

一、简介

背景

技术角度：高可用，高并发

市场角度：流量，促销，用户体验

说明

注意点：

脱离原站部署（防止流量冲垮原始站点，大流量进入，导致原站本来业务崩溃）

预测（流量大不大，多少流量，人员分布什么位置，人员大概上网习惯，能够带来流量是什么样子，这就是预测数据）

测试（有了预测数据后就能进行测试，不说cc和ddos攻击测试，自己写个高并发脚本做个简单测试，也可以的让测试人员测试）

监控（让运维做好监控，万一你的预测和测试都通过，但是你实际跑的过程中有友商来捣乱或者流量超预期，怎么办，最坏的打算就是秒杀活动失败，但是要控制最大影响）

关键点：

1. 程序员的关键点

- a. 高可用:双活 但是注意阿里云不支持双活(绝大部分用keepalive-基于两个机器去竞争一个固定IP，但是阿里有自己的负载均衡机制)
- b. 高并发（将一个机器的任务分给多个机器干）：负载均衡,数据过滤(减少服务器响应)及安全

设计思路：

- 1. 静态页面：cdn（有效减少服务器自身的压力），真是页面隐藏（不能用前端隐藏，防止被破解暴露真正的商品操作页面，可以使用crontab定时替换网址），页面压缩（压缩太狠服务器压力大，不压缩用户压力大），页面优化（图片统一大小优化，css,js），缓存机制

2. 动态页面：排队，异步，资质抢购（先抢资质，抢到后再去买商品）

辅助建议：

1. 可以百度一下
2. 建议云服务器

二、设计思路

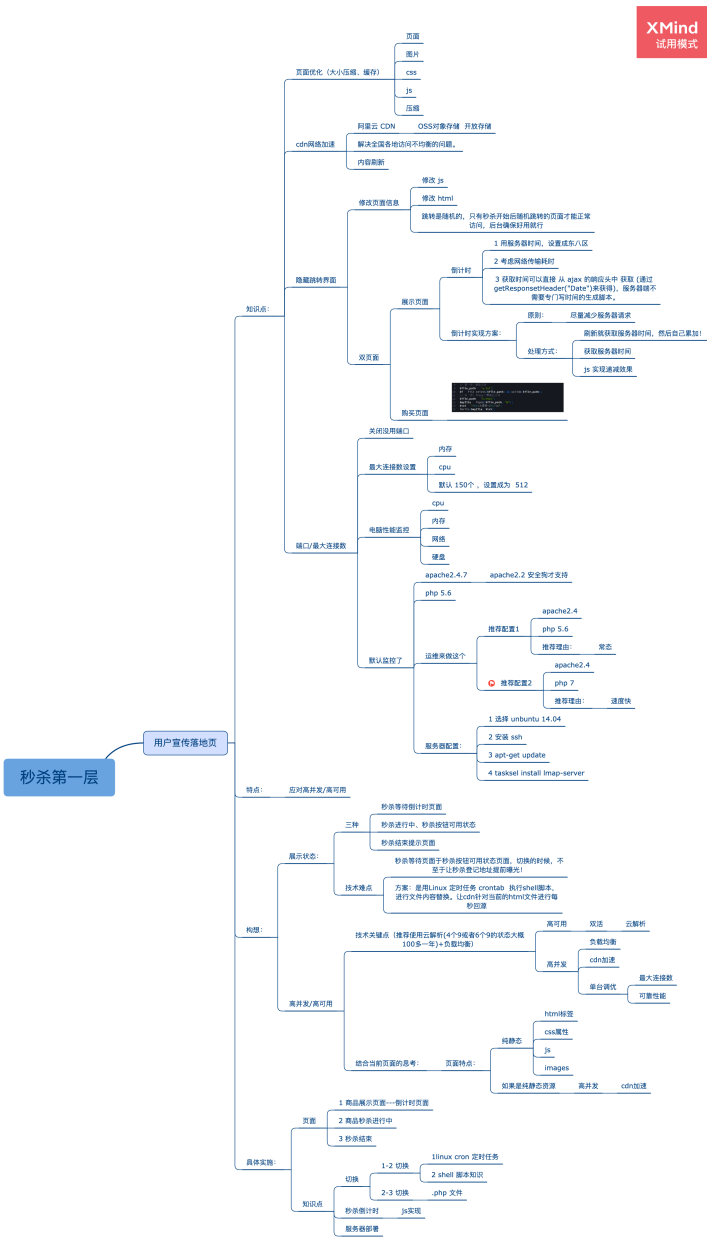
设计分析

1. 用户：量大，坏人（一种是想多秒杀点商品，多个位置登录【此情况可以用数据清洗来处理】。一种是执行攻击，让你秒杀站不能访问，此情况不在此考虑之内，又web安全及其他的进行处理）
2. 地域：全国/地区（cdn做全国性加速）
3. 业务流程：前台、后台

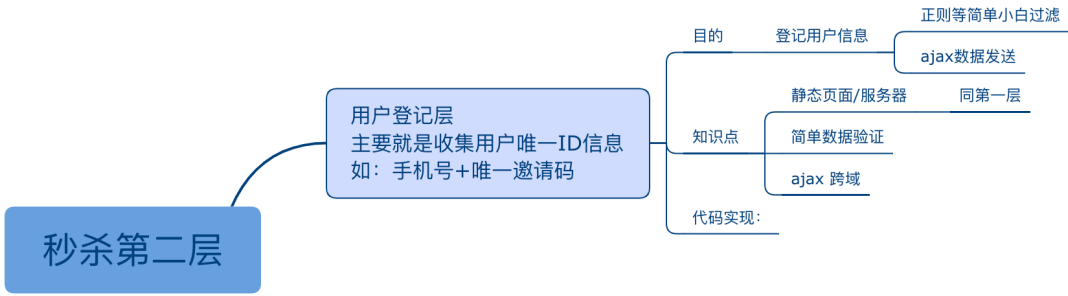
设计分层

1. 第一层：商品展示层（只负责商品的展示）
2. 第二层：用户登记层（让你的用户登记相关信息）
3. 第三层：数据接入层（把你用户登录的信息接入到处理机制里面）
4. 第四层：数据处理层（对接入层没有问题的数据存入到相应的数据库，并且做相应的业务处理）

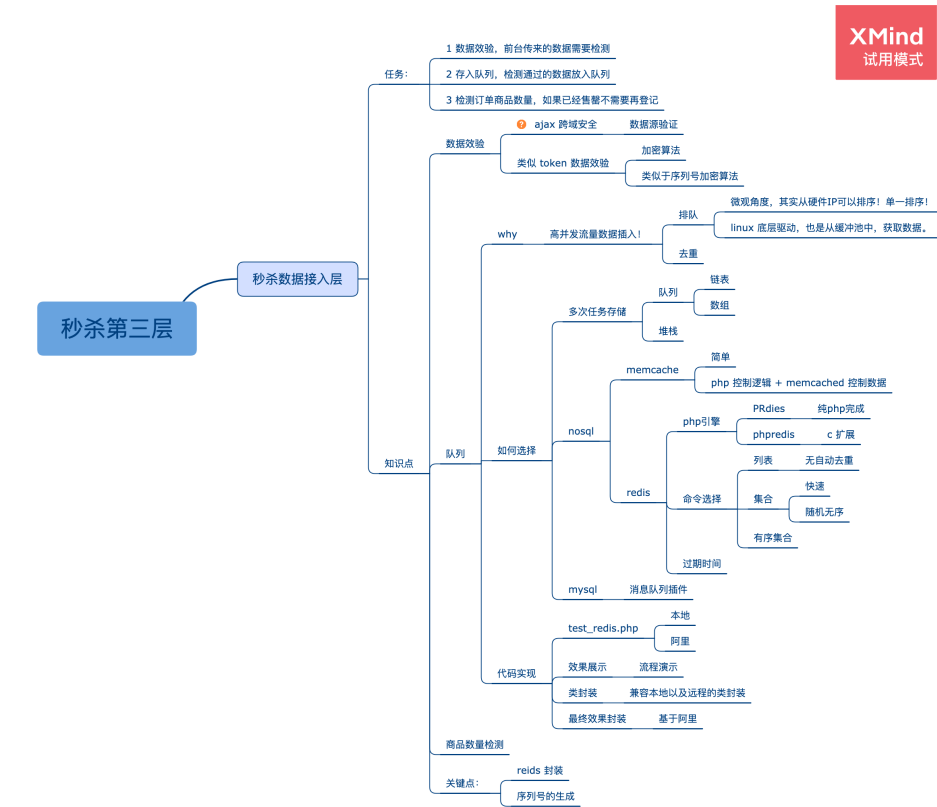
秒杀架构流程图



秒杀第二层



秒杀第三层



秒杀第四层

秒杀第四层

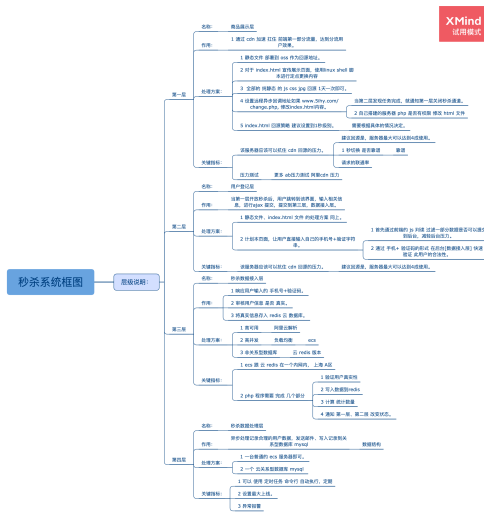
任务 转存nosql数据到 mysql里面，也就是持久化

知识点 无

数据处理层

代码实现

倒计时时间使用服务器时间



实例名称		输入实例名称模糊查询		搜索		
实例ID/名称	监控	所在可用区				
<input type="checkbox"/> i-uf63ds56dm38w5xaanu level4	  	华东 2 可用区 A				
<input type="checkbox"/> i-uf63ds56dm38w5xaanv level3	  	华东 2 可用区 A				
<input type="checkbox"/> i-uf6f244t8ycx6un19wa cdn_back_server_level1	  	华东 2 可用区 A				
<input type="checkbox"/> i-uf65fjxcg2d04d6elo6 webserver_level2	  	华东 2 可用区 A				
<input type="checkbox"/> i-uf65fjxcg2d04d6elo6 webserver_level2	  	华东 2 可用区 A				
<input type="checkbox"/>	启动	停止	重启	重置密码	续费	按量