



链滴

springboot+ 中间件实现秒杀系统感悟

作者: [sirwsl](#)

原文链接: <https://ld246.com/article/1610522437550>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<h2 id="故事">故事</h2>

<p>好久都没有写博客了，最近博客没写，基础知识没有进步，很多东西也没有深入去学，而自己的码实操能力与思维到时进步不少，为什么呢，因为自己一直在写毕业设计项目，本来自己只是想写一秒杀项目，结果由于秒杀只能算是一个模块，到时候毕业答辩好像不是那么容易，然后一鼓作气，好子，把后台管理，前端商城全给写了，硬是因为一个模块写了一整套的商城系统（也不能说是一整套还有支付模块没有写，由于个人无法回调接口）。事大概就这么一个事。</p>

<p></p>

<p>现在这系统算是 V1.0 基本结束了，应付毕设应该没什么问题了，接下来就是写写论文，然后去习心的东西，自己应该是不会再去动它了。毕竟在这套系统上花了很多时间，也学了一些东西，现在就是好好学习那些基础知识和看看那些源码。</p>

<h2 id="说说应届毕业生">说说应届毕业生</h2>

<p>作为一个应届毕业生，我真的无力吐槽，也深深感觉到，学习东西学校是交不了你多少的，要是指路人真的会很好。作为一个应届毕业生，虽然只是二本，但是我觉得很多一本或者 985/211 学校科生、乃至一些普本研究生，觉得真的，中国的高等教育，emmmmmm，大家自行体会，我不过多价。</p>

<p></p>

<p>不评价别人就来说说自已吧，自己之前学的成绩其实也是就一般，但是自我感觉动手能力不错，国家级、到院系级奖状拿了 30 来个，然后专利、立项加起来也 7、8 个吧，然后成绩中上，也就是样的我，nnd，到了大二结束也就是 2019.6 的时候我只是直到 servlet、JSP，然后会用 java 写一些小的算法。然后哎，我就不过多说了，现在一回忆，觉得自己大学学了个屎 :hankey:，啥也没有学会</p>

<p>然后到现在的 springboot、锁机制、缓存、数据库、中间件、多线程全 TM 是 2019.6 以后学。（这里不是鼓励大家不学习，学校的知识也很终于）虽然说学校没交什么实际性的，但是为什么很多企业招人很少有非科班的，有些企业甚至排斥培训班出来的，这就是大学课程的必要性，他会将好多基础知识原理交给你，而这些东西在实操的时候会让你更加炉火纯青。</p>

<p></p>

<p>其实在高校环境是参差不齐的，就拿毕设来说，我的系统已经完工，有的还在开题，所以明白自己想要什么，然后朝着这个方向努力，很多企业招人时候指出要双一流或者 985/211，这其实没有错，是虽然学校一般，并不代表人家就会否定你，但是不可否认中间确实存在很大的差别，因为我之前面过 BAT 还有很多的大厂、随后虽然凉了。有的死在二三面有的死在五六面，但这很正常，毕竟差距确实存在，人家加面 1、2 面怀疑一下能力很正常，但是也就此其实与很多人比起来自己其实打败了很多人但也确实存在差距。</p>

<p>这就是为什么自己写这系统加入了的东西：</p>

springboot+Mybatis-plus 搭建

redis、zookeeper、mq 集群

redis 缓存、jetCache 缓存

RabbitMq 消息通知、死信、延迟队列使用

zookeeper 分布式锁

自定义注解

统一封装返回

切面使用

设计模式使用

事物、回滚使用

docker、nginx 使用

图片服务器 OSS 使用

stram、lambda 使用

多线程、线程池使用

- 定时任务使用
- 短信验证、邮件服务使用
- JWT 验证 TOKEN 令牌使用
- 雪花算法分布式算法使用
- 拦截器、Log4j 日志使用
- 还有很多的工具类、组件使用
- 数据库连接池 Druid 使用
- MapStruct 实体转化工具使用
- 然后部署时候 CDN 之类的使用
-

<p>还有很多，不一一说了，之前自己想找一套系统，就像这次这样融入很多东西，但是一直没有找，这次自己写了，把能用的都用上了，也学会了很多，觉得小白或者是学生或者刚入职不久或者是一码哥们可以看看，我觉得对我收获颇多，对你应该也是。</p>

<h2 id="说说系统">说说系统</h2>

<p>这玩意，其实也不能顶太多的高并发，毕竟他只是代码实现，用的也只是 java，让代码来处理高发，好像有些吃力。之前看到的是采用 nginx+Lua 脚本，在 nginx 层上就处理，然后丢入队列，如有兴趣的可以自己改改，毕竟也很容易，加个队列直接掉 service 接口就好了，只是由于自己没接触 Lua 脚本，所以就没弄。</p>

<p>大家也可以把代码 copy 下来，然后自己搭建以下 redis、zookeeper 和图片服务器，在配置文里面填上对应内容就可以。代码仓库：GitHub：https://github.com/sirwsl/shopping-kill，觉得凑活的就给个星吧，毕竟也不容易，代码里面每一接口注释，和一些业务逻辑的注释打的都很全，也是为了方便大家阅读。也可以去码云上 git 一下，云仓库地址：https://gitee.com/sirwsl/ShoppingKill</p>

<p></p>

<h2 id="该系统适合什么样的人学习">该系统适合什么样的人学习</h2>

<p>说说该系统适合什么样的人学习吧，其实大佬都无所谓，毕竟我也只是写的毕设，和无数前辈来，这只是菜机作品，看这项目的人至少都有些 java 基础，因为里面用到了很多的 stream 和 lambda 达式还有些设计模式，要是没些基础，估计会很吃力。然后至少要用过 SSM，知道 Mybatis，里面久层用的全是 Mybatis-plus，然后需要知道 redis、mq 之类的原理。其他的也不需要知道什么了，代码不懂就百度或者 Google 吧。</p>

<p></p>

<h2 id="能让你收获什么">能让你收获什么</h2>

<p>总之这玩意还是让我收获很大的，因为之前自己写 demo，虽然说是前后端分离，但是并没有完全分离，直到自己写这套系统时候才真正的前后端分离，有的人看到项目，可能会说，这玩意没分离，实只是自己把前端文件放在了 template 下面，部署时候是拿出来，打包时候也没有给他打包在一，只是因为我自己开发，图方便罢了。</p>

<p>然后通过这个系统收获到的：</p>

- 明白了啥是前后端分离，如何做跨域，JWT 实现令牌登录
- 自己写了结果返回统一封装
- 对异常类进行了统一封装处理
- 对部分日志采用 Aspact 进行存储
- 直到了 redisTemplate 还有一些相关的使用和配置，（之前自己只是使用 jedis 进行操作，收获是很大的）
- 使用了 rabbitMq 进行消息分发、延迟队列、死信队列等使用
- 使用了短信登录、邮件发送的一些组件

- 还有 JetCache 的一二级缓存使用
- 切面、自定义注解使用
- 分布式锁使用
- 明白了 OSS 存储、CDN 转发、DockerFile 的一些使用
-

<p>还有很多就不细细说了，大家有兴趣可以自己去研究一下，再次放上项目地址</p>

<p>GitHub: https://github.com/sirwsl/shoppin-kill </p>

<p>码云仓库地址: https://gitee.com/sirwsl/ShoppigKill </p>

<p>项目演示地址: </p>

<p>商城展示: https://kill.wslhome.top </p>

<p>后台管理: https://admin.wslhome.top </p>

<p>大家可以自己申请账号,查看展示效果</p>