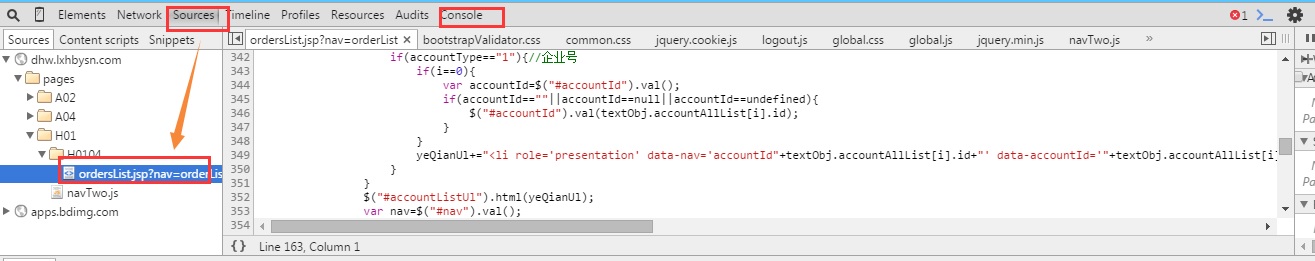
# 工作总结--如何定位web系统前后台的bug，以及bug分析/测试感想

<https://blog.csdn.net/weixin_30750335/article/details/98579086?utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EOPENSEARCH%7Edefault-4.control&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EOPENSEARCH%7Edefault-4.control>

对于web项目前台和后台bug定位分析：  
**一. 系统整体了解**  
懒企鹅营销服务平台用的架构：  
**web前端: Bootstrap 3.0 组件丰富,兼容性好,界面美观**  
**Server端: jsp+Servlet+json 公司技术力量储备丰富,技术成熟,有很多成熟的模块可以直接使用**  
**数据库: mySql 免费,相对成熟**  
前台: 涉及到jstl，jsp，js，css，html方面比较多  
后台：servlet，jms，ejb， 还有很多框架，struts，hibernate，spring，ibatis等，我们用的是Struts和spring框架，shrio控制权限  
**jsp分不清前后台的，因为这里涉及到一个运行时刻的问题，它们的运行时刻是不同。**  
**用户发出请求后，服务器解析用户请求，转至对应的jsp，这个时候可以说是整个jsp都是后台程序。**  
**而Jsp做出响应后，把响应的内容返回给浏览器，这个时候浏览器就只看见html,css,javascript，这个时候所有的程序又都是前台程序。**  
最近的锁行业服务项目总结：

互联网金融软件：营销服务平台的锁服务行业  
锁行业：  
**1.多用户（对权限）**  
锁匠：只能发布产品和发货，派单自动派自己  
锁厂：发布和发货，派单由派单管理员进行  
派单管理员：只负责派单，但是所有权限都有  
用户：不使用系统，但只能看到拉取自己的锁匠发布的产品和锁厂发布的  
**2.多平台（手机端，电脑端，各类微信公众号）**  
服务号：销售产品  
企业号：提供服务，内部成员使用，抢单和派单  
系统：提供产品和订单管理的服务平台  
**3.关联性强：**  
类别 ---产品---产品订单---服务订单，删除任一个（如产品），导致列表list加载错误，主外键约束（不能删除），前台删除，查找关联信息时为空，但是要求不为空，就会报错：加载失败！！  
解决方法：  
方案1：加上二次校验，删除时进行提示，有相关联的数据，不能删除:  
方案2：删除后把相关联的改成未分类！！  
**4.流程复杂：**  
发布产品---权限控制  
购买--付款--发货--收货--上门服务--服务完毕--评价--完成  
购买--未付款前取消订单，发货后不能取消订单  
开始服务要在发货后才可以，评价只能在服务结束后

**二.前后台bug定位**  
**1. 前台的bug通常是功能、界面和兼容性等有关。**后台的bug与性能和安全性有关。  
前台bug定位：按F12在console中查看报错信息，对于出错的js可以在Sources下查看对应报错的资源文件，写入禅道提交给开发即可



前台bug注意以下三个方面：  
 **（1）网站前台的权限控制：**没有权限的用户是不能直接输入url的方式来进行访问的，必须进行登录。以后涉及到权限的测试，一定不能漏掉url的方式也需要验证一下。而在单个页面进行W3C测试时则需要去掉该权限控制。  
**（2）网站前台的title，**对于这个也很容易忽视。进入到不同的功能页面，title显示应该是有，并且要和你进入的页面一致。title就是在浏览器最左上角看到的那些文字  
**（3）http和https的注意点**：https是一种安全链接，它是需要证书的，而http就是普通链接，所以在你的系统中客户会要求某些关键的地方希望加上这种安全连接，那么此时你需要注意的是，对于不需要的安全链接的地方千万也要去重点测试，有些开发会很容易忽略这一点。  
你要打开HTTPS开头的网站，前提是该网站安装了SSL证书，只有安装了SSL证书的网站，并且开启了443端口，你才可以通过HTTPS加密协议无访问。如果没有则不能访问。  
你可要测试，比如在某个网站http协议后面加个s去访问，看能否访问成功，能成功，会显示绿色安全小锁，否则就不能访问。给你举几个安装了ssl证书，可要https访问的网站，1号店，天猫，淘宝，支付宝，百度，沃通CA，工信部网站等等

**2.后台bug定位：根据后台日志文件**  
本系统是使用secureCRT进行日志获取，或者服务器控制方面的操作（关闭和重启）  
重启的一般情况：  
**（1）热部署 （新增部分功能，或者修改部分bug) （2）发布新版本 （整个系统）（3）内存溢出，此时重启服务器即可**

由于项目中有线程程序，./shutdown脚本关闭tomcat程序，不能把启动的线程全部关闭，造成服务器启动线程未关闭的错误,所以  
Linux系统中重启Tomcat的一般步骤：（一般是先关闭进程，然后进行重启 ，如果 /要删除某个文件：rm 文件名，或者不为空的文件夹：rm -rf 文件夹名）  
**cd usr/local/         //测试服务器名称/bin**  
**ps -exf                 //看测试服务器下运行的项目的主进程（最前面的数字为PID进程号）**  
**kill -9 PID            //强制关闭某一项目的主进程**  
**./startup.sh          // ./\*\*.sh 即执行重启shell脚本文件 ，此时在测试服务器的bin下面，直接执行即可，其余的加上 chmod a+x shell脚本文件，也可用./执行**  
（小知识：  
ps aux和ps -ef命令区别  
ps aux 是用BSD的格式来显示 java这个进程  
显示的项目有：USER,PID,%CPU,%MEM,VSZ,RSS,TTY,STAT,START,TIME,COMMAND  
ps -ef 是用标准的格式显示java这个进程  
显示的项目有：UID,PID,PPID,C,STIME,TTY,TIME,CMD）  
**3.如何查看日志？**  
一台服务器可以部署多个应用，  
**cd usr/local/测试服务器名称/logs               //查看先进入到服务器的logs目录下**  
**tail -f catalina.out                                    //监视catalina.out 文件的尾部内容（默认10行）**  
**4.日志中常见的问题**  
获取日志文件中常遇到的问题：  
**(1)编码问题**：tomcat是新的，需要改编码 修改tomcat的server.xml文件<Connector port="8080" URIEncoding="UTF-8"/>  
特别是windows下的项目重新部署到linux系统下，  
**(2)空指针：**程序问题，一般没有考虑到为空情况，或者主外键约束的数据为空，或者删除关联数据，导致为空  
**(3)长度过长**，超过最大长度，测试环境修改数据库字段长度后生产环境未修改，导致报错！！  
**(4)非法数据：**  
**(5)内存溢出：**重启

**5.一般的问题原因总结：**  
**程序：**为空判断，增删改查,不同公众号调用的接口也不一样  
**数据初始化：**数据库表结构和数据初始化，权限配置，

特别注意生产环境上的用户数据修改，此时用户在使用，很重要！！！服务号一个月4条，全部使用掉了！！

故障无法重现时：  
（1）看日志，根据日志定位原因，则在测试环境中按照日志提示构造条件相同的测试案例测试，尝试在测试环境中将问题重现。问开发  
（2）测试环境和配置与实际的工程环境和配置有哪些差异等等。同时主动与开发负责人、工程实施人员以及有经验的项目经理讨论，分析可能导致的原因。

配置环境不一致导致！！  
测试环境ok，生产环境新增时保存失败，查看后台日志报长度溢出，数据库内容字段要求和生产环境不一致！！

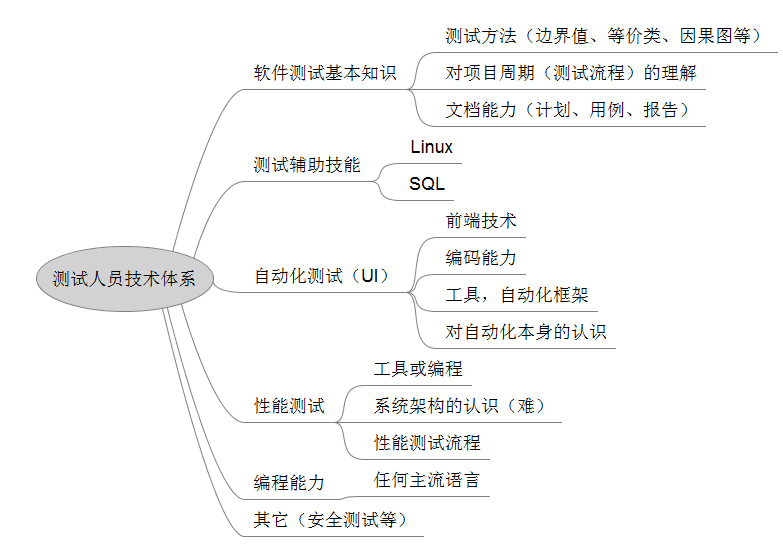
**6.辅助工具：linux和SQL**  
linux查看日志  
SQL用来筛选数据或直接进行数据修改状态，多用于集成测试过程中前后流程相连接，发布产品--下单--付款

**三.浏览器兼容性和网页规范标准测试**  
**浏览器兼容性测试**（偏主流浏览器，如谷歌，火狐，IE8以上）：  
https://dev.windows.com/en-us/microsoft-edge/tools/staticscan/  
**W3C网页验证：**（判断网页书写是否符合规范，记住此处必须去掉权限控制，单个单元页面url需要跟参数）  
https://validator.w3.org/  
**W3C手机端页面检测（**如手机微信菜单下的页面）：  
https://validator.w3.org/mobile-alpha/

另外结合我平时的工作和前不久看到的一盘文章，和大家分享一下**互联网测试与传统测试有什么不一样**。  
**1.**最大的不同：**互联网产品需要自己部署和运营，用户使用瘦客户端**（浏览器，app或一个需要安装的client)，核心的数据和业务逻辑在互联网公司的机房，在IDC，在云端。如：我们做的系统用户只需一个浏览器，服务器用的阿里云，部署和运营只需要一个运维人员即可。  
考虑现网（生产环境）存在下面两个问题：  
**（1）如何发布功能到现网**  
互联网测试完一般可直接发布，测试周期短，有时候需要进行灰度发布，先让部分用户用起来，发布完做生产验证。  
**（2）如何保证测试环境和生产环境同步**  
测试环境比较难搞，拿我们做的懒企鹅来说，牵扯的系统平台比较多，用到很多微信平台的接口，这个很难自己搭建或者用mock。另外保证测试环境和到生产环境都是好的，需要代码和数据库，以及环境配置都要保持一致，这需要相应的机制和工具来验证和同步。  
**2.互联网产品节奏很快**  
之前在的软件公司，基本是进行二次开发，周期长，每次都需要经过下面几个完整的测试流程：  
客户提出需求--BA和客户沟通，确定出需求和解决方案--测试人员根据需求说明书和解决方案编写测试用例---进行概要评审--进行详细设计评审--开始测试--回归测试--生产验证。  
现在的互联网产品测试基本为：  
产品经理确定好测试需求--开发人员写详设-（此阶段可以进行设计bug检查）--开发人员开发--测试人员测试，上线  
来不及测试设计，来不及自动化，短时间内如何保证测试的覆盖率和质量？--（探索式测试应势而生）  
**3.更多的人参与到测试中**  
互联网公司有专门的测试团队的比较少，一般开发和测试比例： 7:1，如何保证质量？  
（1）开发人员的自测  
开发人员进行单元测试，测试用例的通过率，同一个版本拉代码的次数  
（2）产品或运营人员的体验  
在这里基本我就相当于用户，进行产品体验，或者根据免费试用者反馈的意见进行优化  
（3）发布之前的评审  
注意环境，配置，数据方面的问题  
**4.有一些是免测试的**  
并不是所有发布到生产环境的东西都需要在测试环境检验，如：图片样式改动，小bug修复，但是哪些免测是个复杂的问题  
**5.海量用户带来的挑战**  
（1）性能方面  
如何做轻量级的性能测试  
（2）浏览器的兼容性  
例如：实习时测试的中国移动的CRM系统只针对IE浏览器，所以对浏览器兼容性没有太大的要求  
但是现在的系统大多基于主流的火狐，谷歌，IE6以上，放弃浏览器兼容就等于放弃一部分客户。  
**6.测试工具和技术方面**  
传统的企业花钱购买商业软件，如QTP，loadrunner，或者自己开发的项目管理工具 DMP  
大部分的互联网公司使用开源或自主研发的，如 selenium，appium，robotium,monkeyrunner

相同点：  
1.都需要非常熟悉产品和业务  
2.都需要了解产品的技术（深度测试方面性能分析，内存泄露，web服务器，cache，代理)  
3.具体的测试技术  
4.测试设计的方法

所以基本的大方向是没有变化的，还是技术至上，附一张虫师关于测试人员的技术体系共勉！



1. 如何定位Bug是前端还是后端问题（工作中个人总结)

如何定位Bug是前端还是后端问题（个人总结)



一、**UI界面出现问题**

**如果是APP方面UI的问题**—APP相关探查工具

**如果是WEB浏览器方面的问题“**

1.**查看前端UI**实现代码是否报错

2.查看对应API接口响应是否异常（接口测试工具：Fiddler、postman、浏览器F12开发者工具）

**二、服务层界面出现问题**



通过Linux相关命令，**查看对应服务所在日志，是否报错**

命令：

1 当日志文件名没有被切换，可以使用这个方法一直查看文件的实时日志。

**tail -f 日志文件名**

2 当日志文件名被切换到备份文件，同时产生新的日志文件名，这时候使用大写F参数来保持最新实时日志查看。

**tail -F 日志文件名**

3.当我们需要分析日志的时候，**会使用关键字寻找相关的线索**，那么我们肯定会用到grep命令。

但是所在关键字的行不一定完整记录用户的行为，这时候你就需要看关键字附近的上下文日志，比如抛异常的日志，那么怎么截取我们想要的日志呢？

3.1 根据关键字锁定满足条件的行数。

**grep -n "日志关键字"  日志文件名**

3.2 第一步查出某个符合的日志行数是1000行,想要这一行和他后面100行。

  sed -n "1000,1100p"  日志文件名

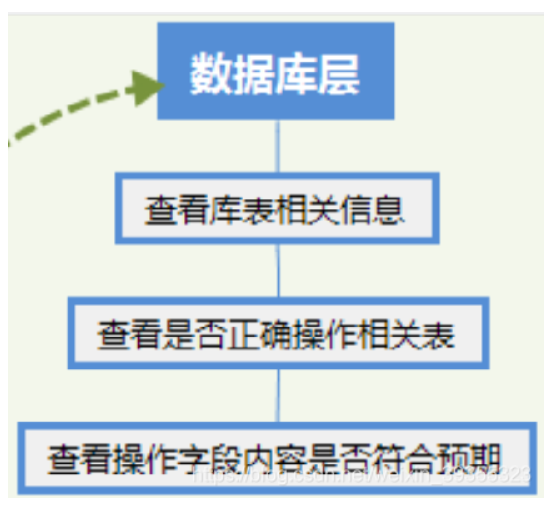
或者grep用以下参数控制上下文的显示：

**grep -C 10 keyword 日志文件名** 显示日志文件中匹配keyword字串那行以及上下10行

   grep **-B 10** keyword 日志文件名 显示keyword**及前10行**

   grep **-A 10** keyword 日志文件名 显示keyword**及后10行**

**三、数据库相关存储数据出现问题**

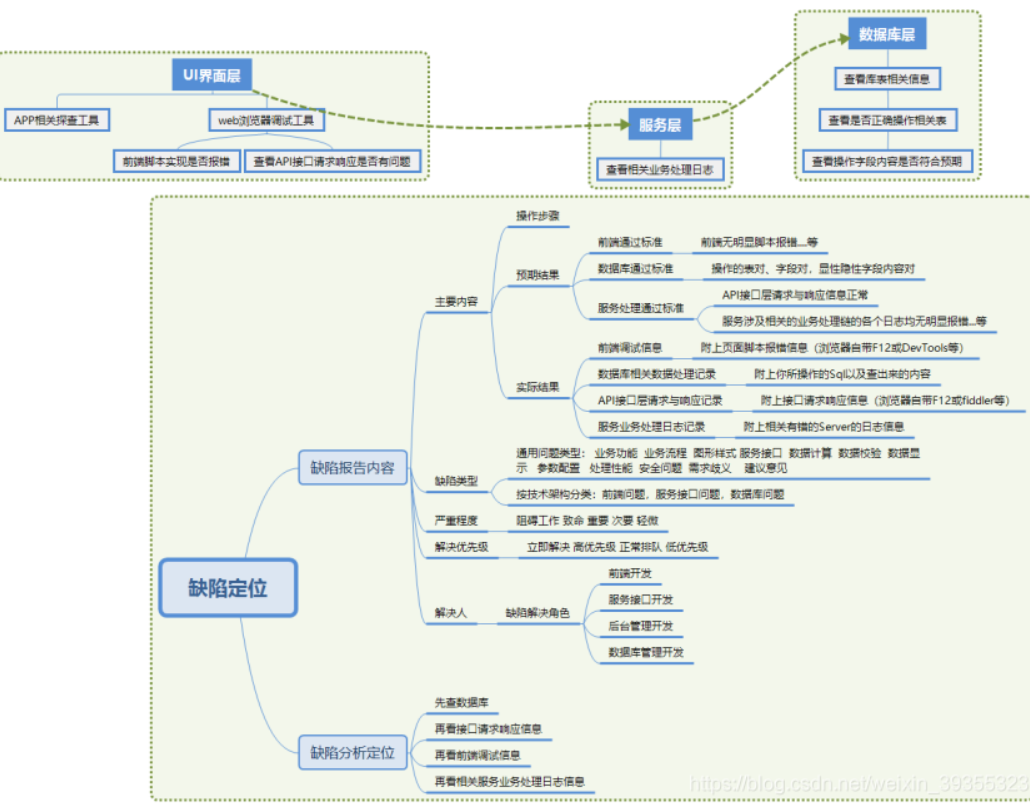


1.查看对应数据的数据库、数据表相关信息是否正常对应、存储、显示

2.对相关数据的操作是否正确

3.操作字段内容是否符合预期规则

四、缺陷定位方法总结与缺陷报告内容



1. **发现一个bug如何定位是前端还是后台问题？**

***1、经验法***

软件测试人员应不断精进自己的技能，负责的项目多了，自然对功能的实现过程有了解，也就明白如何分类bug了。 例如： 网页上的某个图片的分辨率不对，如果我们了解实现过程，可以想到一般情况下，是根据某个地址去服务器取图片的，数据库一般只保存地址，那么图片能正确显示，就说明后端的基本功能是满足需求的。如果具体图片分辨率有误，最可能的原因是前端显示过程出了差错。

2、**查日志**

当我们发现一个bug，并不确定这个bug属于前端还是后端，**可以查看后端服务的日志，复现bug时，查看日志中有没有相关信息。基本可以认为，如果日志没有输出，很可能这个功能并没有与后端交互，也就不存在后端的问题。反之，如果日志有输出，可以进一步查看有无错误日志信息，进一步分析。**

3、**查接口**

这种方法常用于查看是**后端返回给前端的数据有误**，还是**前端显示有误**。 大多数浏览器都有自带的**接口查看工具**，如Chrome，FireFox等都可以通过F12开启抓包，**在NetWork中可以看到当前页面发送的每个http请求**。 我们需要对比通过后端接口拿到的数据和前端显示的数据，来确认问题出在哪里。如果数据错了，页面显示是错的，也是正常的，先从后端入手去解决。如果数据对了，但是显示错了，就需要问问前端的开发人员了。 沟通很重要 我们在定位BUG的过程中，最不能忽略的一个问题是和开发人员的沟通，有时候忙活半天，不如一问一答。经验和技术的成长也都离不开合理高效的沟通。 经验和小结 出现样式的问题基本都是CSS的BUG 出现文本的问题基本上都是html的BUG 出现交互类的问题基本上都是Javascript的BUG 其他问题先沟通，再定位

————————————————

版权声明：本文为CSDN博主「洋….」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/weixin\_52740271/article/details/110365910