### 软件测试的流程：

前期从需求阶段开始 充分了解需求文档 有产品 研发 测试开需求评审会议 一起讨论需求 有不明白的地方 或是不突出的地方 让产品进行讲解 充分了解需求之后 研发开始进入开发阶段 测试进入编写测试用例 测试用例写完之后 如果时间允许的情况下进行测试用例评审 有研发 产品 会议评审测试用例就是用例没有覆盖不全面的进行补充 保证测试用例的覆盖率 我会用xmind编写测试用例把他的功能点一条一条列出来 还有禅道编写测试用例 测试用例写完之后 2. 测试准备阶段 准备测试的数据 还有搭建测试的环境 3 . 测试开始阶段 执行测试用例 还有比较bug 验证bug 我在测试阶段如果发现bug我会提交到bug管理工具里面 指派给相应的开发人员 如果研发不及时修改我会直接去找他和他讲明这个bug影响到那一步流程 让他进行修改 修改好后我在进行回归测试一下 就是对这个bug在进行测试一遍 如果bug已修改好 我在禅道里面在进行关闭bug 功能测试就是看看项目里面的功能点是否都可点击 测试完成之后在编写测试报告 验收测试就是对项目上线前的最后一次验收测试 就是把所以的流程全走一遍 没有问题的情况下 在进行上线 上线后我在进行测试一遍 如果发现bug后 我会及时的提交到相应的开发人员 让他进行修改。

### Windows / Linux系统中测试环境搭建：

Windiws下搭建测试环境：

首先我先官网下载jdk和tomcat

先安装jdk,选择安装目录进行安装 安装jre,直接安装

配环境变量,点击计算机右键属性,点击高级系统设置,找到javahome:jdk的安装目 录;path:jdk下可执行文件的目录;classpath:java运行时需要查找class文件的目录.

打开cmd命令提示符窗口,输入java -version查看jdk的版本

安装tomcat,解压tomcat,在bin目录下找到startup.bat启动tomcat;点击shutdown.bat 来关闭tomcat.

点击启动之后,在浏览器中输入http://localhost:8080是否能进入到tomcat页面

Linux下搭建测试环境： 先安装jdk 再安装tomcat

1 、su 先进入root权限当中，才能再操作其他程序

2、输入密码

3、ifconfig

4、连接ssh

5、cd /usr切换到usr目录下

6、在usr目录下新建 mkdir azb

7、接着去ssh把jdk 、tomcat上传过去。

8、去命令窗口azb目录下输入rpm -ivh jdk

9、接着是返回 cd

10、进入cd /etc目录下

11、输入vi profile 然后咱们接着输入i开始配置java\_home、classpath、path=$path三 个环境变量。

复制这一段话在done下面

export JAVA\_HOME=/usr/java/jdk1.8.0\_91

export CLASSPATH=.:$CLASSPATH:$JAVA\_HOME/lib:$JRE\_HOME/lib

export PATH=$PATH:$JAVA\_HOME/bin:$JRE\_HOME/bin

【注意：jdk 的名字必须你的jdk 的名字，不能直接复制粘贴不看jdk版本】

12、soure /etc/profile 立刻执行的意思

13、java -version 查看jdk版本，或者是java、javac查看咱们的jdk版本。

再安装tomcat

1、azb下解压tomcat

unzip tomcat.zip

2、cd 返回根目录

3、cd /etc/sysconfig

4、vi iptables 注意一定要在root下面。

5、i 编辑 复制一段话添加端口号

-A?RH-Firewall-l-TNPU-m?state?--state?NEW?-m?top?-p?top?--dport?8080?-j?ACCEPT

6、Esc :wq!强制保存并退出。

7、给bin 授权

a 、在cd /usr/java/tomcat/bin

b、输入 chmod +x \*.sh

ls查看授权是不是成功了。

8、service iptables stop

9、启动tomcat sh startup sh

10、如果不行的话在数据库bbs下复制这段话并且运行一下。再杀一下进程号，再重启 tomcat.

GRANT?ALL?PRIVILEGES?ON?\*.\*?TO?root@"%"?IDENTIFIED?BY?"root"

ps -ef|grep tomcat 查看进程号

kill -9 进程号 杀死进程

11、BBS1.0文件里面需要改成你自己的本地小黑窗口的IP地址

更改ip地址在 weppaps-bbs文件夹-classes-WEN-INT-beans.xml里面修改地址。

12、查看日志的命令：

tail -f ../logs/catalina.out 查看日志

### 测试用例的要素，编写测试用例的方法以及如何保证测试用例的覆盖率与正确性

**功能用例**:用例编号 所属模块 用例标题 用例级别 前置条件 输入数据 操作步骤 预期结果

**接口用例**: 项目 、模块 、用例id、接口名称、用例标题 、请求方式 、请求url地址、请求参数、前置条件 、预期结果、请求报文、返回报文、测试结果 通过/失败、测试人员

**APP测试用例:**用例编号，用例类型，所属模块，用例目的，前置条件，具体操作步骤，期结果，实际结果 测试版本 手机类型 手机系统

**性能用例:**用例编号，用例类型，所属模块，用例目的，前置条件，具体操作步骤 性能测试指标的来源 主要的性能指标 性能测试要点  预期结果，实际结果  请求数据 返回结果

**测试用例设计方法：**

测试用例设计方法是等价类、边界值，场景法和错误推断等。最常用的还是等价类划分和边界值法，等价类的划分可以让我们更全面的覆盖功能需求，避免遗漏，也能让我们用尽量少的测试用例来达到最好的测试效果，一般会划分有效等价类和无效等价类，然后在每个等价类中分别提取部分取值去设计测试用例。

边界值的使用主要是因为在等价类的边界部分最容易出现问题，所以要在等价类的基础上重点使用边界值法来设计测试用例。

**如何保证测点的覆盖率**

1.首先画xmind导图 保证功能点的覆盖 ,然后自己再补充。

2.最后召开用例评审。再补充

1、编写测试用例前，检查相关需求需求、设计文档是否有问题（功能描述不清，设计逻辑缺陷），如有问题找相关设计或者开发问清楚。

2、然后整理成需要覆盖的功能列表或者思维导图，功能列表包含新增和修改功能点，性能需求也要列出来（因为要整理对应的性能测试用例），同时还需要对既有功能进行一个梳理，检查是否会与其他功能有交互，整理出影响点。

3、把功能列表发给组员，并找时间进行会议评审，主要对功能等进行查漏补缺。

4、最后才行进测试用例编写，注意编写规范。

5、编写完毕后，把测试用例发给组员，开会进行评审，主要对检查点、用例规范进行查漏补缺。

6、执行测试用例过程中，发现用例不完善或者错误，需对测试用例进行及时的修改与调优

7、测试完毕后对漏测的bug进行测试用例补充。

### 简述测试工作中接触过的数据库

**接触过的数据库的话有** : Oracle和MySQL

**字符串的模糊比较**

MySQL和Oracle里都可以用字段名 like ’%字符串%’但是Oracle这种方法 不能使用索引，速度不快。

**对事务的提交**

MySQL默认是自动提交，而Oracle默认不自动提交，需要用户手动提交

**分页查询**

MySQL处理翻页的SQL语句比较简单，用LIMIt开始位置,记录个数

而Oracle则是需要用到伪列ROWNUM和嵌套查询

**分组**

group up在Oracle中后面要跟出现的所有字段名，MySQL中可以跟单独字段

**保存数据的持久性**

MySQL是在数据库更新或者重启，则会丢失数据，Oracle把提交的SQL操作线写 入了在线联机日志文件中，保持到了磁盘上，可以随时恢复

**最重要的区别**  
    Oracle是重量型数据库，适合用于不太及时响应并且数据运算量比较大的数据结构 查询，并收费，Oracle公司对Oracle数据库提供所有服务，

    MySQL是轻量型数据库，适合用于简单数据结构的查询，并且免费，没有服务恢复 数据

### +测试方案，测试计划文档中的内容

测试计划 : 1.项目简介2.参考文档3.测试范围4.时间进度安排5.资源6.风险 评估7.测试交付件

### Python中string常用方法有哪些？

str.capitalize()**用法：将整个字符串的首字母变成大写，其余小写**

str.upper() **用法：将字符串全部变成大写**

str.isupper()**用法：检测字符串中所有的字母是否都为大写**

str.lower()**用法：将字符串全部变成小写**

str.islower(**) 用法：检测字符串中所有的字母是否都为小写**

str.title()**用法：将字符串的每个单词的首字母大写，其余部分小写**

str.istitle() **用法： 判断字符串是否首字母大写，且其他字母为小写**（会忽视非字母字符）

### Python中元组和列表的区别？

列表和元组非常类似，列表可变，元组不可变，有时候他们都干一样的事情。他们最大的区别是元组一旦被赋值，值不可以被改变，一旦改变就会出错；但是列表可以任意的更改。其次的区别是他们用不同的符号表示，赋值的时候，列表用方括号[]，而元组用小括号（）

### 接口测试如何做的：

接口测试我们公司一般开发都先会（用postman）自测，之后再让我们测，有时候他们比较忙的时候，也会让我们帮着测。{测试时，我们可以用浏览器、postman、python或（jmeter）来做接口测试。Python用来做接口的回归测试、jmeter可以做功能测试也可做性能测试。}

当开发没有给我们接口文档只给了单个接口或少量接口让我来测试时，就可以用浏览器或postman（或jmeter）来测试。用浏览器测试时我们需要在地址栏中输入url路径和参数，查看返回的结果是否匹配。

（1）用postman测试时，当请求以get发送时，我们需要在params中以key value的形式添加参数，也可直接在路径后添加参数；当请求以post发送时，我们需要在body的x-www-form-urlencoded中以key value的形式添加参数；然后点击send发送请求来进行接口测试，返回的结果显示在下方的文本框中，我们可以根据返回的结果来判断接口是否正确。多个接口一起测试的话，我们可以在左侧功能区，找到collections栏目并点击新建的接口，然后在save里面选择 save as，选择刚才新建的集合，接着在上面找到runner,开始运行。选择咱们刚才新建的接口集合，点击start test运行，结果由红色和绿色来区分。后面有响应时间和响应码。

（2）用jmeter进行接口测试时，首先，我们需要创建一个线程组，接着选择sampler，再选择http请求，在http请求中填写服务器的ip地址，服务器的端口号以及要访问的url路径。当接口以get请求时，我们需要在地址栏下的parameters中以key value的形式传参；当接口以post请求时，我们需要在地址栏下的bodydata中以变量名1=值1 & 变量名2=值2 & 变量名3 = 值3 &.....的形式传参。若接口含有中文，我们需要将content encoding的属性设置为utf-8.若对接口只是进行功能测试，只需添加查看结果树，若右侧请求颜色为绿色则成功，红色则失败。若进行性能测试，还需通过扩展插件来增加jmeter的监听器，因为jmeter自带监听器只有3个（图形结果、查看结果数、聚合报告），要进行性能分析远远是不够的，所以需要扩展插件来添加每秒事务数、每秒点击量、每秒响应时间、硬件变化、吞吐量这些监听器。

（3）我们一般用python来做接口的回归测试，一般是在不太忙的时候将接口信息储存在python文件中。我们那时候做接口测试时，一般都是http请求的接口，用的是python3.6版本的，编译器是pycharm。首先我们需要再python上安装一个requests包，并且在python文件中通过import?requests来导入这个包，然后我们需要将每个接口的URL路径、参数、请求方式、名称通过字典以key value的形式储存起来，请求方式有两种，一种是get，一种是post，它俩的区别就是，如果要用post就在参数前面加一个data就可以了。接着我们需要定义一个方法来对这些接口进行封装，并通过requests.get传入url地址和参数，然后赋值给一个值，通过这值来获取响应报文和响应码，并打印，然后看返回结果请求值，成功是200,错误返回404,500等，检查返回数据的正确性与格式是否正常，我们开发那边经常返回的是json格式。当然，也有返回xml格式的或者是html格式的。最后我们再调用这个方法来达到测试的目的。需要注意的是，传参的时候一定要按照接口文档来写，要区分大小写，要不然就只会返回响应码而不返回报文！接口的性能我们也会很重视，因为直接影响用户体验，如果半天不返回值我就果断放弃，和开发那边沟通。我们就是这样来测接口

1. **简述通过Python如何做接口测试：**

我当时用的是pycharm。在网上找了一些免费的接口,我把需要的包requests安装上导进去，安装就是pip install requests,导进去就是import requests.用requests.get放进去url地址和传参，然后复制给一个值，通过这值来获取响应报文和响应码，并打印，然后看返回结果请求值成功是200,错误返回404,500等，检查返回数据的正确性与格式是否正常，我一般都是返回json格式。我对python测接口测试目前就了解到了这么多。

### 简述通过Python如何做自动化测试：

同样用的是Pycharm，就拿之前做过的bbs项目来说，浏览器用的是火狐，首先在火狐上下载一个Xpath插件用来获取html页面标签，还需要下载一个火狐的驱动来实现访问功能。

然后安装selenium插件，安装方法是在控制台进入到python文件里script文件夹下输入pip install selenium。

安装完成后开始编写代码 新建一个python项目 输入from selenium import webdriver 来调入这个库 然后通过import time 调入时间这个库

然后使用webdrive.Firefox调用火狐浏览器，给他一个变量比如是“a”就用a.get（）括号里输入要访问的域名，然后让他跑起来看看能不能访问成功。成功了就进行下一步通过内置方法获取页面的标签（by\_name，by\_id，by\_xpath能用哪个用那个）

我主要用的是xpath，右击想要获的代码 选择xpath 然后选择copy 复制到括号里如果是点击就是.click,如果要填写参数就是.sendkeys（）括号里写要填写的参数。

如果有弹框就需要用d.switch\_to\_alert这个方法 给他赋个值比方是“b”，接着就比赛b.accept就是点击确认按钮。b.dismiss就是取消，a.quit（）就是直接退出浏览器，a.close仅退出当前窗口。

### 总结在Linux系统中常用的命令：

1. 看日志：tail -f ../logs/catalina.out
2. 查看日志前100行：head -n 10 0 test.log
3. 查看日志后100行：tail -n 100  test.log
4. 关闭防火墙：service  iptables  stop
5. 开启防火墙：service  iptables  start
6. 查看防火墙状态：service  iptables  status
7. 解压两种方式：unzip  tar- xvf
8. 删除文件：rm -rf
9. 查看文件内容：cat
10. 授权：chmod + x \*.sh
11. 杀死进程：kill -9 进程号
12. 查看进程：ps -ef |grep 进程名
13. 重启虚拟机：reboot]
14. 查看用户登录历史：last
15. 显示磁盘的空间使用情况以及挂载的分区列表：df -lh
16. 现实工作完整路径：pwd
17. 查看所有文件以及隐藏文件：ll
18. 创建文件夹：mkdir
19. 复制一个目录：cp
20. 注销：logout
21. 启动tomcat: ./startup.sh
22. 关闭tomcat:  ./shutdown.sh
23. 安装一个tomcat：rpm -ivh jdk
24. 卸载一个jdk:  rpm -e--nodeps  jdk
25. 查看8080端口号有没有被占用：netstat -apn | grep 8080
26. 清空控制台：clear
27. 从日志中退出：ctrl +c
28. 强制保存并退出：wq！
29. 编译：make
30. 查看磁盘：df
31. 下载命令：wget
32. 防火墙永久性生效关闭：chkconfig iptables off
33. 查看内存使用量和交换区使用量：free -m
34. 查看路由表：route -n
35. **简述在Linux系统中如何实现 nginx + tomcat集群：**

当一台tomcat不能够承载请求数量，有两种解决方案：纵向扩展、横向扩展

纵向扩展：通过提升硬件设备来解决（成本高、有瓶颈）

加内存、升级cpu、加硬盘

横向扩展：通过技术来解决（成本较低、廉价、无瓶颈）

负载均衡、分布式部署

这两种方式，市面上普遍采用第二种（横向扩展）

首先

1. 根目录下分别安装

yum install gcc

yum install pcre-devel

yum install zlib zlib-devel

yum install openssl openssl-devel

注意：安装过程中遇到停顿 都选Y

1. 下载nginx的tar包

首先在cd /usr/local新建一个nginx 文件夹：mkdir nginx

接着在nginx文件夹中加入以下代码 下载安装包

wget <http://nginx.org/download/nginx-1.13.7.tar.gz>

下载完成后 解压该压缩包

tar -xvf nginx-1.13.7.tar.g

1. 安装nginx

首先进入刚解压好的文件夹：

cd /usr/local/nginx/nginx-1.13.7.tar

接着依次输入以下三个命令 进行 初始化、编译、安装

./configure

make

make install

1. 启动nginx

找到nginx 里的sbin文件夹：/usr/local/nginx/sbin

输入./nginx（两遍）

1. 在虚拟机浏览器中输入“127.0.0.1/”,若出现“welcome to nginx”则证明nginx安装成功，在真机中输入虚拟机的IP地址，若出现“welcome to nginx”则证明nginx安装成功。
2. 停止nginx

在/usr/local/nginx/sbin下输入

./nginx -s stop

**五、Nginx+Tomcat集群**

1关闭nginx

1）在/usr/local/nginx/conf中找到nginx.conf文件

2）用vi命令进入文档，找到gzip on 在下方加入以下代码：

upstream bbsjq{

server 127.0.0.1:端口号1 weight=1;

server 127.0.0.1:端口号2 weight=1;

server 127.0.0.1:端口号3 weight=1;

}

1. 接着在此代码下方找到server 中的location

删除大括号里面的内容，加入以下代码：

proxy\_pass [http://bbsjq;](http://自定义的集群名称;)

保存并退出

**注意：**weight 指的是Tomcat被访问的几率；代码后面的分号不要忘记写！！否则会报错

2修改Tomcat的默认访问页面

1. 在/usr/1904/tom1/conf中找到 server.xml文件
2. 用vi命令进入文档，在最后找到**</Host>前**插入以下代码：

<Context path="" docBase="/usr/12/tom1/webapps/bbs1.0" debug="0"/>

3）保存并退出

**注意：**docBase后面跟的是该Tomcat包里项目的路径

3做完以上步骤 如果nginx 页面报错，那就把nginx和所有的Tomcat包都重启一遍

1. **简述性能测试 压力测试 并发测试 负载测试的区别：**

性能测试：模拟用户负载来测试系统在负载情况下，系统的响应时间，吞吐量等。所以它的目的是为了获取系统的指标。

1. 性能测试: 指在多用户对服务器或者项目进行访问时，服务器的整体性能检测

压力测试: 在一定的软件硬件环境下，通过高负载的手段来使服务器资源处于极限的状态，测试该系统在极限状态长时间运行是否稳定。包括系统指标，服务器性能指标。

并发测试:在同一个时间点

负载测试：在一定的软硬件环境上，通过不断的加大负载来确定在满足性能指标情况下所能够承受的最大用户数。所以它的目的是为了获取最大用户数。一般不超过80%cpu，正常情况工作下最大用户数数据。

综上所述：

一般情况下我们所说的性能测试就是在服务器指标不超过80%下的测试来获取性能指标，负载测试是测试的一个方法，通过不断调试并发数获取性能瓶颈。比如80个并发，这个叫80用户负载测试。通过80—>180这样的并发数变化过程，就叫做性能测试。也就是说，性能测试是通过不同的负载测试来实现的。

压力测试，就是高负载的情况下进行的，目的不是为了获取性能指标，而是想要了解系统是否稳定。这时候服务器的指标一般不超过90%。压力测试通过长时间的运行较性能测试更能容易发现内存泄露的问题。

简单来说，负载测试是个方法，性能测试是一个过程。压力测试是个高压力下的性能测试

2. 压力测试: 指多个用户在一个时间段内对服务器进行访问，不断的增加人数，

直到服务器出现崩溃或者宕机。

3. 负载测试: 指多个用户在一个时间段内对服务器进行访问，不断的增加人数，

直到服务器不满足指标(服务器的cpu,内存，磁盘读写，网络读写等)为止

4. 并发测试: 在同一个时间点上同时向服务器发起请求

区别： 压力测试是要通过不断的增加用户量，直到服务器崩溃或者宕机，

负载测试只是需要测试出服务器在满足一定指标的前提下最大的访问量。

压力测试是测试服务器在某一个时间段中所能承受最大访问量，

并发测试则是测试服务器在某一个时间点上所能承受的最大访问量。

所以，综上所述，压力值 > 负载值 压力值 > 并发值

### 简述如何通过Jmeter做自动化测试：

首先我们使用BadBoy录制脚本

1.启动jmeter添加线程组。通过参数化实现自动化

2.为线程组添加元件和取样器

针对于http的简单应用，只需要添加HTTP Cookie 管理器，HTTP信息头管理器，HTTP请求默认值。

3.HTTP 请求默认值可以设置http请求的协议，域名，端口号。

4.三个元件添加好之后就可以添加HTTP 取样器了。

5.为线程添加监听器

监听器可以得到取样器中返回的结果，主要添加两种，聚合报告和察看结果树。

6.然后点击启动就可以了

### 简述如何通过Jmeter做性能测试：

先用badboy录制好脚本导入到jmeter中，比如登录功能，就在登录请求前设置同步定时器，设置好并发人数，当多个线程的用户到达该点后开始集结，达到集结人数后同时执行该功能，以获取并发情况下该功能的详细数据，对项目的性能进行评估推断 根据

1.接口文档中选择一条url放到http请求中

2.发送post或者get请求

3.调整线程数如100,500,1000

4.看图形报表，分析吞吐量

### 简述Jmeter如何做并发测试：

1、创建线程组 集合点集合策略

2、设置线程参数

3、网路请求也可添加cookie管理器  网站有登录限制的需要用到记录用户信息

4、添加HTTP请求，为录制http连接使用

5、可以添加添加HTTP代理服务器   （添加代理需要在浏览器中设置代理）

6、选择需要得到测试的报告样式

7、开始执行查看样式

8、停止查看返回的数据和请求是否正常 用表格样式查看

### 简述Jmeter的断言 脚本关联的操作步骤：

**断言：**

1. 首先在线程组上右击 选择“添加”再选择“断言”再选择“响应断言”
2. 在弹出的“响应断言”窗口中要测试的响应断言属性中选择“响应文本”在“模式匹配规则”属性中默认选择“substring”在要测试的模式中的文本域填写断言信息
3. 运行线程组 添加“察看结果树”在察看结果树的请求列表中则会发现 当请求的响应数据中没有匹配到断言的内容时 即使请求的相应状态码为200请求也被视为错误请求

**脚本关联：**

1. 首先在需要获取相应数据的请求上右击选择“添加”选择“后置处理器”再选则“正则表达式提取器”
2. 在弹出正则表达式提取器中自定义表达式提取器 在窗口中的引用名称填写引用该内容的变量名称
3. 在窗口中的正则表达式中填写正则表达式格式为（<标签>（.+?）<标签B> 属性=“属性值” value=“（.+?）”）
4. 在窗口中的模板属性中填写：$1$表示获取第一个获取到的信息
5. 在窗口中的匹配数字中填写：0表示生成的随机参数的多个结果时获取第几个位置作为返回的结果的下标
6. 缺省值不需要填写
7. 在需要做参数化的请求中通过EL表达式${引用名称} 来获取正则表达式提取器获取的内容作为参数

### 简述Jmeter如何做性能分析：

1. 图形结果: 图形结果监听器中显示的是服务器的偏离和样本数目等信息

2. 查看结果树: 查看结果数中展示的是每一条线程执行请求时的详细信息，包含请求的路径，参数，方式以及服务器端的响应信息

3. 聚合报告: 展示的是服务器响应的时间和容错率以及服务器吞吐量（接收和发送的数据量）

4. Bytes Throughput Over Time(单位时间内服务器吞吐量变化趋势图): 展示的是服务器发送和接收的请求的个数

5. Hits Per Second(每秒点击量): 展示的是服务器每秒接收的请求的个数（客户端每秒向服务器端发送的请求的个数）

6. PerfMon Metrics collector(服务器硬件监听器)：展示的是服务器硬件性能数据（服务器的cpu,memory,network,disk）

7. Response Times Over Time(单位时间内服务器响应时间变化曲线图): 展示的是所有请求的响应时间变化趋势

8. Transaction per second(每秒事务数)：展示的是服务器每秒中处理的事务

**测试基准:**

1. 硬件: cpu占有率不能超过70%
2. 内存占有率不能超过50%
3. 网络传输速率必须满足网络最低传输要求
4. 硬盘读写速率必须满足每秒读写的数据量
5. 响应时间: 所有请求的响应时间不得大于3秒
6. 服务器的吞吐量必须达到每分钟1300个
7. 请求的错误率（容错率）不能大于5%

项目型公司 产品型公司

博客网

虫师

Csdn