

**张嵘**

性格开朗，较好的团队合作能力，勤劳刻苦的精神，乐于学习

男 ︳ 25岁 ︳本科 ︳ 3年工作经验 ︳ 上海

15601596366 ︳[910818981@qq.com](mailto:example@exampleabc.com) ︳blog.csdn.net/boneix

说明: images/lagou-division.png

* **工作经历**

**上海佰晟通信息科技有限公司**

java 软件开发工程师 2017.03 - 至今

**南京途牛旅游网**

java 软件开发工程师 2015.11 - 2017.01

**南京夏恒网络系统有限公司**

java 软件开发工程师 2014.03 - 2015.07

说明: images/lagou-division.png

* **教育经历**

**南京三江学院**

本科 · 软件工程 2014年毕业

说明: images/lagou-division.png

说明: images/profile_square.png **项目经验**

**产品客服看板** 2015.11 - 2017.01

产品客服看板主要向客服提供旅游线路的快速查询工作。展示相应的线路基本信息、价格日历、库存、订单以及签证备注等信息。项目使用到ElasticSearch、Redis、ActiveMQ、类似dubbox的RPC框架

1.Boss3系统和价格中心系统通过ActiveMQ推送线路基本信息和产品价格信息，看板的查询系统接收数据并落地至数据库，然后再转发给看板的ES系统创建或者更新es索引对应的文档

2.为了提高接口响应速度，使用Redis缓存城市信息。减少对外部系统接口的频繁调用，同时提高了页面的响应速度

3.产品客服看板的查询功能使用ES的TransportClient实现，相比查询Mysql数据库，查询速度大幅提升

4.库存、订单及签证等信息需要实时性，库存、订单需要精确到天。造成一个页面依赖的接口很多，而浏览器是有并发连接数的。解决方案为：优先展示产品的基本信息；将相近的功能整合为一个接口提供给前台调用，后台通过向线程池加入调用多个接口的任务，以提高接口的响应速度

5.某个产品的某个团期的标签具备存活时间。如果通过Redis来做产品\*团期的数据量太大，不合适。分析业务的逻辑：后台操作数据库设置标签，再同步至ES对应的文档，前台查询的数据源为ES。所以解决的方式 同步标签数据至ES时做限制，只推送当前产品设置时间前一周的主推标签，接着查询时只显示今天以后的主推标签，添加定时任务推送符合条件的标签数据至ES

产品客服看板系统让我开阔了眼界，对互联网项目有了更加清晰的认识和理解。并认识到到性能对于项目、用户体验对于客户的重要性

**Boss3系统** 2015.11 - 2016.10

Boss3系统通过对资源配置，生成旅游产品线路，并提供给其他系统。

加入Boss3系统的开发工作，对于当时刚入职的我，能够加快了解项目的业务结构，开始时主要维护旅游线路基本信息。后面开发及维护推送产品基本信息至产品客服看板的功能。

**网站外包** 2014.03 - 2015.07

这段时期技术主要是那个快淘汰的JSP。刚入毕业那会儿，对于程序员的工作并没有清晰的认识，更别提代码规范、性能等等，期间遇到过很多问题，当时也不知道如何解决、甚至不知道产生的原因是什么。唯一解决的方式就是重启服务器、恢复备份数据。当时遇到的问题有：

1.一个tomcat部署了多个应用，如果一个项目内存溢出，导致整个tomcat挂掉。现在的解决方案是尽量分开部署项目，查找oom的原因，是优化代码及项目结构还是设置启动参数

2.数据库被插入脚本数据，导致服务器被黑。。现在的了解到是SQL注入攻击、XSS、CSRF等攻击手段造成的。

说明: images/lagou-division.png

说明: images/profile_square.png **自我描述**

在三年的程序员工作过程中，技术上的有了很大的进步，养成了学习新技术和新思想的习惯，工作上很勤勉，按时完成需求任务，bug率也比较低。乐于和同事交流，能够听进同事的建议，并结合实际思考可行性。以前对日志并不重视，在和测试经理合作过程中，逐渐发现在测试过程及线上定位问题是能够起到很好的作用，同时他也建议我要及时整理技术及业务文档，以后遇到类似的问题能够快速解决。遇到技术难题时，首先查找相关资料，找出可行性方案，然后咨询项目经理，向他请教其中可能存在的问题及风险，最后改善自己的方案。

在学习Spring源码的过程中有尝试将其部分思想用于实际的开发过程中，在使用阿里云MQ时，有同事Listener提出switch处理topic，if处理tag 在代码中不易维护。联想到Spring中有个NamespaceHandlerSupport，大致逻辑是继承类向Map中put解析器，在解析xml时通过key去拿解析器来解析，其中的设计模式包含：接口-适配器模式、抽象工厂模式、策略模式及模板方法模式。进行改造后Listener部分的代码很简洁，对应的维护只需要新增继承类和配置解析器即可。平时公司内部有技术分享，积极参加并结合自身实际情况，有选择的学习相关内容。最近在学习Python，各种语言有各自的优势，作为一个程序员涉猎多些编程上会有许多便利，虽然语法上有不同，但是其思想有些是相似的。

作为一个程序员，技术能力是立命之本，同时对需求的理解转化能力也很重要。逐渐尝试将产品经理提出的需求结合技术环境转化为具体的接口功能，基于外部环境的考虑对此进行一个可靠的排期，从而对即将开发的项目能有一个大局观。