# **B简历**

# 一月份

## 自我介绍（基本信息，教育经验，工作经历）

大家好，我叫纪昀,来自山东曹县，毕业于山东省临沂大学，有2年左右工作经验，

在公司做过的项目,有Web项目APP项目, 像电商，医疗，旅行app等各种类型项目。

在平常的工作中我主要参与接口测试, 功能测试，性能测试，自动化测试等等。

接口测试的时候我用的是postman, jmeter；

性能测试就是Jmeter , LoadRunner，

熟练用python做自动化测试;

对于Linux系统下的项目，可以熟练运用Linux命令来操作，

可以配置nginx+tomcat集群,

能熟练用使用Oracle和MySQL数据库这些测试工具，

近期的一个项目是旅行app在这个项目中，我负责的是我的这个模块，首先我采用的是python+selenium来做的界面自动化测试，看里面的功能点都是否可以实现，测试有没有bug，就说里面有一个“我要直播功能”，填写的是个人信息，我就采用find\_element\_by\_id,name,xpath,等来获取信息，+.click .sendkeys()来填写参数等进行测试。

我个人的性格比较开朗，和同事的关系都能相处的很融洽，有上进心，乐于学习，以上就是我的个人情况。

## 测试的流程

1. 首先项目经理通过和客户的交流，完成需求文档
2. 由开发人员和测试人员完成需求文档的评审，评审内容有：需求描述不清楚的地方，有冲突的，以及无法实现的功能
3. 项目经理通过综合开发人员，测试以及客户的意见，完成项目计划
4. 开发根据需求文档完成需求分析文档
5. 测试人员进行评审，看是否有遗漏或是双方理解不同的地方
6. 测试人员完成测试计划文档，测试人员根据修改好的需求分析文档开始编写测试用例，同时开发完成两份文档，概要设计文档，详细设计文档，给测试作为补充材料，测试用例完成后进行评审
7. 测试用例通过后，搭建测试环境，开始测试，开发提交第一版本，可能存在问题或是没有完成的，都要详细说明，发现bug提交bug
8. 开发提交修改后的第二版本，主要测试修改的bug和修改后是否会引起其他问题，或是新增的功能，测试执行，重复上面的工作，直到测试完成。
9. 验收测试在功能测试和系统测试之后进行，所以验收测试进行的前提条件是系统或软件产品已通过了系统测试，而且要求软件系统必须在真实的环境下运行。

**1、测试内容**  
　　验收测试就是检验系统是否达到用户需求说明书的要求，验收测试就是尽可能地改善产品，使产品最终被用户接受。其主要包括易用性测试、兼容性测试、安装测试、文档（如用户手册、操作手册等）测试等几个方面的内容。  
**2、测试步骤**  
　　（1）测试计划在需求分析阶段建立，主要了解软件功能和性能要求、软硬件环境要求等，并特别了解软件的质量要求和验收要求。根据软件需求和验收要求编制测试计划，制定需要测试的测试项，制定测试策略及验收通过准则，并经过客户参与的计划评审。  
　　（2）建立测试环境。根据验收测试计划、项目或产品验收准则完成测试用例的设计，并经过评审。  
　　（3）准备测试数据、执行测试用例，记录测试结果。  
　　（4）分析测试结果。根据验收通过准则分析测试结果，作出验收是否通过及测试评价。通常会有四种情况：  
　　1 测试项目通过。  
　　2 测试项目没有通过，并且不存在变通方法，需要作很大的修改。  
　　3 测试项目没有通过，但存在变通方法，在维护后期或下一个版本改进。  
　　4 测试项目无法评估或者无法给出完整的评估。此时必须给出原因。如果是因为该测试项目没有说清楚，应该修改测试计划。  
　　5提交测试报告。根据产品设计说明书、详细设计说明书、验收测试结果和发现和错误讯息，评价系统的设计与实现，最终通过验收测试报告和缺陷报告等体现出来。  
**3、验收测试完成标准**  
　　1 完全执行了验收测试计划中的每个测试用例。  
　 2 在验收测试中发现的错误已经得到修改并且通过了测试。  
　　3 完成软件验收测试报告。  
**4、注意事项**  
　　1 必须编写正式的、单独的验收测试计划。  
　　2验收测试必须在实际的用户运行环境中运行。  
　　3 由用户和测试部门共同执行比较好。如果是公司自我开发的产品，由测试人员和产品设计部门、市场部门等共同进行，可能还包括技术支持、产品培训部门。

## 3.测试人员的职责

测试人员在测试过程中起到不可或缺的任务

1 编写测试计划：仔细阅读项目规格说明书、设计文档、使用说明书等，充分掌握软件的性能、特点、使用方法、业务流等，协助测试经理制定项目的测试计划，保证产品测试工作的计划和与规范性

2 编写测试用例：按照测试流程、计划以及对产品特性的把握，沟通确认测试的范围、重点，考虑逻辑、数据完整性等要求，详细规定测试的要求，策划、编写测试用例，设计测试用数据及预期结果，做好测试前的准备工作，确保测试目的的达成

3 搭建测试环境，保证测试环境的独立和维护测试环境的更新，做好测试前的准备工作，确保测试环境的稳定和版本的正确。

4 执行测试，根据测试计划及测试案例，执行测试，并根据产品特点及测试要求，实施集成测试、系统测试等，及时发现软件缺陷，评估软件的特性与缺陷，确保测试目的的达成。

5 进行BUG验证根据测试结果，与开发部门反复沟通测试情况，督促开发部门解决问题，修正测试中发现的缺陷，完善软件功能

6 进行测试记录和相应文档编写。

7 编写测试报告和对测试结果分析，通过测试，掌握软件具有的能力、缺陷、局限等，对软件质量给出评价性的结论与意见，整理测试文档，填写软件测试报告,编写测试总结，为软件开发成果提供总结性意见

8.熟悉测试流程，能独立完成测试计划，测试用例设计。熟悉数据库检索操作，

9.根据项目计划制定项目测试用例并实施，保证项目质量和进度-根据项目设计与实现有关自动化测试的代码与用例-根据项目特点，开发合适测试工具或自动化解决方案，提高测试效率-参与程序架构和代码的评审工作，并提出改进意见-设计与执行测试用例，跟踪定位产品软件中的缺陷或问题-与项目相关人员就项目进度和问题进行沟通

10.负责测试计划编写，测试用例设计，测试执行和报告，编写必要的测试工具。

## 4.黑盒，白盒，敏捷，冒烟

1黑盒

主要就是测试的功能，就是根据这个产品的特性，操作描述和用户方案，来测试这个产品的可操作性，来确定他们是否可以满足设计要求，功能测试就是为了确保程序以期望的方式运行，然后按照功能要求对软件进行的测试，通过对一个系统所有特性和功能都进行测试，来确保这个程序是否符合需求和规范。

2白盒

也称结构测试或逻辑驱动测试，它是按照程序内部的结构测试程序，通过测试来检验程序内部动作是否按照设计规格说明书的规定正常运行，检验程序种的每条通路都是否能按预定要求正确工作。

3敏捷

敏捷开发的最大特点就是高度迭代，有周期性，并且能够及时，持续的响应客户的频繁反馈，敏捷测试既是不断修正质量指标，正确建立测试策略，确认客户的有效需求得以圆满实现，和确保整个生产过程安全的，及时发布最终产品，简单来说，敏捷测试就是顺应敏捷开发方法，力求达到效率平衡的一系列实践。

4冒烟

冒烟测试这一术语源自于硬件行业，对一个硬件或组件进行更改或修复后，直接给该设备加电，如果没有冒烟，该组件就通过了测试，对软件的局部嵌入代码验证抽查测试，以避免整个版本出现问题。

## 5.软件的生存周期（瀑布模拟，v模型）

软件的生存周期是指该软件开发的全部过程，活动和任务的结构框架，是从可行性研究都需求分析，软件设计，编码，测试，软件发布维护的过程，在经历需求，分析，设计，实现，部署后，软件将被使用并进入维护阶段，直到最后因为缺少维护费而逐渐消亡，这样的一个过程，被称之为“软件的生存周期”。

（瀑布模型（Waterfall Model） 是一个项目开发架构，开发过程是通过设计一系列阶段顺序展开的，从系统需求分析开始直到产品发布和维护，每个阶段都会产生循环反馈，因此，如果有信息未被覆盖或者发现了问题，那么最好 “返回”上一个阶段并进行适当的修改，项目开发进程从一个阶段“流动”到下一个阶段，这也是瀑布模型名称的由来。）  
（V模型强调测试在系统工程各个阶段中的作用，并将系统分解和系统集成的过程通过测试彼此关联。V模型从整体上看起来，就是一个V字型的结构。这就是V模型名称的由来。）

## 6.简述测试工作的重要性

测试在整个项目开发过程中，是有举足轻重的作用。一个项目能不能如期完成，能不能满足客户的要求，以及后期的服务质量，都跟测试小组的测试有着很大关系。

测试小组的成员，除了测试经理以外，还有进行白盒测试和黑盒测试的员工。白盒测试和黑盒测试在工作职责上的区别是，白盒测试：在开发结束前，排除尽可能多的逻辑错误，黑盒测试：在开发结束后，排除尽可能多的集成错误。

然而无论是白盒测试，还是黑盒测试，在软件开发产业中有一种非常普遍习惯，那就是让那些经验最少的新手、没有效率的开发者或不适合干其他工作的人去做测试工作。这绝对是一种目光短浅的行为，对一个系统进行有效的测试所需要的技能绝对不比进行软件开发需要的少，事实上，测试者将获得极其广泛的经验，他们将遇到许多开发者不可能遇到的问题。为什么这样说呢？因为测试不仅是在测试软件产品本身，而且还包括软件开发的过程。如果一个软件产品开发完成之后发现了很多问题，这说明此软件开发过程很可能是有缺陷的；而且测试的目的是寻找Bug；避免软件开发过程中的缺陷；衡量软件的品质；关注用户的需求。而这些都是项目至关重要的部分。

## 7.Oracle和mysql的区别

我们公司Oracle和MySQL都是平时比较常用的,Oracle和MySQL都是关系型数据库，他们之间的区别主要有：

1.oracle 的存储量、吞吐量比较大，但是查询速度慢，适合于不太及时响应并且数据运算量比较大的数据查询，

2．mysql 属于轻量级的 安装简单，开源免费 适合于简单数据结构的查询

3. Oracle的默认端口号是1521，而mysql数据库的默认端口号是3306；

4..mysql有主动自动递增的功能，oracle数据库没有这个功能，

5.分页：mysql分页和oracle分页不一样，相比较mysql更简单一些

6.mysql数据库类型是int, oracle数据库类型是number

Mysql是自动提交数据， oracle是手动提交数据

mysql是date类型，而oracle是需要通过to\_date 来转换的。

## 8.Sql语句的优化

1.对查询进行优化，应尽量避免全表扫描，首先应考虑在 where 及 order by 涉及的列上建立索引。

2.应尽量避免在 where 子句中使用 or 来连接条件，否则将导致引擎放弃使用索引而进行全表扫描

3. in 和 not in 也要慎用，否则会导致全表扫描

4. 应尽量避免在 where 子句中对字段进行表达式操作

5. 不要写一些没有意义的查询

6. 尽量避免大事务操作，提高系统并发能力

7. 尽量避免向客户端返回大数据量，若数据量过大，应该考虑相应需求是否合理

8. 任何地方都不要使用 select \* from dual，用具体的字段列表代替“\*”，不要返回用不到的任何字段

## 9.数据库的优化（分表，分库）

1.调整数据结构的设计。  
2.应用程序结构设计  
3.调整数据库sql语句。  
4.升级硬件配置。  
5.调整系统的参数。  
6.尽量使用索引。  
7.分表分库。

## 10.Windows中搭建测试环境

1.jdk安装

2.配置好jdk环境变量  
3.Tomcat安装  
4.将war包放在Tomcat的发布目录中webapps中，  
5.conf》server.xml里面设置默认解压，unpackWARS = true   部署：autoDeploy = true  
6.在bin目录下启动Tomcat的startup.bat就可以。  
7.启动后输入“IP地址:端口号/Tomcat》WebApps文件里面解压后文件的名字（没有后缀）”  
8.在需要改数据库地址的可以问开发怎么改，可以链接远程的数据库。需要jar包的可以问开发要。

9.给本机设置用户名和密码  
10.点击计算机右键属性》选择左上角点击远程设置》点击远程》选择允许远程协助和允许任意版本远程桌面计算机》确定好后，同一网   段的用户可以输入本机的IP地址，输入账户和密码进行访问。

注意：

1.如果只要启动一个项目，安装一个Tomcat就好。

2.如果启动多个项目（一个项目中的所有子项目也算多个项目），可以将所有的项目都放在WebApps里面运行。

3.如果要单独启动项目，可以创建多个Tomcat，单独运行。

## 11.简述测试用例（测试用例的要素，测试用例的编写方法，测试用例的覆盖率和正确性）

测试用例就是对软件的目标编写的一组测试数据、执行条件以及预期结果，来检验是否满足软件需求。写测试用例首先要熟悉测试相关的功能和流程，也需要仔细了解产品的各项功能。要知道测试的目的就是找到缺陷，想方设法的让软件更加优化，更加完美。

用例要素，如何写测试用例

我在公司写测试用例的时候，用例主要是由用例编号、用例类型、所属模块、用例标题、用例级别、前置条件、测试数据、操作步骤、预期结果、实际结果、缺陷编号和编写人这几个组成的。

然后我简单说明一下这个用例标题、前置条件,操作步骤,预期结果，缺陷编号的意思

用例标题：给当前这条用例起个名字，要求测试人员看见名称就大概明白测试用例所测试的点，达到见名知意。

前置条件：就是在执行操作步骤前，系统需要达到的状态

操作步骤：如果有多个步骤，每一个步骤都需要填上序号，每一行一个步骤，不能写得太简略了，至少要让测试人员可以执行，也不能写得太复杂。

预期结果：当前这个功能点这条用例要达到什么样的效果，如果有多个检查点，需要都列出来，每一行标个序号，让人一目了然，检查点尽量写详细些，不要出现结果正常、不正常等，应该描述出正常的具体情况。

缺陷编号：是当前项目在测试过程中，出现的bug或问题，进行记录，或者是详细说明。

写测试用例的时候我是从等价类划分法、边界值分析法、因果图法和错误推测法等方面分析来写的。

测试用例并不是说写的越多越好，如果测试用例中有太多无效的用例，执行测试用例时会浪费大量测试人力，也不会产生测试效果。它的关键在于用比较少的用例覆盖尽可能多的模块。不可能做到完全测试，以较少的用例在合理的时间内发现最多的问题。

我们公司都是在确定好需求文档后就开始写测试用例了，我经常word文档里写，写完了评审，等开发开发好一个模块发给我，我就开始按照测试用例执行测试。

# 二月份

## 1.简述白盒测试工作

白盒测试：在开发结束前，排除尽可能多的逻辑错误

负责业务系统的测试工作，编写测试文档和测试报告  
制定测试计划和测试流程，保证测试质量  
开发自动化测试框架，提高测试用例的自动化程度  
研究和实施测试框架和测试方法，提高测试效率

## 2.简述朋友圈测试

**等价类划分法：**  
点赞显示是否正常，一行能显示多少个赞。  
允许朋友圈查看的范围（例如：三天，半年，全部）功能是否正常使用  
**边界值分析法：**  
对朋友圈评论的字数，朋友圈上传的照片最多几张，上传视频可以上传多长时间的。  
是否允许陌生人查看十张  
因果法：  
为什么会有朋友圈这个功能，为什么要在朋友圈上面发照片，发视频，发音乐，评论等  
**异常推测法：**  
在什么情况下朋友圈不能正常使用，不能正常发视频，发音乐，不能评论等。  
首先要从它的功能性进行测试，看能不能正常发朋友圈。  
**功能性：**可用性测试：  
点赞是否按照时间顺序正常排序，头像对应的点赞人是否正确  
是否能在消息里列表中显示点赞人的昵称，备注。  
对发出的不良信息（色情低俗，政治敏感，违法暴力，欺诈骗钱，广告其他，暴恐血腥）有没有屏蔽，删除，举报等操作  
点赞状态是否同步更新显示。  
点赞状态，共同朋友是否可见  
是否可以把可有权设置成尽自己可见，是否可以设置不让某些人查看自己的朋友圈  
是否开启朋友圈更新提示。  
  
**性能测试：**  
加载速度，延迟，网络快慢对其影响、  
**并发测试：**如果5000人同时发送朋友圈。朋友圈会不会崩溃。  
**可拓展性测试：**  
点赞后，是否可以发表评论，评论是否可以添加表情，图片等。  
**安全性测试：**  
朋友圈发布的信息会不会泄露  
**界面测试：**  
是否符合正常人审美。

## 3.简述python集合区别（元组，列表，字典）

**tuple元组是不可变序列**

1.tuple中可以存放任意类型

2.tuple中通过下标访问元素，下标从0开始

3.对象可以引用数组

**列表是可变序列**

List 可变序列的特点就是

1.list可以存放任意类型

2.list通过下标访问元素，下标从0开始

3.支持原位改变

4.长度可变，可以任意嵌套

5.对象可以引用数组

**字典的特点**

1.以键值对形式存储，通过key键访问value值

2.可以包含任意类型的无序集合

3.可变长度，可以任意嵌套，属于可变映射

## 4.简述接口测试的回归测试

那我就说一下用python来测接口的回归测试吧：

用python测接口的话,就是开发把接口文档给我,我核对之后,就先把requests包安装上导进去，然后根据文档要求开始写脚本,先定义好文档上需要的key,value值,然后把每个接口的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,复制给一个变量名称，然后把这些变量名称通过列表的方式存储，然后使用for循环来循环整个列表，然后requests.get放进去url地址和传参来获取响应报文和响应码，并打印。

2.然后看返回结果请求值，成功是200,错误返回404,500等，，如果是301的话表示永久跳转，302表示临时跳转，403表示没有权限，拒绝访问，500服务器内部错误，503系统繁忙。

3.我们传参的时候一定要按照接口文档来写

4.接口的性能我们也很注意的，因为他直接影响了用户的体验，如果半天不返回数据，我们就果断放弃，让开发继续优化。

当全部接口测试通过以后我们会生成接口测试报告并提交给测试经理。我们就是这样做回归测试的。

## 5.简述界面自动化测试（分为接口自动化，界面自动化）

我说下用python做自动化吧:

1.我用python做自动化的时候，先把项目跑起来，然后我就开始测试，浏览器我一般用的是谷歌，提前得先下载一个谷歌驱动来帮我们实现访问的功能

2.接着用pycharm编写代码，新建一个python项目。安装好selenium插件，然后调selenium,time时间库等等,在需要的时候可以让它休眠一下，防止报错,这个是比较需要注意得。接着咱们启动浏览器,用webdriver.Chrome(),给他一个赋一个值比如“cr”,然后就用cr.get(),括号里写上咱们需要访问得域名,记得域名要加引号,接着就运行让他跑起来,看看能不能访问到域名里面得内容,如果成功得话,就继续下一步,

3.我们获取元素得方式有by\_name,by\_id,by\_xpath等等,功能就是哪个好用就用哪个,一般我们用得比较多的就是xpath,如果是按钮的话我们后面给他加个.click,如果是文本框的话我们后面加.sendkeys(),括号里写上参数,如果有弹窗得话我们还需要用到cr.switch\_to\_alert这个方法,可以直接在后面加accept和dismiss,也可以给它赋值,比如叫a1,那就a1.accept表示确定,a1.dismiss表示取消,完了就.close或者.quit直接关闭页面或者退出浏览器

4.这就是我用python做自动化测试得流程

Find\_element\_by\_id() 通过ID定位  
Find\_element\_by\_name() 通过name定位  
Find\_element\_by\_xpath() 通过xpath定位  
Find\_element\_by\_class\_name() 通过classname定位  
Find\_element\_by\_tag\_name() 通过tagname定位  
Find\_element\_by\_css\_selector() 通过css定位  
Find\_element\_by\_link\_text() 通过linkText定位  
Find\_element\_by\_partial\_text() 通过匹配到的部分linkText

目前使用比较多的是cssSelector 和 xpath，因为一个页面中ID、name、className

## 6.简述在liunx搭建环境

1.Linux搭建测试环境,我们当时使用的是centos6.8, jdk1.8, tomcat1.88版本的  
使用远程连接工具SSH连接服务器ip地址.  
首先进入usr目录,为了方便管理我们创建两个文件夹文件夹,jdk和tomcat将jdk和tomcat分别上传至linux 的/usr目录下的jdk和tomcat文件夹下，然后rpm -ivh 安装jdk,可以去/usr下查看一个叫java的文件，也可以通过mv更改名称，然后返回/etc文件夹,vi 编辑profile文件.在最底部配置jdk环境变量(JAVA\_HOME=/usr/java/jdk1.8.0\_121  
CLASSPATH=.:$JAVA\_HOME/lib/tools.jar:$JAVA\_HOME/lib/dt.jar   
PATH=$JAVA\_HOME/bin:$HOME/bin:$HOME/.local/bin:$PATH  
)  
添加三个变量,java\_home,classpath和path  
添加完成后 :wq! 强制保存并退出,使用source [sou er si]profil命令,使其立即生效.  
使用java -version验证jdk是否配置成功.出现jdk版本信息说明配置成功.  
  
将tomcat压缩包上传至linux,进入usr/tomcat目录查看ls命令查看是否上传成功.  
使用unzip 解压tomcat解压完成后进入tomcat的bin目录,

ls查看是否有权限，没有就

chmod +x \*.sh启动sh为后缀名的文件权限

./startup.sh启动tomcat.在浏览器输入ip地址:8080出现welcome to tomcat说明tomcat配置成功.  
Linux测试环境搭建成功.  
如果tomcat起不起来,先检查jdk,因为tomcat需要基于java环境.  
检查jdk环境变量配置，如果是在本机不能启动，就检查一下防火墙有没有关。

**2.简述在linux系统当中搭建nginx集群的步骤以及nginx的执行原理；**  
  在linux系统测试环境搭建完成后,使用的nginx搭建nginx集群,首先分别解压3个tomcat，mv改名称：比如t1，t2，t3，等将项目的war包分别放入t1,t2,t3的webapps目录,去更改端口号，授予权限，分别启动三个tomcat  
上传nginx安装包到linux,tar -zxvf 解压,先安装依赖环境gcc-cc++,gcc  
然后用./configure进行初始化,在用make编译,用make install进行安装,然后在nginx下的sbin目录中进行./nginx命令启动nginx,在浏览器输入ip地址可以看到welcome to nginx  
在nginx目录下的conf文件编辑conf文件夹里的nginx.conf文件,在nginx.conf里找到gzip on在下一行写配置文件  
(通过upstream{自定义名称}关键词+自定义集群名称，然后再花括号中进行集群的tomcat配置，包括tomcat的ip地址，端口号，和根据tomcat性能配置的权重信息，然后再server属性的花括号中在请求nginx时，指定调用的集群名称，即可在请求服务器地址的时候，转向请求集群的tomcat，然后因为多个tomcat需要解决session的问题，可以通过ip\_hash,  
  
 最后在浏览器输入访问的域名此时nginx 集群搭建成功.  
  
Nginx是一个代理服务器,可是实现负载均衡.  
代理分为正向代理和反向代理.  
正向代理简单的说,我是一个用户,我无法直接访问一个网站,但是我可以访问一个代理服务器,代理服务器可以访问这个网站,我告诉他我无法访问网站的内容,代理服务器取回来,然后返回给我.  
反向代理就是此时我一个集群为一个nginx+三个tomcat,此时一个tomccat出了问题,另外两个可以正常使用,代理服务器将推送给我正常的tomcat.  
负载均衡就是分布式部署,简单的说当两台或者两台以上服务器,根据规则随机的将请求分配给制定的服务器处理.这种方法成本较低,廉价无瓶颈.

## 7.简述nginx+tomcat集群技术

首先我们配置好tomcat,然后上传nginx安装包到linux,tar -zxvf 解压,先安装依赖环境gcc-cc++,gcc  
然后用./configure进行初始化,在用make编译,用make install进行安装,然后在nginx下的sbin目录中进行./nginx命令启动nginx,在浏览器输入ip地址可以看到welcome tonginx.  
在nginx目录下的conf文件编辑conf文件夹里的nginx.conf文件,在nginx.conf里找到gzip on在下一行写配置文件  
(通过upstream{自定义名称}关键词+自定义集群名称，然后再花括号中进行集群的tomcat配置，包括tomcat的ip地址，端口号，和根据tomcat性能配置的权重信息，然后再server属性的花括号中在请求nginx时，指定调用的集群名称，即可在请求服务器地址的时候，转向请求集群的tomcat，然后因为多个tomcat需要解决session的问题，可以通过ip\_hash,  
  
 最后在浏览器输入访问的域名此时nginx 集群搭建成功.

## 8.简述http和https的区别

1.https协议需要到ca申请证书，一般免费证书较少，因而需要一定费用  
2.http是超文本传输协议，信息是明文传输，https则是具有安全性的ssl加密传输协议  
3.http和https使用的是完全不同的连接方式，用的端口也不一样，前者是80，后者是443  
4.http的连接很简单，是无状态的；https协议是由ssl+http协议构建的可进行加密传输、身份认证的网络协议，比http协议安全

5.https连接前需要下载数字证书来确认服务器的真实性，避免其他劫持者伪装服务器对客户机进行劫持和篡改。

6.http无法对应用层的结果进行加密处理，https就可以，然后再应用层解密来显示结果。

## 9.简述服务器的双机热备，主从同步（电商网站使用比较多）

**数据库双机热备：**

双机热备是应用于服务器的一种解决方案，其构造思想是主机和从机通过TCP/IP网络连接，正常情况下主机处于工作状态，从机处于监视状态，一旦从机发现主机异常，从机将会在很短的时间之内代替主机，完全实现主机的功能。

**数据库主从同步：**

主从数据库是一个异步的复制过程，主库发送更新记录到从库，从库读取更新记录并执行更新记录，使得从库和主库的内容保持一致，在主库里只要有更新记录从库就会跟着更新。

## 10.简述BS（浏览器端，不用安装客户端，服务器，网络压力大）和CS（客户端，服务器模式）架构区别

简单来说一下什么是cs和bs架构，CS架构，就是在你的电脑，需要装个软件，或者app，才能连接服务器。而BS架构，就是你的电脑，只需要用浏览器，就可以连接服务器了。

## 11.简述电商项目购物车，订单，付款的业务

**1.界面测试：**  
（1）打开页面后，页面的布局是否合理，页面是否完整。

（2）不同卖家的商品在不同的区域显示。   
（3）页面的功能按钮可以正常显示。  
（4）商品最下面显示失效宝贝。  
（6）向下滑动页面，在购物车顶端展示这个中间展示购物车  
（7）购物车有商品降价或者库存不足或者限购件数，在商品详情下面，会有对应字体展示。  
**2.基本功能：**  
（1）购物车页面所有链接正常。  
（2）从商品信息页面添加的商品能显示在购物车中。  
（3）若未登录，点击购物车中的商品直接进行结算，则提示用户输入用户名和密码，或者提示其他的非注册用户购物方式，或是第3方登陆。  
（4）若没有选择任何商品，点击结算，则提示用户请添加要结算的商品  
（5）勾选商品后，已选商品的总价（和优惠满减活动）会显示。

（6）勾选商品，点击结算按钮后，进入确认订单信息页面。  
（7）购物车页面中，可以对添加的商品信息做信息的修改，并自动保存成功。  
（8）可以在购物车中重新修改商品规格。  
（9）购物车能添加的商品种类是有数量上限的。  
（10）结算的时候商品可以全选，选择底部的全选按钮。

（11）可以在购物车页面对宝贝进行管理。  
**3.性能测试**：  
（1）打开购物车时间是否在已定的用户可以棘手的时间范围内。  
（2）编辑购物车：删除、增加商品需要的时间。  
（3）在购物车页面选择需要购买的商品进行结算的时候，结算金额可不可以实时显示。  
（4）清空失效商品需要的时间。  
（5）商品批量  
**4.兼容性测试：**  
（1）不同型号，不同的系统下，操作方法，功能是否一样。  
**5.网络环境：**  
（1）3G、4G、wifi网络环境下应用的各功能可正常运行。

（2）网络异常时，数据交换是否会有提醒。  
（3）有网到无网再到有网，数据自动恢复，界面是否可以正常加载。  
（4）只允许内网访问的APP，在连接到外网时是否会有提醒。  
**6.异常测试：**  
（1）没有内存时，APP是否能够正常响应。  
（2）横竖屏切换展示。  
（3）APP运行时网络中断。  
（4）反复操作某一个功能，不断点击和刷新，是否出现闪退黑屏。

（5）APP运行时接入电话、短信、微博视频时，是否正常运行。  
**订单测试**  
**1.收货信息：**  
 1新增、修改、删除、收货信功能是否正常。  
 2最多可以添加多少个收货信息。  
 3新增收货信息中有限制，输入框对字符类型和长度是否有 限制。必填项为空能否保存收货信息  
 4保存成功之后能否在列表正确显示。后台能否查看到保存的信息。  
 5没有收货信息能否提交订单。  
 6收货信息能否多选。

2.付款测试

1选好要购买的商品，店家右上或者右下的 结算按钮会调到一个结算页面 ，页面上部分是收货地址 下面是宝贝列表 ，每个宝贝下面都有一个买家留言，（可写可不写）

2每种支付方式是否功能正常。比如说选择货到付款，还需不需要支付。  
3第三方支付是否支持。  
4能否选择同时选择多种支付方式。  
5右下方有提交订单的按钮，点击提交，会跳到付款页面 ，如果之前有跟卖家说好要给你免邮或者额外优惠的，提交订单后打住，联系卖家已提交订单，让卖家给你改价。

比如说：

1. 优惠券、红包、积分显示是否正确，是否与后台匹配。  
   2.优惠券、红包、积分能否正常使用，能否叠加使用。  
   3.优惠券、红包、积分使用后支付金额是否相应减少。后台优惠券、红包、积分是否也相应减少

6卖家价格改好后，系统会自动跳出已改价的通知，之后手动刷新该页面。你已经往支付宝或者是微博里存了钱，就直接输密码付款。好了之后，系统会通知你购买成功

7异常场景：  
 1.金额不够的情况下，能否购买，是否有提示  
 2.支付过程中断网，断电，对支付是否有影响  
 3.在弱网情况下，能否正常支付

## 12.简述线程，进程以及同步和异步的区别

**进程和线程的区别**

1.进程是一个程序在自身地址空间中的一次执行活动，是资源申请，调度和独立运行的单位

2.线程是进程中一个单一的连续控制流程，一个程序可以拥有都个线程

3.线程本身没有独立的存储空间，而是和所属进程中的其他线程共享一个存储空间。

**同步和异步的区别**

**同步：**进程之间的关系不是相互排斥的关系，而是相互依赖的关系。简单举个列子来说，（就是相当于同一个跑道，必须该项目所有参赛人员到齐，一起出发，这是同步），就是前一个进程的输出作为后一个进程的输入，当第一个进程没有输出时第二个进程必须等待。具有同步关系的一组并发进程相互发送的信息称为消息或事件。

**异步：**异步和同步是相对的，同步就是顺序执行，执行完一个再执行下一个，需要等待、协调运行。异步就是彼此独立,在等待某事件的过程中继续做自己的事，不需要等待这一事件完成后再工作。线程就是实现异步的一个方式。异步是让调用方法的主线程不需要同步等待另一线程的完成，从而可以让主线程干其它的事情。

## 13.简述消息队列（消息队列的定义，作用，四种使用场景，如何测试消息队列）

**定义：**

消息队列是服务架构中常见的组件，可用于服务间解耦、事件广播、任务异步/延迟处理等，常用于分布式系统之间互相信息的传递。

**作用：**

消息队列的主要作用，有解耦，异步，消峰  
**解耦：**将消息写入消息队列，需要消息的系统自己从消息队列中订阅，从而系统不需要做任何修改。  
**异步：**将消息写入消息队列，非必要的业务逻辑以异步的方式运行，加快响应速度  
**削峰：**系统慢慢的按照数据库能处理的并发量，从消息队列中慢慢拉取消息。在生产中，这个短暂的高峰期积压是允许的

**四种场景：**

消息队列在实际应用中常用的使用场景。异步处理，应用解耦，流量削锋和消息通讯四个场景  
**1. 异步处理：**用户注册后，需要发注册邮件和注册短信。传统的做法有两种 :**串行的方式**和**并行方式  
2. 应用解耦：**用户下单后，订单系统需要通知库存系统。传统的做法是，订单系统调用库存系统的接口。  
**3. 流量削锋：**秒杀活动，一般会因为流量过大，导致流量暴增，应用挂掉。为解决这个问题，一般需要在应用前端加入消息队列。  
**4.消息通讯**  
消息通讯是指，消息队列一般都内置了高效的通信机制，因此也可以用在纯的消息通讯。比如实现点对点消息队列，或者聊天室等。  
**如何测试消息队列**  
**接口测试:**  
 1、跟相应的开发要rabbitmq的链接以及用户名、密码，在chrome中登录，然后根据接口文档提供的队列名称在在里面找到相应的队列，点击进入到队列的详情页，再传入需要传的参数（消息json）；  
 2、打开chrome的开发者工具，Network,点击publish message按钮，找到对应的操作名称，headers中就有我们想要的信息（URL和请求类型）；  
 3、然后在jmeter中填入我们已知的URL和请求类型，在body中填入正确的参数，body中的信息可同样从headers底部中获得，然后将其转换为相应的格式直接复制过去即可；  
 4、还有一个特别需要注意的地方，要在header中加rabbitmq的登录信息（需要找对应开发提供）。否则可能会报错

**整体大概消息队列流程**

**数据发送方**  
数据发送方是否可以成功给消息队列发送数据，  
是否可以成功接收到消息队列返回的操作成功的消息，数据发送方是否会断开与消息队列的连接  
**消息队列**  
当消息队列接收数据发送方的数据，最多存储多长时间以及存储的机制顺序或者倒叙  
接收到发送方的消息是否会返回给发送方状态码   
发送方发送失败是否会生成存储该条异常日志  
消息队列是否支持手动或者自动推送信息  
消息队列是否会监控接收方的状态，接收方正常是否会连接接收方，接收方异常是否会与接收方断开连接  
**数据接收方**  
是否能接或收读取消息队列的数据

## 如何做好测试工作

1.在测试前期要充分的了解和掌握项目需求  
2.要多和项目经理，开发经理，产品经理甚至客户进行沟通  
3.根据项目的整体情况以及后续的变化不断的调整我们的测试计划  
4.每次测试结束之后都要总结和反思  
5.在整个测试过程中，要学会调整自己的状态，时刻保持愉悦的心情工作  
总的来说就是：适当的引入自动化测试，减少繁琐重复的测试工作  
　　手动测试固然能保证测试的质量，但也是对人耐性的一个考验，而自动化测试既能保证测试质量，又能节省时间。  
　　对程序的回归测试更方便  
　　可以运行更多繁琐复杂的测试  
　　可以执行一些手动测试困难或者不可能进行的测试  
　　可以开展兼容性测试，减少实际的兼容性测试时间以及提高测试的覆盖率  
　　所以说为了保证测试的质量和进度，提高测试效率，我们可以适当的引入自动化测试。  
　　当然，传统的手动测试是必不可少的，以手动测试为主。

## 如何提交一个高质量的bug？

1.多次复现bug，提高bug的浮现率

2.判断前后台bug，分开提交bug  
3.准备测试数据  
4.简化测试步骤  
5.附带接口返回值  
首先bug多浮现几次，提高bug的复现率，判断是前端的bug还是后台的bug

## 当开发认为不是bug时，如何处理？

开发认为不是bug，通常有两种情况。

就是需求没有确定，我们首先找到需求文档，看有没有对预期结果进行具体说明，从而提高说服力度，然后就是我们要确保自己的bug能够重现（可以截图保留留证据，或是拍照保留证据）。

然后再分析一下自己的bug的级别，如果说只是建议性的bug可以保留，但是也要在相应的工具上进行记录，如果说bug的级别比较高，就要跟开发进行沟通，讲明这个bug的重要性以及危害和重现步骤等，实在不行就跟开发经理，测试经理进行沟通，说明bug的重要性，有必要时进行问题评估，一起讨论解决这个问题。

报Bug的礼仪

不要对一个程序 员说:你的代码有bug。

他的第一反应是: 1.你的环境有问题吧; 2.傻逼你会用吗。

如果你委婉的说:你这个程序和预期的有点不-致，你看

看是不是我的使用方法有问题。

他本能的会想:操，是不是出bug了!

运软件测试天师

项目二:在系统测试期间，我遇到过一个严

重的问题，服务器的响应时间越来越长，尤

其是越来越多的人在同- -时刻使用该系统,

在很很大的压力下，我仔细的检查了tomcat

日志和java代码，最后我准确地判断数据库

的连接在程序的结尾处没有被关闭。

## 测试计划和测试方案的区别？

测试计划是管理文档，统领整个测试工作，对测试项目是一个大范围的掌握。

测试方案是属于技术类的文档，在测试计划之后，要比测试计划详细，需要归类测试点，对测试方法、类型、场景等有一个详细的描述。

## 18.测试人员应具备的职业素养

首先是责任，作为一个优秀的测试人员，一定要有对于工作的责任心，还有就是能妥善处理问题，坚信自己提出的每一个问题，站在用户和公司的角度考虑问题。

# 三月份

## 简述单元测试（什么叫单元测试，操作步骤）

**什么叫单元测试**：

简单来说就是对程序的每一个模块进行测试，集中对源代码实现的每一个程序单元进行测试，检查各个模块都是否实现规定的功能。

**具体操作：**

比如说我们经常使用的junit4单元测试，就在测试的项目lib文件夹下导入junit的两个jar包，比如说我们创建一个类com.qzx.math,然后对这个类进行单元测试，创建一个和src同级别的文件就叫test吧，然后我们把这个这个文件设置成根目录，创建一个com.qzx.mathtest类，然后在mathtest里输入内容，去运行，如果是绿色则证明通过。

## 简述集成测试（什么叫集成测试，操作步骤）

**集成测试**

是单元测试的逻辑扩展，它最简单的形式是：两个已经测试过的单元进行组合，测试他们之间的接口，这就是集成测试

**集成测试的操作步骤**

1首先单元测试组合完成后

2准备测试计划

3准备集成测试场景和测试用例。

4准备测试自动化脚本

5执行测试用例

6执行用例后提交缺陷

7重新测试缺陷以及其他问题

8一直到集成测试完成

## 简述压力测试，负载测试，并发测试的区别

**压力测试：**又叫强度测试，压力测试是评估系统处于或超过预期负载时系统的运行情况，关注点在于系统在峰值负载或超出最大载荷情况下的处理能力。

**负载测试**：是通过逐步增加系统负载，测试系统性能的变化，并在满足最终确定性能指标的情况下，系统所能承受的最大负载量。

**并发测试**：通过模拟多用户访问，测试多个用户同时访问一个应用，模块或数据记录时是否存在卡死或其他性能问题，发现其他并发访问时的问题。

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

## 4.简述jmeter工具

Jmeter是java开发的开源测试软件，它可以用来实现接口测试，性能测试，自动化测试。还可以用于测试静态和动态资源，jmeter可以对于服务器模拟巨大的负载来检测系统的强度，并分析整体的性能。  
另外jmeter可以用来做功能测试和回归测试。一般和bodboy结合使用。Bodboy录制好脚本以后，导入到jmeter，添加断言来验证程序返回的结果是否正确，断言就是对测试的结果进行预判，如果返回结果和预测值匹配则为正常。断言就是为了让jmeter有逻辑判断能力。  
我们公司在用jmeter做接口测试的时候（接口自动化测试），拿到开发给的模块，创建线程组，选择配置元件，选择http默认值，输入接口的路径，参数，可以使用get或者post请求等等，返回json形式的结果，添加结果树查看返回的数据是否正常。对于返回结果也可以进行断言。这就是我用jmeter做的接口测试。  
用jmeter做性能测试的时候，也得使用bodboy录制脚本，（因为jmeter自带的录制脚本不好用）例如登录功能，我就在登录前设置同步定时器，设置好用户的人数和集合等待的时间，在执行该功能，以获取并发情况下同时执行该功能的详细数据。对项目的性能进行评估，推断。  
因为jmeter图表比较少，如果要检测服务器的内存，cpu，吞吐量，磁盘，io等信息的话就要下载jar包，放在jmeter的lib目录的ext文件中，创建监听器，然后将serverAgent启动，就可以对服务器硬件性能进行监听。  
**Jmeter中还可以添加扩展插件：**  
 Hits per second:服务器梅每秒点击率。  
 Response Times Over Time：响应时间。  
 PerfMoin Metrics Colllector：服务器性能监控件。

## 5.简述jmeter的断言

jmeter没有逻辑判断的能力，断言就是对服务器响应数据进行预判，根据断言判断服务器响应信息，在需要添加断言的http请求上添加一个断言，然后添加一个响应断言，在响应断言的文本框中填写断言的内容

有5种模式匹配规则

1包括：返回结果包含了指定的内容，支持正则匹配。

2匹配：必须和返回的结果完全匹配。正则表达式必须能匹配整个返回值，而不是部分返回值。  
3 Equls：（咦快死）和包括功能相同，但是要严格区分大小写，不支持正则匹配  
4 Substing （沙伯崔影死）与匹配功能相同，返回结果必须全部匹配。不支持正则表达，严格区分大小写。  
5否：取反的意思。

## **简述jmeter的自动化测试**

**首先用户自定义参数化**:在线程组上鼠标右键,在弹出的一级菜单中选择”配置元件”选项,接着在弹出的二级菜单中选择”用户定义的变量”,然后在弹出的窗口中设计变量名和变量值,最后在请求的参数列表中通过EL表达式(${变量名})的形式进行参数化.  
 **随机参数化:**点击jmeter函数助手,在弹出框中选择\_random选项,然后在选项下面需要填写要产生的随机数的最大值和最小值,填写完毕后,点击”生成”按钮,在按钮前面的文本框中会自动生成随机数参数化的字符串,格式为:${\_random(最小数,最大数)}  
 **CSV参数化**:首先在任意目录下创建一个文件,将该文件作为参数化的文档,然后打开该文件进行参数化内容的编辑并保存该文档的内容,接着在线程组上鼠标右键,在下拉列表中选择”配置元件”,然后在配置元件的二级菜单中选择”CSv date set config”,然后在弹出框中的file name属性中填写参数化文档的绝对路径,在file encoding(文档编码)填写utf-8,在variable name中填写文档中的每一列的别名,在delimiter中填写文档中列和列之间的分隔符,最后在请求的参数列表中通过EL表达式(${变量名})的形式进行参数化.  
   **JDBC参数化**:在线程组上鼠标右键,弹出的一级菜单中选择”配置元件”选项,接着在弹出的二级菜单中选择”JDBC Connection Configuration”接着在弹出框中的variable name中填写该数据库配置信息的引用名称,在databaseURL中填写要连接的数据库地址(包含数据库的ip地址,端口号和数据库名称),在jdbc driver class中填写数据库的驱动名称,在username中填写数据库连接的用户名,在password中填写数据库连接密码,填写完毕后,再添加两个jdbc request,其中一个jdbc request查询数据库表中的参数化信息的总条数,另一个jdbc request查询具体的参数,接着在”逻辑控制器”中添加一个”循环控制器”,循环控制器的循环次数就是查询出来的表中参数化信息的总条数,然后将http请求放在循环控制器的下方,由循环控制器决定http请求执行的次数,因为参数化在获取的过程中需要不断的换行,需要添加一个”配置元件”中的计数器,从第二行开始获取每一行的参数值,最后在参数列表中通过${\_V(参数名称\_${计数器引用名称})}获取数据库表中查询出来的参数值

## 7.简述jmeter的并发测试

用jmeter做并发测试的时候，也得使用bodboy录制脚本，（因为jmeter自带的录制脚本不好用）例如登录功能，我就在登录前设置同步定时器，设置好用户的人数和集合等待的时间，在执行该功能，以获取并发情况下同时执行该功能的详细数据。  
并发测试的目的是为了测试服务器对于压力的性能影响。所以还需要监控服务器的各项性能指标因为jmeter图表比较少，如果要检测服务器的内存，cpu，吞吐量，磁盘，io等信息的话就要下载jar包，放在jmeter的lib目录的ext文件中，创建监听器，然后将serverAgent启动，就可以对服务器硬件性能进行监听。

## 8.简述jmeter的性能测试，分析

性能测试工作的目的是检查系统是否满足在需求说明书中规定的性能，性能测试常常需要和强度测试结   
合起来，并常常要求同时进行软件和硬件的检测。   
性能测试主要的关注对象是响应时间，吞吐量，占用内存大小（辅助存储区），处理精度等。   
从维度上划分，性能指标主要分为两大类，分别是业务性能指标和系统资源性能指标。   
业务性能指标可以直观地反映被测系统的实际性能状况，常用的指标项有：   
1.并发用户数   
2.事务吞吐率（TPS/RPS）   
3.事务平均响应时间   
4.事务成功率   
系统资源性能指标，主要是反映整个系统环境的硬件资源使用情况，常用的指标包括：   
1.服务器：CPU 利用率、处理器队列长度、内存利用率、内存交换页面数、磁盘 IO 状态、网卡带宽使用   
情况等；   
2.数据库：数据库连接数、数据库读写响应时长、数据库读写吞吐量等；   
3.网络：网络吞吐量、网络带宽、网络缓冲池大小；   
4.缓存（Redis）：静态资源缓存命中率、动态数据缓存命中率、缓存吞吐量等；   
5.测试设备（压力发生器）：CPU 利用率、处理器队列长度、内存利用率、内存交换页面数、磁盘 IO 状   
态、网卡带宽使用情况等。

## **jmeter测试报告**

测试场景   测试结果   测试基准   测试分析  测试结论  
1.测试场景：  
1.测试的场景2.场景人数3.场景执行的时间4.场景的并发点多少人进行并发。进行书面表达并对其截图。   
2.测试结果：  
对八大监听器图表中的

属性以及图形显示结果进行属性分析。  
3.测试基准：  
     根据需求定义测试数据的基本标准。  
4.测试分析  
     根据结果图表得出的数据进行和测试的基准进行对比。  
5.测试结论：  
    通过测试分析得出最终的结论，并给出相应的优化建议。

## 10.jmeter和loadrunner区别

**Jmeter:**

Jmeter是java开发的开源测试软件，是轻量级的

2.jmeter没有IP欺骗功能

3.jmeter可以代理录制脚本，但是并不好用，所以需要结合badboy录制脚本

4.jmeter的报表比较少，对于要分析的性能测试不足以作为依据，想知道具体的服务器信息，cpu等，得在相关的服务器上另写脚本进行记录服务器的性能

5.jmeter做性能测试，主要是通过添加线程组的数目，或者是在循环控制器上来增加并发用户

6.jmeter主要是做黑盒测试

7.jmeter是开源的

**Loadrunne:**

loadrunner是由惠普公司通过c语言开发的重量级性能测试工具

loadrunner有ip欺骗功能，可以自身录制脚本

loadrunner可以通过场景选择，设置虚拟场景，虚拟用户数

loadrunner主要是做性能测试

loadrunner是商业软件，如果是正版，有技术支持，同时网络上资料相当多

**区别：**

1.jmeter轻量级性能测试工具，loadrunner重量级性能测试工具

2.jmeter测试工具支持最大测试人数为2000个，loadrunner为6.5万人次

3.jmeter不具备ip欺骗功能，loadrunner具备ip欺骗功能

4.jmeter不具备图表合成功能，loadrunner具备图表合成功能

5.jmeter不具备测试报告的自动生成，loadrunner具备测试报告的自动生成

6.jmeter不具备多格式的测试报告导出功能，loadrunner具备多格式的测试报告导出功能

7.jmeter主要是做黑盒测试，loadrunner主要是做性能测试

## 11.简述loadrunner的三大组件

loadrunner的三大组件分别有vugen controller Analysis

**vugen**是基本设置，检查点，参数化，关联，调试，事务

**Controller** 是场景设计，场景执行，资源监控，集合点，ip欺骗，负载均衡

**Analysis** 是摘要报告，常见图分析，结果分析实践

## 12.简述loadrunner如何做性能测试

1. 使用VuGen录制要执行的测试脚本，其中需要完善的项目有：参数化、关联、检查点、集合点、思考时间、事务等。再完善了脚本后最后一步对脚本进行检查工作。（录制的脚本回放时不出错不代表脚本是正确的）
2. Loadrunner将性能测试过程分为计划测试、测试设计、创建VU脚本、创建测试场景、运行测试场景和分析结果6个步骤。  
           1) 计划测试：主要进行测试需求的收集、典型场景的确定；  
           2) 测试设计：主要进行测试用例的设计；  
           3) 创建VU脚本：主要根据设计的用例创建脚本；  
           4) 创建测试场景：主要进行测试场景的设计和设置，包括监控指标的设定；  
    　 5) 运行测试场景：对已创建的测试场景进行执行，收集响应数据；  
           6) 分析结果：主要进行结果分析和报告工作。

## 13.简述loadrunner的并发测试

首先录制一个脚本，然后在脚本中添加一个集合点，打开loadrunner的第二组件，将启动一个用户执行一个用户的复选框取消，取消完后，进入第二组件，在工具栏中选择‘scenario’在下拉列表框中选择‘rendezvons’在弹出的框中设置集合并发的人数和时间，设置完成后，点击测试测试场景设置上升，持续，下降，时间，并启动执行。

## 14.简述loadrunner性能分析

**1  Average Transaction Response Time（平均事务响应时间图）**显示完成每个事务所花费的时间。观察事务响应时间以了解服务器在负载下对客户的响应时间。  
  
**2  Hits per Second（每秒点击次数）**  
“每秒点击次数”，即使运行场景过程中虚拟用户每秒向Web服务器提交的HTTP请求数。  
评估虚拟用户产生的负载量，和“平均事务响应时间”图比较，可以查看点击次数对事务性能产生的影响。通过对查看“每秒点击次数”，可以判断系统是否稳定。系统点击率下降通常表明服务器的响应速度在变慢，需进一步分析，发现系统瓶颈所在。  
**3  Throughput（吞吐率）**  
“吞吐率”显示的是场景运行过程中服务器的每秒的吞吐量。其度量单位是字节，表示虚拟用在任何给定的每一秒从服务器获得的数据量。  
可以依据服务器的吞吐量来评估虚拟用户产生的负载量，以及看出服务器在流量方面的处理能力以及是否存在瓶颈。  
“吞吐率”图和“点击率”图的区别：  
“吞吐率”图，是每秒服务器处理的HTTP申请数。  
“点击率”图，是客户端每秒从服务器获得的总数据量。  
**4  Transaction Summary（事务概要说明）**说明事务的 Pass 个数和 Fail 个数，了解负载的事务完成情况。通过的事务越多，说明系统的处理能力越强，失败的事务越少，说明系统越可靠  
**5  Running Vusers（负载过程中的虚拟用户运行情况）**  
此图反映系统形成负载的过程，随着时间的推移，虚拟用户数是如何变化的

## 15.简述loadrunner的ip欺骗功能（欺骗服务器）

简单来说，ip欺骗就是从同一台电脑发起的多个用户，使他们都有有不同的ip地址，可以让服务器认为这些用户都是从不同的电脑上发起的，欺骗电脑的拦截系统，从而进行测试的，使得测试场景更加接近于真实的测试环境

首先需要将测试机的ip地址设置成固定的ip地址；

3.2 然后在开始菜单中以管理员的身份运行HP loadrunner下的tools下的ip wizard.

3.3 在打开的窗口中有三个选项，分别是:

1. create new setting 设置新的ip欺骗场景

2. load previous setting fro 从文件中获取指定的ip地址

3. restore original setting 释放ip地址

默认情况下，都使用第一种

3.4 设置完毕后点击“下一步”，在窗口中的文本框中输入服务器的ip地址，便于服务器给客户端响应信息

3.5 输入完服务器的ip地址后，点击“下一步”，然后在弹出的窗口中设置ip地址:

from ip属性是设置虚拟ip地址起始

number to属性设置ip地址的个数

submask属性设置这些ip地址的子网掩码

设置完毕后，点击下一步。

3.6 当ip地址设置完毕后，点击“OK”，loadrunner自动生成设置的多个ip地址。

3.7 通过cmd后的ipconfig命令在dos窗口中察看是否生成了这些虚拟ip地址。

3.8 通过loadrunner的第二组件来实现让场景中的虚拟用户具备不同的ip地址。

3.9 在打开的loadrunner的第二组件中设置ip欺骗，在第二组件的导航栏中点击“scenario”, 然后点击“enable ip spoofer”,打开第二组件的ip欺骗功能（默认是关闭状态）

3.10 然后在loadrunner的第一组件中打开脚本代码，在代码中添加如下代码:

//验证IP欺骗代码

char \* ip ;

ip = lr\_get\_vuser\_ip();

if (ip)

lr\_output\_message("当前虚拟用户使用的IP为： %s", ip);

else

lr\_output\_message("[enable IP spoofing ]选项没有被启用！");

3.11 添加完代码后，在第一组件的导航栏中点击“vuser”，选择“run\_time\_setting”,然后在弹出的窗口中对loadrunner运行时的日志文件进行扩展.

3.12 接着在loadrunner的第二组件中点击"tools"，然后选择“export mode”,然后在点击“options”,在弹出框中选择“General”，选择“ip address allocation per thread”

3.13 然后运行测试场景，运行完毕后，点击测试场景右边的三个小人“vusers”,弹出框中会展示测试场景中所有虚拟用户，选中某一个虚拟用户，鼠标右键选择“show vuser log”，然后就可以查看到虚拟用户携带的ip地址信息。

## 16.简述loadrunner中常用的方法

1事务

2检查点（断言）

3参数化

4集合点

5 Ip欺骗

6关联

## 17.给你一个网站如何进行测试

**制定测试计划**

确定测试范围和测试策略，一般包括以下几个部分：   
功能性测试；界面测试；性能测试；数据库测试；安全性测试；兼容性测试   
**设计测试用例**  
功能性测试可以包括，但不限于以下几个方面：   
链接测试，链接是否正确跳转，是否存在空页面和无效页面，是否有不正确的出错信息返回等。

**提交功能测试**  
多媒体元素是否可以正确加载和显示。   
多语言支持是否能够正确显示选择的语言等。   
**界面测试**：可以包括但不限于以下几个方面：   
页面是否风格统一，美观；页面布局是否合理，重点内容和热点内容是否突出；控件是否正常使用；对于必须但为安装的空间，是否提供自动下载并安装的功能;文字检查   
**性能测试：**一般从以下两个方面考虑：   
压力测试；负载测试；强度测试   
数据库测试要具体决定是否需要开展。数据库一般需要考虑连结性，对数据的存取操作，数据内容的验证等方面。   
**安全性测试**：   
1 基本的登录功能的检查 2 是否存在溢出错误，导致系统崩溃或者权限泄露 3 相关开发语言的常见安全性问题检查，例如 SQL 注入等。4 如果需要高级的安全性测试，确定获得专业安全公司的帮助，外包测试，或者获取支持   
**兼容性测试**：根据需求说明的内容，确定支持的平台组合：浏览器的兼容性；操作系统的兼容性；软件平台的兼容性；数据库的兼容性   
开展测试，并记录缺陷。合理的安排调整测试进度，提前获取测试所需的资源，建立管理体系（例如，需求变更、风险、配置、测试文档、缺陷报告、人力资源等内容）。定期评审，对测试进行评估和总结，调整测试的内容。

## 18.完整的性能测试流程

测试准备：

需求分析文档  
建立性能目标  
了解应用软件  
了解软件部署平台  
建立系统负载模型  
制定项目计划和方案  
测试搭建环境:

测试机环境：jdk/edlipse/loadrunner or  jmeter a各种监控环境  
服务器环境：liunx系统/数据库/应用系统/各种监控工具  
测试脚本开发loadrunner和jmeter类同  
选取协议：选取采用的协议，为了测试变更流程  
使系统适应测试，去掉验证码  
增强脚本：参数化用户输入，关联数据，增加事务，增加检查点  
调试脚本：vugen单次回放  
           Vugen多次回访  
           Controller单脚本多用户  
           Controller多脚本多用户  
查看回放那个日志信息  
验证脚本：通过loadrunner检查点断言  
通过查看后台服务器日志验证  
通过测试系统查看运行侯变化  
利用sql查询插入/更新/修改效果  
测试数据准备：利用脚本自动成成数据  
               利用测试工具生成数据  
测试执行：设置加压运行场景  
监控操作系统：liunx监控：监控工具

监控命令：cpu,内存，磁盘，网络  
Jvm监控：jvm性别  
数据库监控  
Windows监控  
结果分析与调优  
测试后续跟踪

## 19.简述web输入框的测点

首先考虑是一个文本输入框还是一个数值类型的输入框

**文本框：**

非空

长度

非法字符

是否支持多行

字符中带有html标记对是，是否正常显示

复制，粘贴功能是否正常使用

**数值类型：**

小数

位数不够时是否自动补0，超出时是按四舍五入，还是直接舍去

整数

0是否支持，是否符合

负数

负数是否支持

范围

小数，整数，非数字，空格+数字

## 简述轮播图测点

**移动端：**

1.是否自动轮播（大于一张时）。

2.每张轮播图轮播的时间，一般3秒。

3.是否循环连续播放，首张至尾张，尾张至首张。

4.是否可手动滑动轮播，向左或向右。

5.首张是否可手动向右滑动至最后一张，尾张是否可手动向左滑动至第一张。

6.轮播图为一张时，不需要轮播效果，两张以上才需要。

7.轮播图的标题不可覆盖标识轮播图数量的标志，标题过长需省略，或把标题做成自动播放。

8.图片是否清晰，是否符合窗口比例。

9.轮播图对应的链接内容是否正确。

**Pc端：**

1. 是否自动轮播（大于一张时）
2. 每张轮播图轮播的时间，一般3秒
3. 是否循环连续播放，首张至尾张，尾张至首张
4. 鼠标悬停到轮播图时，是否后停止播放轮播图
5. 鼠标点击下方小圆点，是否切换轮播图
6. 鼠标点击是否进入到该界面
7. 轮播图为一张时，不需要轮播效果，两张以上才需要
8. 图片是否清晰，是否符合窗口比例

**后台管理系统：**

1.发布的轮播图是否和移动端对应。

2.不同权限管理员发布的轮播图是否显示在对应的的用户上。

3.轮播图可撤销，撤销之后可删除或编辑。

4.轮播图上传的顺序是否是按时间排序

5.轮播图最多可以上传几张

## 21.简述直播功能的测点

分为两个方面来考虑：**主播** 和 **粉丝**

**主播**：

首先直播界面是否可以正常打开

界面是否显示正常，有无错别字

有网，无网，以及弱网等测试

多次进出，是否会有黑屏，卡顿现象

首次使用是否会有页面导航

是否可以显示在线人数

是否有美颜，关于美颜等级（开美颜和关美颜app的反应等待时间）

是否可以正常使用直播（前后摄像头是否可以切换）

是否要下载特效插件

是否可以正常使用直播中的特效插件，查看礼物，以及刷礼物人的等级

直播中有pk，和连麦功能，是否正常使用

声音，以及音乐等是否正常

直播中是否有对敏感字符的屏蔽，对语言不文明的观看者是否有提示踢出直播间（禁言）

直播中有违反平台规定时，是否直接关闭直播间

观看直播人员的评论，主播是否可以看到

直播过程中有电话进入是否会停止直播

直播过程中出现不良行为，是否会封直播

直播过程中，网络差是否会卡掉直播等

直播间有卖货时，小黄车是否成功上架商品

**粉丝**：

首先也是是否成功打开该主播的直播房间

首次使用是否会有界面导航

界面是否显示正常，有无错别字，每个直播房间是否正常打开

是否会因为人数过多，进不去该房间

发表评论是否有限制（非法字符，敏感字符）

是否可以发表图片，表情等

刷礼物的限制

充值快币的充值方法（微信，QQ，支付宝等等）,限制（每次充值多少，每天最高上限。。）等

直播间有带货主播，点击小黄车是否跳转到购买界面（淘宝，京东，等等）

购买后是否可以在购买平台查看到购买订单信息

和主播连麦是否可以选择视频，语音等功能，功能是否可以正常使用。

主播直播间是否分享给微信好友，微信朋友圈，微博，QQ等等。。。

## 22.简述在线视频的测点

摄像头是否可以打开

摄像头是否可以切换前后摄像

是否可以从视频通话切到语音通话

是否支持群聊视频通话

画面是否正常  
声音是否正常  
点击静音是否可以静音  
点击扬声器是否正常  
点击挂断是否正常  
点击缩小窗口是否可以  
断网、信号弱、wifi、wifi网切换到数据网的一个状态测试

## 23.简述语音消息的测点

语音

**功能测试**：

点击是否可以录音

松开是否可以发送

点击上滑是否可以取消或转文字

转文字正确率是多少

语音最长发送时间是多少，最短是多少

点击发送的语音是否正常播放

长按发送的语音是否可以选择播放方式，收藏功能是否正常，提醒功能是否正常，是否可以转换文字，是否可以删除，多选，点击多选，选中语音内容能否转发

**界面测试：**

是否正常显示发送语音的时长

长按发送的语音是否会弹出浮窗，显示播放方式、收藏、提醒、转换为文字（仅普通话）、删除、多选

是否符合用户需求

**性能测试：**

发送语音需要多长时间

网络快慢对发送时间的影响

**兼容性测试：**

不同浏览器，不同的系统，台式电脑，笔记本等功能是否正常，页面布局是否正常；

相同浏览器的不同版本测试功能是否正常；

**网络兼容性：**不同网络下，能否正常使用；

**可用性测试：**在发送语音最后10秒有倒计时提示

## 24.简述分享朋友圈，微博，说说的测点

**分享到朋友圈的测点：**

没有微信这个app，第三方显示是否出现微信图标

如果有这个app可以出现，没有登录的情况下提示是否登录，是否注册

如果可以出现，没有登录的情况下提示是否登录，是否注册

没有微信这个app,点击微信分享链接，是否要跳转到微信下载界面（提示下载微信）

没有该软件时，是否引导跳转至应用商店下载该软件或者打开网页版

微信app安装好后，不登陆是否可以分享

点击分享，是否弹出分享平台图标，点击取消时，是否隐藏分享平台图标

分享到微信朋友圈，是否可以添加文字，表情，视频，图片等

分享成功后，是否返回该界面，

还是停留在微信，是否有提示（留在微信/返回该界面）

分享时网络不好，分享时间是否会改变

位置是否可以自动定位

取消分享时，不记录分享数

苹果手机是否和安卓手机能够通用

弱网状态下，是否能分享成功

好友是否能转发分享的内容

好友是否能正常打开分享的链接

当好友有该软件时，是否会自动跳转至该软件

**分享到微博的测点：**

1.没有微博这个app，第三方显示是否出现微博图标

2.下载是跳转到应用商店，还是先下载一个应用宝

3.微博app安装好后，不登陆是否可以分享

4.分享到微博朋友圈，是否可以添加文字，表情，视频，图片等

5.分享成功后，是否返回该界面，

6.还是停留在微博，是否提示（留在微博/返回该界面）

7.分享时网络不好，分享时间是否会改变

8.位置是否可以自动定位

9.取消分享时，不记录分享数

10.苹果手机是否和安卓手机能够通用

11.弱网状态下，是否能分享成功

12.好友是否能转发分享的内容

13.好友是否能正常打开分享的链接

14.当好友有该软件时，是否会自动跳转至该软件

**分享说说测点：（QQ）**

1.没有说说这个app，第三方显示是否出现QQ图标

2.下载是跳转到应用商店，还是先下载一个应用宝

3.说说app安装好后，不登陆是否可以分享

4.分享到说说，是否可以添加文字，表情，视频，图片等

5.分享成功后，是否返回该界面，

6.还是停留在说说，是否提示（留在说说/返回该界面）

7.分享时网络不好，分享时间是否会改变

8.位置是否可以自动定位

9.取消分享时，不记录分享数

10.苹果手机是否和安卓手机能够通用

11.弱网状态下，是否能分享成功

12.好友是否能转发分享的内容

13.好友是否能正常打开分享的链接

14.当好友有该软件时，是否会自动跳转至该软件

简述分享朋友圈、微博、说说的测点

1.点击分享，是否弹出分享平台图标;点击取消时，是否隐藏分享平台图标。

2.分享的内容是否正常显示在朋友圈，微博，说说中;

3.好友是否能正常打开分享的链接;

4.当好友有该软件时，是否会自动跳转至该软件;

5.没有该软件时，是否引导好友跳转至应用商店下载该软件或者打开网页版;

6.若不下载该软件，是否能显示分享的内容;

7.好友是否能转发分享的内容;

8.分享界面是否能够添加文字或表情

9.文字或表情测试

10.分享数为零时，不显示计数;大于零时，才显示计数。

11.当分享数过大时，达到千或万(有时百)， 需考虑简化分享数，如1k或1w (跟手机屏 幕也有关系)

12.取消分享时，不记录分享数。

13.苹果手机是否和安卓手机能够通用

14..点击应用图标，是否跳转至相应的应用

15.弱网状态下，是否能分享成功

# 四月份

## 电脑端测试和app测试的区别

想要知道APP端测试与web端测试的区别 ，那么我们就要先来了解，web和app的区别。  
web项目，一般都是bs架构，基于浏览器的，而app则是cs的，必须要有客户端。那么在系统测试测试的时候就会产生区别了。  
首先从系统架构来看的话，web测试只要更新了服务器端，客户端就会同步会更新。而且客户端是可以保证每一个用户的客户端完全一致的。但是app端是不能够保证完全一致的，除非用户更新客户端。如果是app下修改了服务端，意味着客户端用户所使用的核心版本都需要进行回归测试一遍。  
**1.从功能测试的来讲的话**

在流程和功能测试上是没有区别的。系统测试和一些细节可能会不一样。  
**2.性能方面**

web页面可能只会关注响应时间，而app则还需要关心流量、电量、CPU、GPU这些了。  
**3.兼容方面**

web是基于浏览器的，所以更倾向于浏览器和电脑硬件，电脑系统的方向的兼容，不过一般还是以浏览器的为主。而浏览器的兼容则是一般是选择不同的浏览器内核进行测试（IE、chrome、Firefox）。app的测试则必须依赖phone或者是pad，不仅要看分辨率，屏幕尺寸，还要看设备系统。

**4.相比较web测试，app更是多了一些专项测试：**  
　　一些异常场景的考虑以及弱网络测试。这里的异常场景就是中断，来电，短信，关机，重启等；而弱网测试是app测试中必须执行的一项测试。包含弱网和网络切换测试。需要测试弱网所造成的用户体验，重点要考虑回退和刷新是否会造成二次提交。需要测试丢包，延时的处理机制。避免用户的流失。这些在前面的弱网测试那篇已经讲过，这里不再讲了。安装、卸载、更新，现在app产品的用户都是使用的触摸屏手机，所以测试的时候还要注意手势，横竖屏切换，多点触控，事件触发区域等测试。  
　　web测试是基于浏览器的所以不必考虑这些。而app是客户端的，则必须测试安装、更新、卸载。除了常规的安装、更新、卸载还要考虑到异常场景。包括安装时的中断、弱网、安装后删除安装文件，更新的强制更新与非强制更新、增量包更新、断点续传、弱网，卸载后删除app相关的文件等等。

## 简述app和服务器之间的运行原理

app端（客户端）与服务端的交互其实理解起来很容易，客户端向服务器端发送请求，服务器端进行数据运算后返回最终结果。结果可以是多种格式，text文本格式，json格式，xml格式。客户端以接口的形式发送，服务端返回json结果。

## 简述通过fiddler抓包工具确定bug的前后位置

**Fiddler的介绍**

Fiddler是一款比较好用的抓包工具，测试人员经常用的使用工具，开发人员也在普遍使用，进行问题的定位分析，是非常有助于工作的一款工具。

**为啥要用Fiddler**

因为fiddler是测试中常用的抓包工具，一般在测试过程中，出现bug不清楚是前端还是后端的bug,这个时候我们一般使用fiddler工具来抓取请求信息，返回信息来进行分析是前端还是后端。

**Fiddler使用流程**

1启动Fiddler，打开菜单栏中的 Tools 选择  Options，打开“Options”对话框。

2在Options对话框点击https选项，进行勾选，切换到 Connections 选项卡，然后勾选Allow romote computers to connect后面的框，点击OK按钮。  
3在本机命令行输入：ipconfig，找到本机的ip地址。  
4打开链接的设备的“设置”->“WLAN”，找到你要连接的网络，在上面长按，然后选择“修改网络”，弹出网络设置对话框，然后勾选“显示高级选项”。  
5在“代理”后面的输入框选择“手动”，在“代理服务器主机名”后面的输入框输入电脑的ip地址，在“代理服务器端口”后面的输入框输入8888，然后点击“保存”按钮。  
6然后启动android设备中的浏览器，访问百度的首页，在fiddler中可以看到完成的请求和响应数据。

7.链接设备没有网络时，重新打开fiddler

8.有需要就在手机上下载一个证书，认证一下 ，

方法如下：

在链接的设备上的浏览器中，输入本机的ip地址加端口号（8888）弹出一个链接，点击链接，直接下载，下载好后进行填写信息就可以啦

**Fiddler如何来判断前后端的bug**

fiddler 是我们在测试中常用的抓包工具。我们一般使用fiddler工具来抓取请求信息，返回信息来进行分析是前端还是后端。

**第一种情况：**

fiddler 在没有设置过过滤器的情况下面没有抓到请求信息，可能是前端页面元素没有绑定事件，也有可能是前端发生了JS 错误，这就是前端的bug 。  
 **第二种情况：**

若抓取到的请求返回的结果错误，我们要确认一下，是否是前端传输的数据是错的，是的话就是前端的bug ，如果确定传值是正确的话，那就是后端的bug 。  
 **第三种情况：**

若抓取到的请求返回值中间的http 的状态码是500的话，说明是后端服务器一般的内部错误，那这就是后端的bug 。  
 **第四种情况：**

若抓取到的请求返回值中间的http 的状态码是404的话，说明可能后端服务器根本就没有对应地址的服务，当然还有可能是前端JS 提交请求的时候提交了错误的地址。

**简单举个例子：**

就那我们之前的一个电商项目来说吧，里面有一个付款功能，假如说购买一件连衣裙价格是188元，提交订单跳转到付款页面，显示200元，比显示价格多出12元，明显是出现bug，这个时候无法来判定bug是前端还是后端的,所以这个时候我们一般使用fiddler工具来抓取请求信息，响应信息来进行分析是前端还是后端的bug。

**如何判断：**

**首先看请求信息他的价格是不是188元，这个时候有两种情况**

1. （价格不是188元）就说明是我们的请求出现错误，app(客户端)出现的bug
2. （价格是188元） 就说明我们的请求没有出现出现错误，然后我们去看响应信息

**响应信息的价格是不是188元，这个时候有两种情况**

1. （价格不是188元，是200）就说明是我们的响应数据在后台结算时出现错误，bug就是出现在服务端。
2. （价格不是200元，是188）那就是服务器在后台结算时没有出现问题，bug还是出现在前端，说明响应数据在返回给app端时，是客户端解析显示的问题。

App (客户端) （c）

服务端 （s）

发送请求

App fiddler中间截获数据 服务端

返回响应数据

1. 请求出现错误 app（客户端）
2. 一旦响应出现错误 服务端 （s）
3. 付款方式有一个单独的后台管理

**举例2**

比如说我上家公司的一个线上问诊项目，在里面我负责的有一个修改宝宝信息的模块，我在app端修改宝宝的年龄，然后服务器就显示报错，我就使用fiddler抓取请求信息和响应信息，如果我在请求信息中显示修改数据正常。那么就是服务端的问题。如果请求信息中没有接收到请求信息或者出现路径路径、端口号、参数等错误，那么毫无疑问就是APP端的问题。还有就是我们抓取到的请求信息无错的情况下报错，说明是后端服务器的内部错误。还有种情况就是抓取到的请求信息与服务端的响应信息均正确的情况下仍然报错。那毫无疑问就是APP的错误。极有可能就是在接收响应信息时解析错误。

**举例3**

1就拿我们之前做的一款点餐APP来说吧，在付款方式这块，假如客户在提交付款的时候，付款方式在经修改后是4种付款方式，但APP显示付款方式仍是原来的5种，分不清是APP端出现的问题还是服务端出现的问题。这个时候，我就会搭建一个feddler桥梁，然后检测他是APP端的问题还是服务端的问题。当APP端发出请求，在feddler里显示的请求信息是4种付款方式，那说明就是服务端出现的问题。  
当APP端发出的请求中，路径、参数信息、请求头均没问题的时候，那也就是服务器出现的问题。反之就是APP端出现了问题。  
当APP端发出的请求信息中没有问题，而服务器端再返回的响应信息中出现问题，那就是服务器端出现的bug。  
当服务器端在返回响应信息的时候，没有返回响应信息，也就是服务端出现的bug。  
还有一种可能就是，服务器端返回的响应信息也都没有问题的时候，还是出现bug，那么就是APP端在接收到响应信息解析时，显示出现问题，那么就是APP端的bug。

## **4.简述bug的属性**

关于bug的属性主要有：

所属产品：就是这个bug属于我们这个产品的名称。

所属模块：这个bug属于这个项目中那一个模块或是子模块

所属项目：就是我们这个项目所在的名称

影响版本：是当前大模块，还是分支的小模块

当前指派：bug指派给谁去修改

截止日期：bug修改的截止日期

Bug类型：什么类型的bug （代码，界面，脚本等）

Bug标题：在测试的那一环节出现错误

严重程度：分为（1致命，2严重，3一般，4优化）

优先处理级别：分为（1优先，2中等，3一般，4可等待处理）

重现步骤：bug的重现步骤，详细说明

相关需求：

相关任务：

抄送给谁：这个bug抄送给谁

关键词：bug的关键字

附件：

标题：

## **5.简述bug的状态都有哪些**

1 new (新的)：新发现的一个bug，跟项目负责人确认确实是一个bug，记录起来，状态设为“新的”  
2 assigned (已指派的)：确定为bug后，指派给专门开发人员处理，状态就为“已指派的”。  
3 open (打开的)：开发人员打开并开始处理bug，状态就为“打开的”。  
4 fixd(已修复的)：开发人员已经将bug修复解决完后，状态为“已修复的”  
5 pending reset(待再测试的)：开发人员修复完成后再次返回给测试部门，状态为“待再测试的”  
6 reset(再测试的)：返回来的修复好的bug给某位测试人员进行再次测试，状态为“再测试的”  
7 declined(拒绝)：拒绝修改的bug  
8 defrred(延期)：不在当前进行修复，延期进行修复的bug  
9 closed(关闭)：bug已被修复成功。

## **6.bug严重级别的评判标准**

对于bug的严重级别分为以下4种：

**1（致命）：**

致命的bug就是说这个bug危害到别的模块，自己本身这个模块不能使用，还牵连到别的模块也不能使用，我们称之为致命bug。

**2（严重）：**

就是说自己本身这个模块不能使用，出现错误，但是并不牵连到别的模块，我们称之为严重bug。

**3（一般）：**

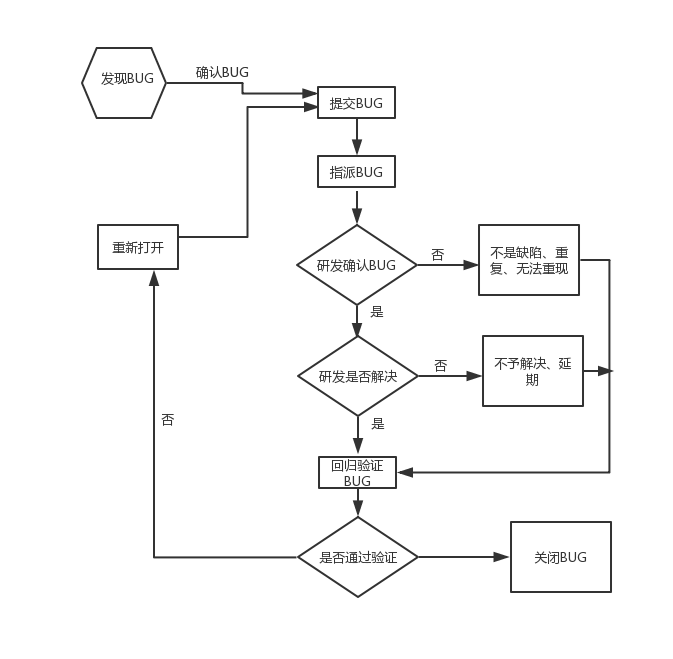
对于一般bug就是说，这个bug在功能上有一些问题，比如说文本框，按钮不能正常使用的一些问题，我们称之为一般bug。

**4（优化）：**

优化类的bug一般都是一些，界面有错别字，界面不美观，不符合大众的审美等，对于这样的bug我们称之为优化类bug。

## **bug的生命周期**

对于bug的生命周期首先就是发现BUG ，然后反复确认是否BUG，如果是，提交BUG，指派BUG给相对应的开发人员，开发确认是否是一个bug，是就去修改解决，（不是，就驳回，不给处理）然后就是回归验证bug，进行回归测试，验证bug是否通过验证，通过就关闭bug（不通过就提交，反复测试）



## **你印象较深刻的两个bug**

**案例1**

之前测试过一个小程序，出现一个bug，当时是一个功能在汉语的情况下可以正常使用，但是切换到汉语繁体的时候却不好使了，系统识别不成功，当时不知道会是因为切换到汉语繁体的原因，都以为是整个输入法的原因，整个项目组进行讨论，反复测试，加班到很晚，没想到会是繁体的原因，所以影响特别深刻。

**案例2**

1.对于印象比较深刻的bug记得之前在公司做过一个腾讯视频的项目，在缓存视频中有一个bug印象比较深刻，用户手机上链接无线网络，同时打开数据，缓存视频在无线网络下进行缓存，在缓存过程中突然网络断开，这个时候自动切换到数据，是否提示用户用数据缓存。

2.然后再次连接无线网络是否提示用户使用无线网缓存，如果没有提示，使用移动数据缓存后扣费，会使得用户体验度下降，对软件进行不满。

3.主要是测试验证从 无线 切换到 数据 从 数据 切换到 无线 之间的一个过程，最后会不会自动再次切换到无线的一个测试。

**案例3**

比如说还有一个淘宝订单的，下单都有一个备注留言，这个其实是可写可不写的，当时测试了非空验证这个测试，居然不可以非空，就是一个影响很深刻的bug

## 简述在测试工作中如何使用svn

SVN是一个开放源代码的版本控制系统，通过采用分支管理系统的高效管理，简而言之就是用于多个人共同开发同一个项目，实现共享资源，实现最终集中式的管理。

比如：开发会把自己的代码放到SVN中，你要测试的时候可以从SVN里下载最新的代码到本地进行测试，同时你的测试报告等也可以上传的SVN和其他的测试人员进行交流。

## 简述svn版本冲突产生的原理以及解决方法

**原理：**

假设我们a,b两个文件同时是在svn，down出来的，版本号都是7，此时我们修改a文件，行1：1234改为12345提交，此时版本库里会有一个新的版本8，而且会有一个记录，记录大致是这样的：1234 ->12345，我们还有b文件，b的版本号此时是7，如果直接update，svn会根据这个记录对现有的b进行修改，将原有的1234->改为12345。冲突是怎么发生的呢，是这样的，如果我们在update之前修改了b，1234->12346 svn更新的时候，先看自己的版本号再与最新的比较如果是相同则不做变动，如果不同会拿着更改日志对b文件操作，当他操作的时候发现修改日志中要求1234->12345但是 本地确改为了12346svn就会把这种情况看作是冲突。  
**解决方法：**（1）合理分配项目开发时间  
 甲：上午开发  
 乙：下午开发  
（2）合理分配项目开发模块，为每个人分配单独的模块  
 甲：负责开发购物车模块  
 乙：负责开发文章模块  
（3）通过SVN解决版本冲突问题  
            （1）更新服务器端数据到本地  
            （2）删除除index.php以外的其他三个文件  
            （3）修改整合index.php冲突文件  
            （4）重新提交数据到SVN服务器端，即可解决版本冲突问题

## 11.简述svn结合eclipse对项目的管理工作

在常用的工作中我们经常使用svn结合eclipse来对项目进行管理工作，通常就是开发提交项目在svn里进行管理。

安装SVN插件，拿到SVN路径，import检出项目，修改文件，提交，产生冲突，资源库同步查看本地代码资源库代码有何不同，上传，正确后把项目达成war包，配置在Tomcat中

## 12.同时有多个项目应该咋办

首先和项目经理了解项目的进度 有没有正在测的测试人员  分优先级 进行排期  不太着急得到项目会放在加班的时间来测 对于特别着急的在代码错误的地方，我会及时和开发沟通，对于不太着急的项目 一般进度都是在中后期 基本上代码出错的可能性不是很大的 所以我会放在加班的时候去测，但是基本上每个测试人员手上不太有太多的项目 ，因为项目过多的话 ，会相应的降低产品的质量