电商的项目总结分析(别人的)：https://blog.csdn.net/lao\_wang\_tou/article/details/79058695

1.系统架构

采用当前最流行的ssm（springmvc+spring+mybatis）框架开发，是当前电商网站首选的技术架构。系统是基于SOA架构设计，采用dubbo作为服务中间件，系统后台使用jsp作为视图层，操作简便用户体验好。商城系统使用freemarker做静态化页面来提高系统的性能，使用nginx做负载均衡服务器以应对大规模的用户量的并发。电商搜索系统采用当前最流行的全文检索技术solr实现。系统中使用redis做缓存，使用Activemq做消息中间件。后台数据库使用mysql数据库，做读写分离

2.dubbo的使用方法

Dubbo是一个分布式服务框架，提供了统一的高性能的远程服务调用平台。所有的业务逻辑都使用dubbo发布供表现层工程调用。发布dubbo服务需要使用spring容器的支持来发布服务，调用服务同样使用spring容器来应用服务。其中服务的发布和服务的发现都是通过注册中心来实现，我们使用zookeeper作为注册中心。使用dubbo的监控中心监控中心来统计各服务调用次数，调用时间等。

3.如何保存图片的

在电商项目中需要保存大量的图片，需要一个独立的图片服务器来保存，而且存储的容量需要可扩展。并且还需要解决在高并发及高可用的问题。所以我们采用一个FastDFS一个分布式文件系统来保存图片。FastDFS可以搭建服务器集群，解决了存储空间的水平扩展、负载均衡以及服务器的高可用问题。

4.网页静态化的实现方案

创建一个独立的工程，此功能的功能就是生成静态化页面的，例如商品详情页面静态化。将此工程独立部署到一个服务上，页面就生成到当前服务的磁盘上，并且此工程监听MQ的消息，一旦后台工程添加商品，此工程将接收到消息并且生成静态页面。在此服务器上安装一个nginx做为访问静态资源的http服务器。

5.把商品数据放到缓存中，如何提高缓存的利用率

做商品数据的缓存时，因为商品的数据量很大，而且缓存是把数据保存到内存中，此时不可能把所有的商品数据都放到缓存中。所以需要设置商品数据缓存的有效期，当用户访问到非热点数据后，此数据放到缓存中，当缓存到期后就从缓存中删除，而且长时间不会添加到缓存。而热点数据一旦从缓存中删除会马上又添加到缓存。这样可以提高缓存的利用率，同时也减轻了数据库的压力。

6.如何实现缓存同步的

只要使用了缓存就涉及到缓存同步的问题。缓存同步其实就是当缓存的信息发生变化，也就是后台对缓存的数据进行增、删、改操作后，数据库中的数据发生了变化同时要把缓存中的数据对应删除即可。当页面再次请求数据时，缓存中不能命中就会从数据库中查询并且添加到缓存中，即实现了缓存同步。

7.如何处理数据量大、并发量高的搜索

如果要搜索的内容数据量很大并且并发量很高的情况下，一个solr服务是不能满足要求的，所以此时需要SolrCloud来解决。SolrCloud也就是solr的分布式解决方案。是zookeeper+solr实现的。

8.Activemq有几种消息通信方式

使用MQ中间件可以有两种通信方式queue和topic。Queue可以实现点到点之间的通信，可以有多个Producer也可以有多个Consumer，但是消息只能被一个Consumer接收，一旦消息被消费后就没有了。

Topic可以实现类似广播的通信方式，可以有多个Producer和多个Consumer，一旦有Producer发送消息后，此消息可以被所有Consumer接收。

本项目中，当后台系统对商品数据进行添加、删除、修改后，将会发送一个消息，此消息通过topic进行通信，有多个消费端，搜索系统会把索引库进行同步，商品的缓存数据会进行同步、商品详情页面的静态页面会重新生成。

9.集群环境下的登录问题(单点登录)

原理是不再使用tomcat提供的Session，而是使用redis来模拟Session，一旦用户登录后会生成一个token，类似于jsessionid的功能。访问redis获取用户信息时，key就是token，value就是用户信息。然后把token信息写入cookie中，并且实现cookie可以跨域访问。当需要对用户身份进行验证时，需要先从cookie中把token取出，然后根据token查询redis，来判断用户的登录状态。

10.购物车是如何实现的

当用户未登陆时，将购物车信息放到cookie中,并设置存活时间，即使关闭浏览器，购物车也不会消失，除非用户自己清理浏览器中的cookie。

当用户登陆时，将购物车信息放到Redis缓存中，下次登陆时，即使换了浏览器或电脑，购物车也不会消失。

11.如果换电脑,购物车是如何实现的

如果把购物车放到cookie中，更换电脑是不能看到购物车信息的。如果想更换设备实现购物车同，需要把购物车信息保存到redis中。

实现购车商品数据同步：

1、要求用户登录。

2、把购物车商品列表保存到数据库中。推荐使用redis。

3、Key：用户id，value：购物车商品列表。推荐使用hash，hash的field：商品id，value：商品信息。

4、在用户未登录情况下写cookie。当用户登录后，访问购物车列表时，

a) 把cookie中的数据同步到redis。

b) 把cookie中的数据删除

c) 展示购物车列表时以redis为准。

d) 如果redis中有数据cookie中也有数据，需要做数据合并。相同商品数量相加，不同商品添加一个新商品。

5、如果用户登录状态，展示购物车列表以redis为准。如果未登录，以cookie为准。

12.订单号是怎么生成

所以我们采用redis的incr命令来生成订单号，每次加一。由于redis是单线程的所以使用此命令不会出现重复的订单号，并且可读性好。

13.秒杀抢购时库存是如何控制的

把商品的数量放到redis中。秒杀时使用decr命令对商品数量减一。如果不是负数说明抢到。一旦返回数值变为0说明商品已售完。

14.异常是如何处理

系统中的异常使用springmvc的全局异常处理器来实现，在系统中发生的异常都向上抛出，最后由表现层的全局异常处理器捕获，捕获异常后会写错误日志、向相关负责人发邮件发短信。然后展示一个友好的异常页面。