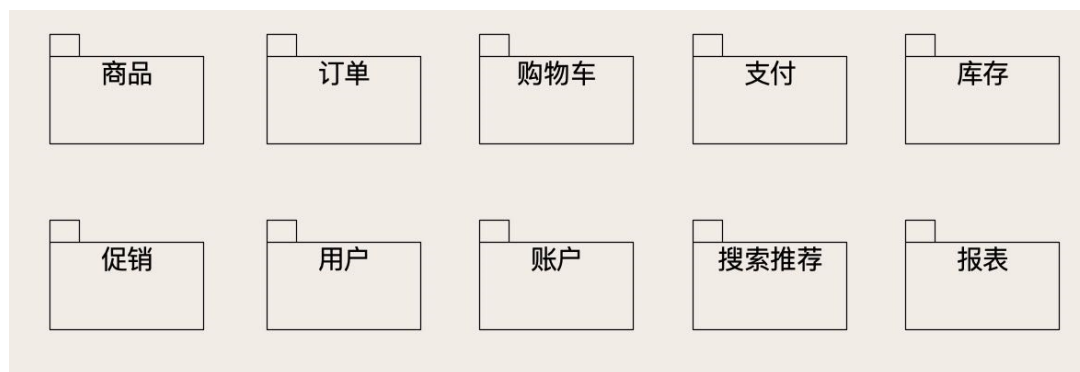
促销：制定促销规则，计算促销优惠。

用户：维护系统的用户信息，注意用户模块它是一个业务模块，一般不负责用户登录和认证，这是两个完全不同的功能。

账户：负责维护用户的账户余额。

搜索推荐：负责商城中，搜索商品和各种列表页和促销页的组织和展示，简单的说就是决定让用户优先看到哪些商品。

报表：实现统计和分析功能，生成报表，给老板来做经营分析和决策使用。



这里面需要特别说一下促销模块，它是电商系统中，最复杂的一个模块。各种优惠券、满减、返现等等这些促销规则，每个都非常复杂，再加上这些规则叠加计算，常常是复杂到连制定促销规则的人都搞不清楚。

可行的做法是，把促销模块与其他模块的接口设计的相对简单和固定，这样系统的其他模块就不会因为新的促销玩儿法而改变。

在创建订单时，订单模块把商品和价格信息传给促销模块，促销模块返回一个可以使用的促销列表，用户选择好促销和优惠，订单模块把商品、价格、促销优惠这些信息，再次传给促销模块，促销模块则返回促销价格。

促销需要提供几个服务：

一是哪些促销可用：输入是一堆商品和用户ID，促销服务返回可用的促销：比如满减、优惠券之类的。

一个“算价”服务，输入是：一堆商品和使用的一堆促销，促销服务需要验证这些促销是否都可以使用，有没有冲突之类的，如果可用，返回的是计算完成的促销价。

这个促销价不仅包括使用促销之后的总价，还要包含每个商品的“实际优惠价（不显示给用户）”，保证：促销后的总价=Σ所有商品的实际优惠价。这个价格就是用完促销之后，每个商品的实际价格，后续退货、计算成本收益，都以这个价格为准。

这样对于购物车和订单来说，处理就比较简单了，不管你促销怎么玩儿，反正最终你告诉我用完促销之后每个商品什么价格就可以了。促销规则变来变去也不用去改订单服务。