

前提背景

未上班的同学，进行系统复习

复习：3~5天，根据实际情况调整

过一下之前课表上的知识点，划分一下重点（面试、工作）

划重点 ---> 记住，抽空投递简历

（约面试，

线下 --- 自己面试，所有需要学员自己背诵/记住知识点

线上 --- 老师帮忙面试，（并不是所有人都能约到线上面试）

一、Linux：centos版本

1.1 在工作中，使用Linux都做过什么？

①、搭建环境（有环境才能部署服务）

②、定位BUG，通过查看服务日志去定位

日志：请求入参--->处理过程---->输出结果

举例：计算10x10 ----->10x10 ----->把100输出

关键字：error，exception，fatal error-一般是代码错误，语法错误 exception-一般是系统出现故障，eg.空指针异常、没有定义对应的类 fatal-致命

③、监控服务器系统资源，top、df、free

1.2 常用Linux命令（至少15个以上）

101.33.250.42

root/Cxy2022.?fw

文件基本操作：ls、ll、pwd、mkdir、cd、touch、cp、mv、rmdir、rm、cat、more、less、head、tail -f

用户与用户组：groupadd、groupdel、useradd、userdel、passwd、usermod、id

文件属性：chown、chgrp、chmod

查找命令：which、whereis、find

文件内容操作：grep、wc、sort、diff、cut、

压缩打包：tar、zip

监控资源: top、free、df -h

查看进程: ps、kill

二、数据库: MySQL

2.1 面试以及工作中常用(90%) 数据的操作:

测试自己写 查询语句 (70%)

修改、新增、删除 语句, 大多数情况下都是开发写, 少许 (30%) 情况是自己写

2.2 数据操作的sql语句

sql语句: 固定格式, 类似汉语的主谓宾, 英语的语法

SELECT distinct 字段(*) from 表名 where 条件 group by 字段 ORDER BY 字段

distinct 去重 结合计数来使用

ORDER BY 排序 limit n 前n个

group by 分组

【面试可能还会问 增加、修改、删除数据】

insert into 表名(字段1, 字段2) values (数据值1, 数据值2)

update 表名 set 字段1=值1,字段2=值2 where 条件

delete from 表名 where 条件

2.3 数据库什么时候会用到

①、搭建环境的时候, 执行开发提供的sql语句: 创建数据库、创建表 (增加数据)

②、测试过程中, 验证测试结果

注册成功--->去到数据库中查询注册的账号是否存在

订单付款成功, 但界面提示失败---->去数据库查订单状态 (数据库的数据有问题, 都是后端开发BUG)

③、做性能测试时, 需要造大量数据

存储过程

2.4 数据库的练习题:

数据库: 106.52.182.140 账号: root 密码: Cxy199807.

一、操作题

以下1~6题依赖students_yx、scores_yx、courses_yx 表

- 1、查询张飞的课程和成绩：
- 2、查询没有成绩的学生信息：
- 3、查询所有北京人的信息以及成绩：
- 4、使用 left join 查询诸葛亮的所有分数：
- 5、查询3班的总分与1班的总分之差
- 6、使用右连接查询河南省的所有市和区：（areas_b 表）

以下7~11依赖dep表

- 7、查询所有比湖南人工资大的人：
- 8、查询比广东技术的平均工资大的省份信息：
- 9、查询男销售的平均工资与女销售的平均工资之差（dep表）
- 10、查询所有销售的老乡（dep表）
- 11、查询所有比江西技术年龄大的广东经理（dep表）

二、假设有这么一个表：

表的字段如下：ID、姓名、年龄、性别、部门、工资、省份、备注

- 1、查询公司中来源于河南省的员工姓名
- 2、查询备注为空的员工信息
- 3、查询公司中有哪些员工跟飞哥同姓？（注意别写错别字）
- 4、查询下公司总共有多少人？总数
- 5、查询下该公司的工资开销，求和、求平均值
- 6、查询公司的员工都来自于哪些省份？需要去重
- 7、查询公司工资最高的前三位员工的姓名
- 8、查询公司中最大年纪和最小年纪、平均年龄分别是多少
- 9、查询每个省份不同性别的人分别有多少？
- 10、查询哪些省份的员工可以在公司内找到老乡？
- 11、查询表中重复的姓名及其数据量？

参考答案：

一、操作题

- 1、SELECT * FROM students_yx st,scores_yx sc,courses_yx co WHERE st.studentNo = sc.studentno AND sc.courseNo = co.courseNo AND st.name = '张飞'

2、SELECT * FROM students_yx st left join scores_yx sc on st.studentNo = sc.studentno WHERE sc.id is null
3、SELECT * FROM students_yx st ,scores_yx sc WHERE st.studentNo = sc.studentno and st.hometown='北京'
4、SELECT st.name ,sc.score FROM students_yx st left join scores_yx sc on st.studentNo = sc.studentno WHERE st.name = '诸葛亮'
5、查询3班的总分： SELECT sum(sc.score) from students_yx st ,scores_yx sc WHERE st.studentNo=sc.studentno and class ='1班'

查询1班的总分： SELECT sum(sc.score) from students_yx st ,scores_yx sc WHERE st.studentNo=sc.studentno and class ='3班'

匹配结果： SELECT (SELECT sum(sc.score) from students_yx st ,scores_yx sc WHERE st.studentNo=sc.studentno and class ='1班')-(SELECT sum(sc.score) from students_yx st ,scores_yx sc WHERE st.studentNo=sc.studentno and class ='3班')

6、SELECT * from areas_b a RIGHT join areas_b b on a.pid=b.aid RIGHT JOIN areas_b c on b.pid =c.aid WHERE c.atitle='河南省'

7、查询所有湖南人的工资： SELECT salary from dep WHERE address ='湖南'

结果匹配： SELECT * from dep WHERE salary > all (SELECT salary from dep WHERE address ='湖南')

8、查询广东技术： SELECT avg(salary) from dep WHERE address='广东' and job='技术'

匹配结果： SELECT address,salary from dep GROUP BY address having salary >(SELECT avg(salary) from dep WHERE address='广东' and job='技术')

9、查询男销售的平均工资： SELECT avg(salary) from dep WHERE sex='男' and job='销售'

查询女销售的平均工资： SELECT avg(salary) from dep WHERE sex='女' and job='销售'

匹配结果： SELECT (SELECT avg(salary) from dep WHERE sex='男' and job='销售')-(SELECT avg(salary) from dep WHERE sex='女' and job='销售')

10、查询销售的家乡： SELECT address from dep WHERE job='销售'

匹配结果： SELECT * from dep WHERE address =any (SELECT address from dep WHERE job='销售')

11、查询江西的技术年龄： SELECT age from dep WHERE address ='江西' and job='技术'

匹配结果： SELECT * from dep WHERE age > all (SELECT age from dep WHERE address ='江西' and job='技术') and address ='广东' and job='经理'

二、假设有这么一个表

1、select 姓名 from 表名 where 省份 ="河南省"

2、select * from 表名 where 备注 is null

3、select * from 表名 where 姓名 like "侯%"

4、select count(id) from 表名

5、select sum(工资) from 表名

6、select distinct(省份) from 表

7、select 姓名 from 表 order by 工资 desc limit 3

8、select max (年龄) ,min(年龄),avg (年龄) from 表

9、预期举例： 广东 男 12 广东 女 7 北京 男 4 上海 女 5

select 省份,性别,count(性别) from 表名 group by 省份,性别

10、预期举例： 省份数>=2

select 省份 from 表 group by 省份 having count(*) >= 2 11、预期举例： 张三 3 李四 6 王五 24

select 姓名,count () from 表 group by 姓名 having count() > 1

三、测试流程

3.1 测试依赖：产入、产出

产入：需要项目组其他成员提供资料（原型、需求文档、接口文档、功能点、项目资料等等）给到测试，才能开展测试工作。（如果公司的流程不完善，只能通过沟通交流去获取资料

产出：测试工作完成后，需要输出结果文档。测试点、测试用例、测试报告/性能测试报告

3.2 标准化的流程

由产品提供需求文档/产品原型 --->

需求评审大会（目的是开发跟测试对项目的需求理解一致） --->

测试计划 ---> 测试用例编写 ---> 用例评审 --->

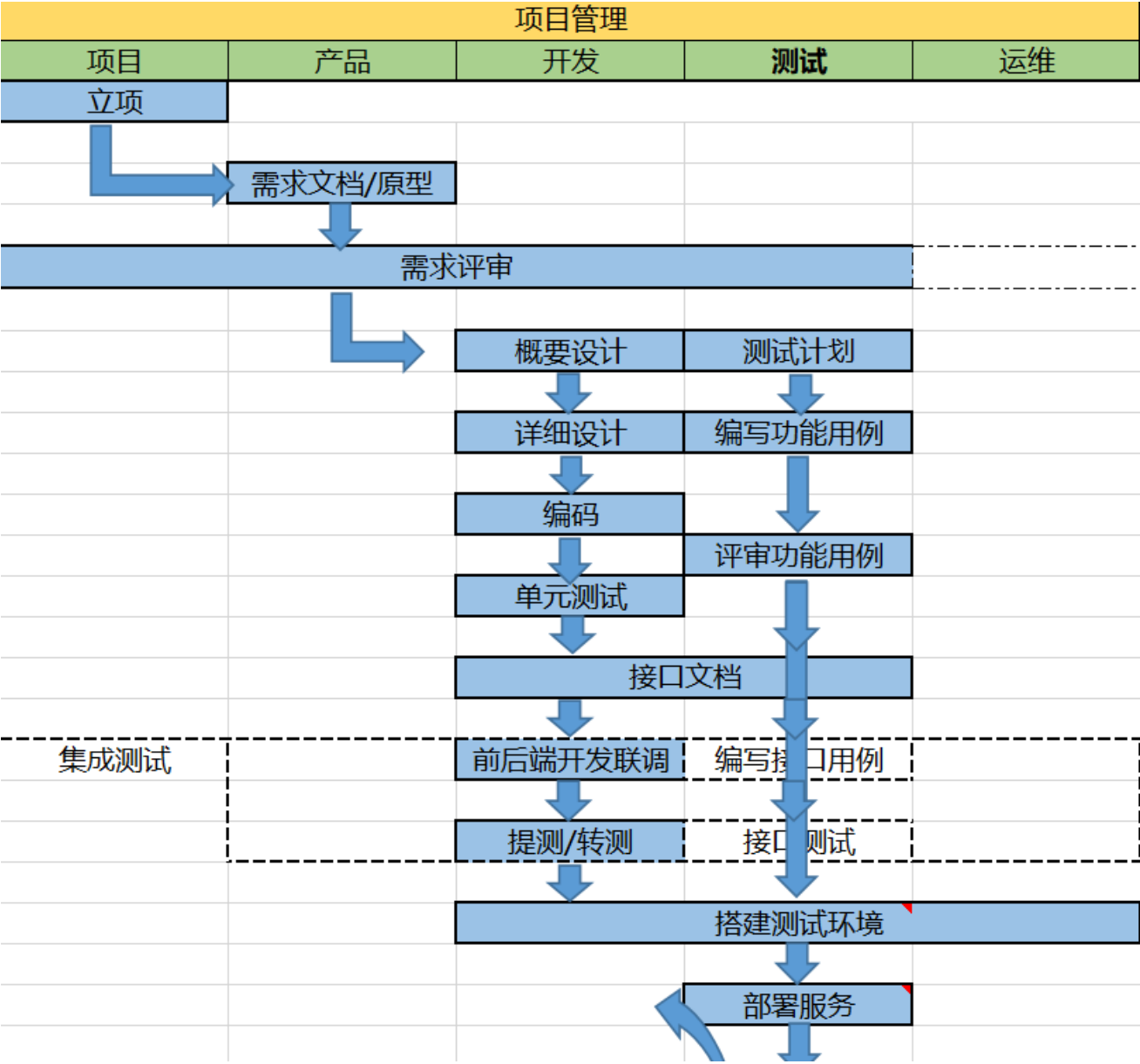
(后台开发提供接口文档--接口测试，可以说或有时候批次太小或时间紧急就没做)---

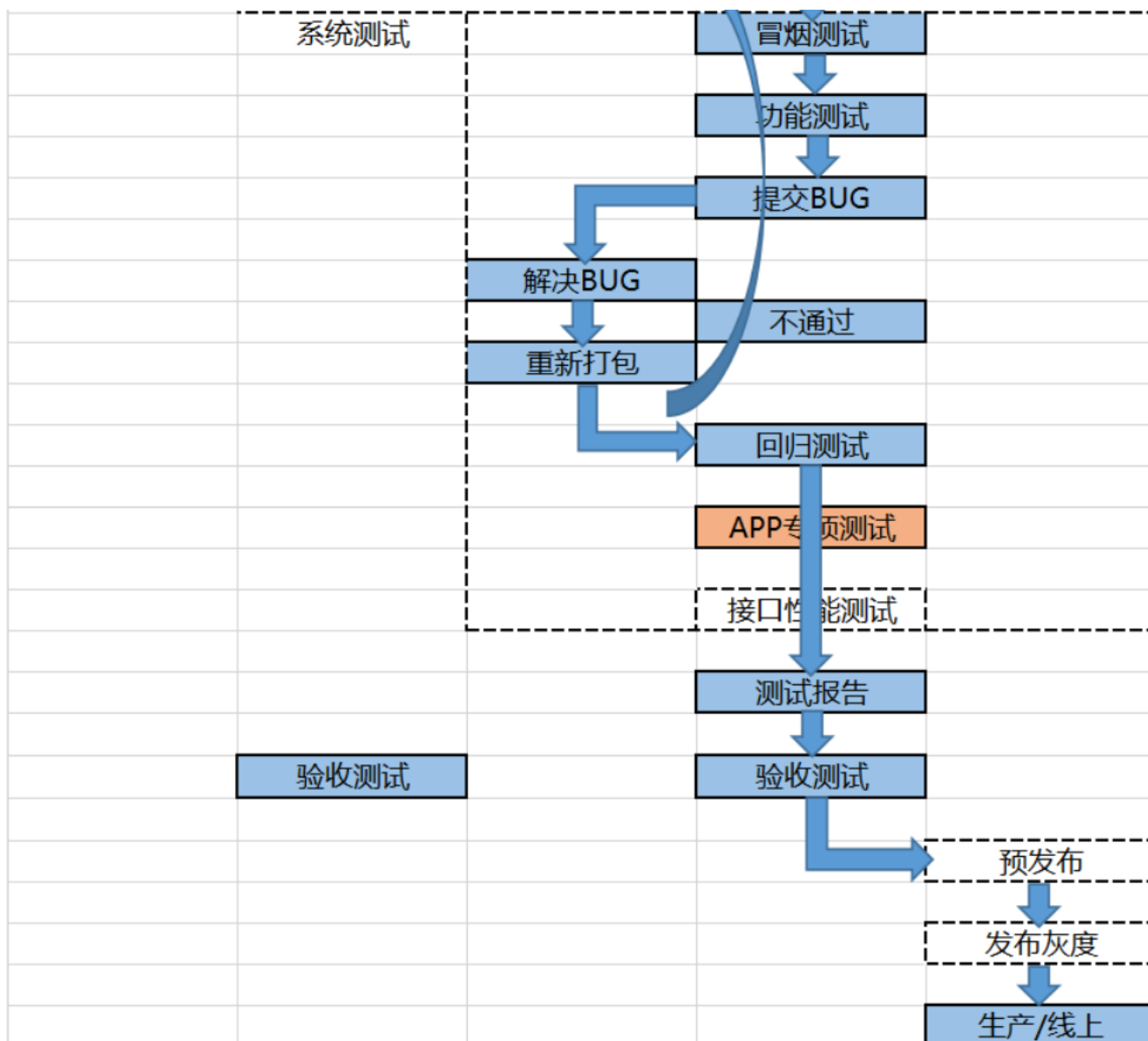
冒烟测试 ---> 通过就进行功能测试(不通过就打回开发修改) --->

提bug/开发修复bug/回归验证 --->

2~3轮测试后达到上线标准(123级BUG100%修复，4级BUG修复95%以上) ---> 测试报告 ---> 上线

(实际工作中要结合公司的现状开展工作，自己做好笔记)





3.3 找老师问问题：

不要给老师出问答题，而是要出选择题

选择题：学生自己是有思路，是有想法的，但是不确定哪个更合适，老师可以帮忙选择。

四、测试用例设计方法

4.1 设计方法

等价类、边界值、错误推测法、判定表法、因果图、正交实验法；

对于测试用例编写来说，常用的四种方法基本就够用了：等价类、边界值、正交实验法、错误推断法，需要辅以场景测试法、需求/设计转换法、探索式测试思想，可以应付绝大多数产品的测试。

个别的产品还需要在某一点细化和扩充，需要就事论事。

4.2 测试用例颗粒度（粗细）

按公司原来的做法/模板 来开展 ---- > **抄作业**

说辞：组长，咱这边的测试用例颗粒度要写多细，有没有之前的用例给我学习下？我好把握程度
拿过来之后参照下思维/风格。

4.3 测试用例场景考虑

（最好自己心里有谱：除了流程、功能，**任何场景都需要考虑专项测试**）

80%+ 是功能（流程）

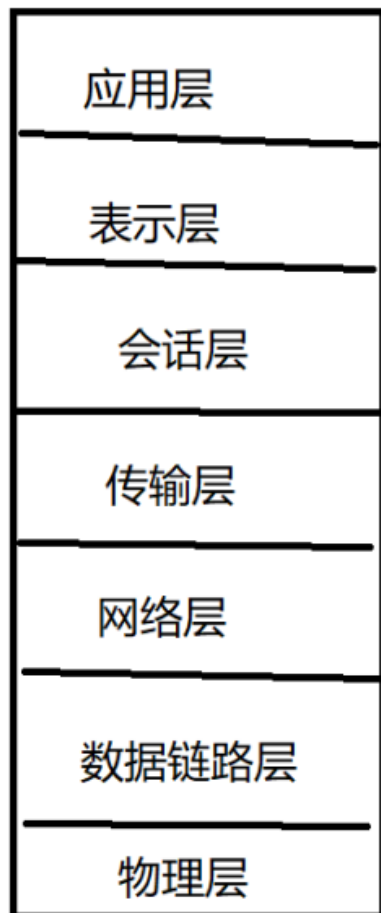
20% 专项测试（兼容、网络、权限、交互、安全、稳定性、性能....

学习测试思维：XXX功能模块你怎么测？（测试点，需要考虑哪些）结合自己的项目

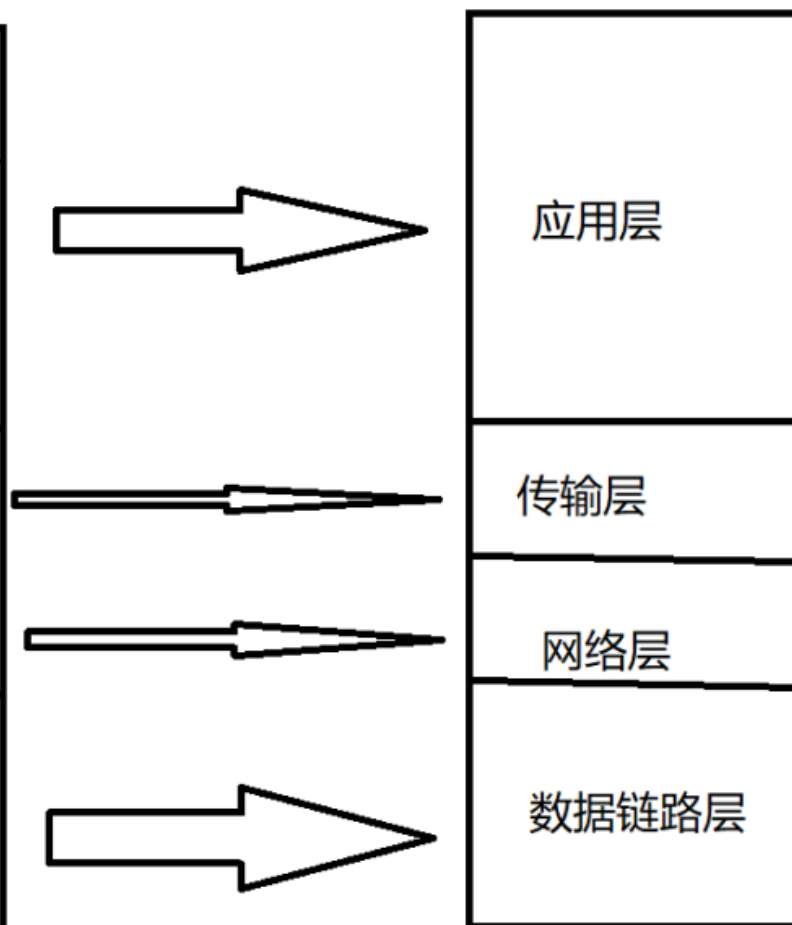
五、协议与抓包

5.1 七层与四层

七层网络协议模型



TCP/IP 四层模型



<u>OSI七层网络模型</u>	<u>TCP/IP四层概念模型</u>	对应网络协议
应用层 (Application)	应用层	<u>HTTP</u> 、 <u>TFTP</u> , <u>FTP</u> , NFS, WAIS、 <u>SMTP</u>
表示层 (Presentation)		Telnet, Rlogin, SNMP, Gopher
会话层 (Session)		SMTP, DNS
传输层 (Transport)	传输层	<u>TCP</u> , <u>UDP</u>
网络层 (Network)	网络层	<u>IP</u> , ICMP, ARP, RARP, AKP, UUCP
数据链路层 (Data Link)	数据链路层	FDDI, Ethernet, Arpanet, PDN, SLIP, PPP
物理层 (Physical)		IEEE 802.1A, IEEE 802.2到IEEE 802.11

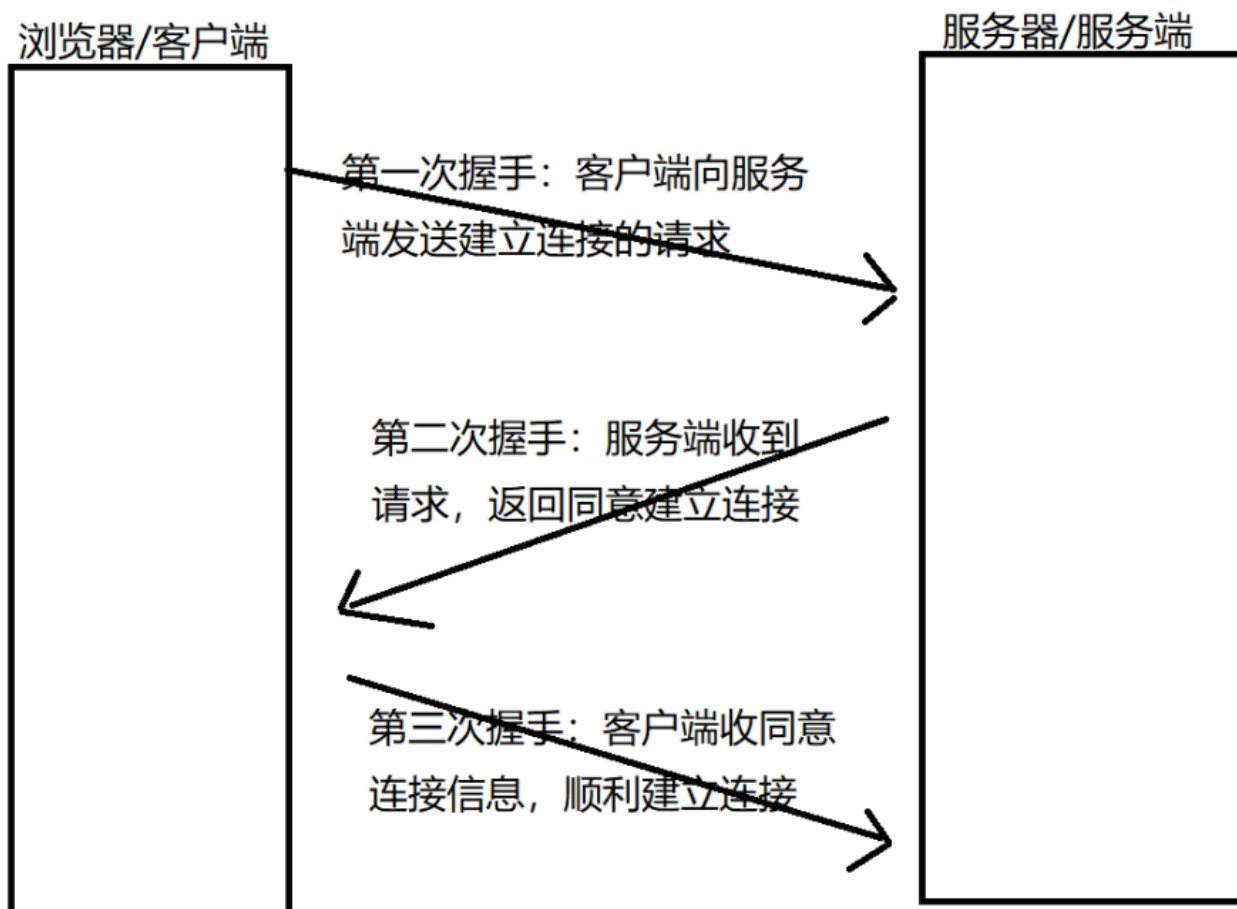
5.2 三次握手与四层挥手

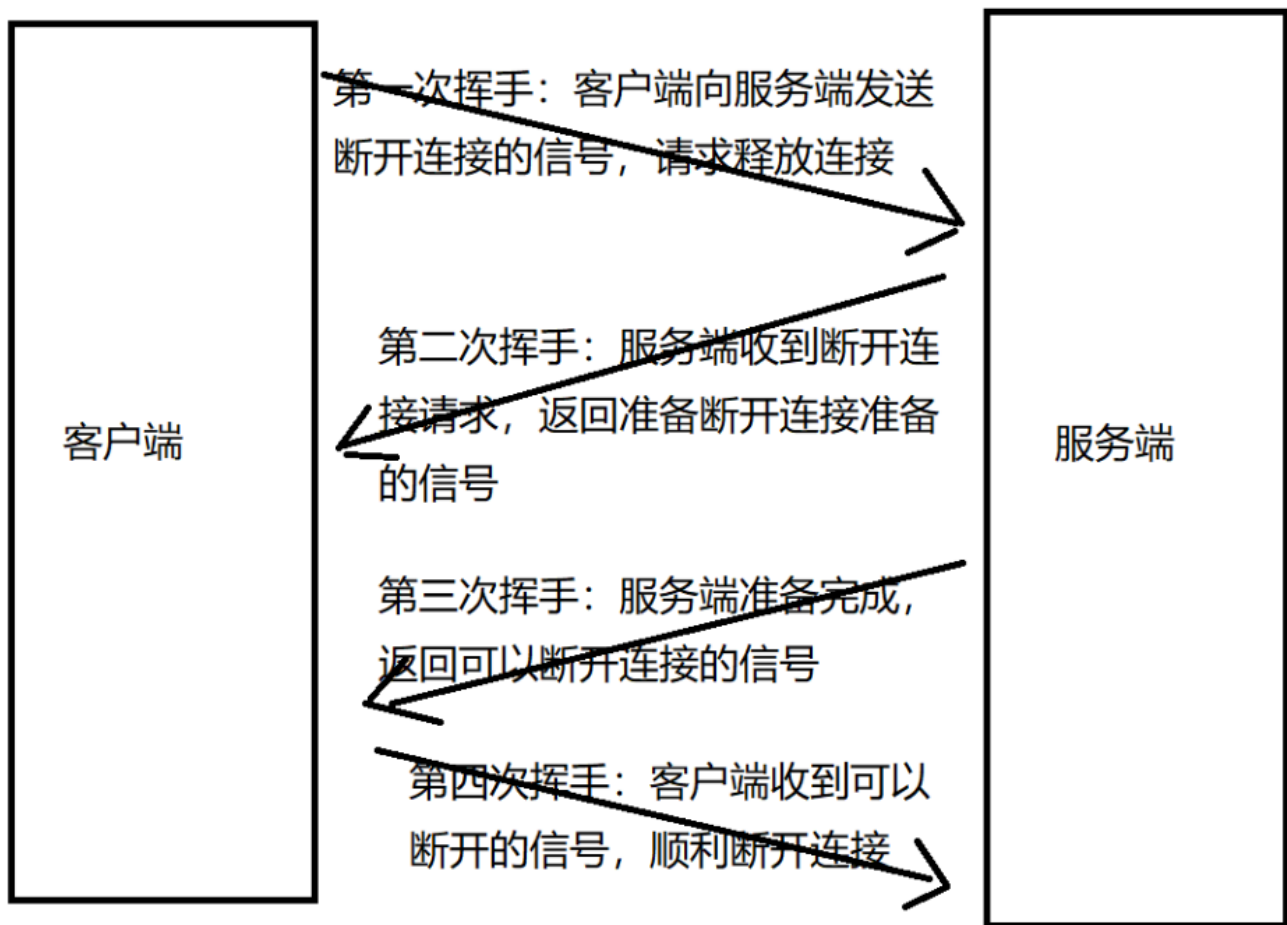
面试问题（除非我们非常清楚问题答案，不然都统一要求复述面试官的问题）：

在访问xx页面的一瞬间，发送了什么？ ---建立连接（三次握手）

在关闭页面时，网络层面是怎么处理的？ ---断开连接（四次挥手）

（面试官有些时候，也**没有表达清楚面试问题**，自己也不清楚自己想问什么）





5.3 TCP与UDP的异同点

TCP UDP

相同点: 都是在传输层

不同点:

连接: 面向连接 非面向连接 (面向是需要确认的, 非面向不需要确认)

传输可靠性: 可靠 相对不可靠

使用的场合: 传输大量数据 传输少量数据

传输速度: 速度相对慢 速度快

5.4 GET与POST请求的区别

get请求参数包含在url中, post通过request body传递参数。

get请求参数长度有限制 (这是因为不同浏览器对url 有限制的愿意), 而post传参理论上是没有大小限制的。

get请求只能进行url编码, post支持多种编码方式。

get 比post请求更不安全, 因为它的参数直接暴露在url上面, 所以不能用get传递敏感信息。

5.5 请求与响应

请求：请求行、请求头、请求正文

（发送的请求，具体的内容，是需要根据**后端开发定义要求（接口文档）**来的）

响应：响应行（响应状态、响应状态码）、响应头、响应正文

5.6 常见的响应状态码

200 OK 表示请求被服务器正常处理

301 Moved Permanently 永久重定向

302 Found 临时重定向，表示请求的资源临时搬到了其他位置

400 Bad Request 表示请求存在语法错误或参数错误

404 Not Found 表示服务器找不到请求的资源

500 Internal Server Error 表示服务器执行请求的时候出错了

5.7 抓手机APP的https的请求

fiddler跟APP同一个局域网

fiddler：设置伪造证书，开启允许抓取https请求

fiddler设置远程代理，开启默认代理端口号8888

APP：手动修改网络，设置代理，主机是电脑的IP地址，端口是fiddler的端口号

通过浏览器访问ip+端口号，下载安装证书

5.8 手机APP弱网测试

fiddler跟APP同一个局域网

fiddler：设置远程代理，开启默认代理端口号8888

Ctrl+R 打开fiddler的配置文件，找到网络上下行(xing)网速进行设置（m_SimulateModem）

设置好后，在菜单中勾选配置生效

APP：手动修改网络，设置代理，主机是电脑的IP地址，端口是fiddler的端口号

5.9 安全测试与断点篡改请求/响应

安全测试：专门的安全测试是由安全测试小组（个别公司是有运维/开发进行）进行的，我只会简单的安全测试。

简单的安全测试：

- ①抓包 查看请求跟响应，敏感关键信息有无明文展示（身份证号码、银行卡号、密码、家庭地址、手机号.....）
- ② 加密方式的简单破解，抓包去进行md5解密，是否可以轻易解密；
- ③抓包篡改，能篡改成功，证明存在安全隐患。

篡改请求：

第一步：在客户端最后确认操作前，开启请求断点：快捷方式F11

第二步：在客户端进行操作

第三步：fiddler断点拦截到了请求，进行篡改/修改

第四步：取消断点

第五步：放行、发送请求

篡改响应：

第一步：在最后确认操作前，开启响应断点：快捷方式 Alt+F11

第二步：在客户端进行操作最后一步

第三步：fiddler断点拦截到了响应，在响应处进行篡改/修改

第四步：取消断点

第五步：放行、发送响应

六、接口测试

6.1 接口测试的目的？或者 接口测试的意义？或者 为什么要做接口测试？

- 1) 越底层发现BUG，修复成本越低。举例：盖楼房的时候，可以提早测试地基，越早发现问题，修复成本越低。
- 2) 缩短项目周期。接口测试是针对后台代码，所有的BUG都是后台开发的。接口测试阶段把后台BUG都解决了，后续的集成测试(功能测试，基本都是前端BUG)
- 3) 接口测试可以检查系统的安全性，基于前端传参不可信，可以通过接口向服务端发送反向参数去验证系统。比如充值负数的金额。
- 4) 系统越来越复杂，接口测试相对功能测试，更容易发现底层(后台)BUG。
- 5) 接口测试相对实现自动化，而且相对UI自动化也是更稳定的。

6.2 接口文档

面试：之前你们的接口测试怎么做？给你一个接口怎么测试？怎么做好接口测试？

回答：根据开发提供的接口文档，进行测试点的分析，主要是考虑正常场景与异常场景，

正常场景，条件的组合，参数 的格式校验等价类/边界值；

异常场景，多一个参数，少一个必填参数，参数为空；

接着编写测试用例。再使用接口测试工具（jmeter/request）来执行 接口测试用例

6.3 接口测试工具

jmeter

request

接口测试，只是简单把脚本跑起来，去验证各种正常/异常的场景。每跑一下脚本，都可能需要去修改请求参数，来验证不同的场景

举例：

账号：13466668888

密码：符合、不符合规则

七、接口自动化

7.1 自动化

写好的脚本，在不需要人工干预的情况下，可以持续一直定时运行，都能跑起来

那么就需要使用上：参数化、动态关联、断言，最后使用jenkins定时执行jmeter脚本

7.2 自动化在什么阶段做？或者说在哪个环节流程做？

自动化的目的：是使用脚本来代替测试人工回归，验证之前版本/批次的功能是否正常，我们新的版本/批次的功能新增/改动，没有影响到原来的功能。

自动化实现的阶段：在每个版本/批次完成上线后，需要新增/修改自动化脚本，以便于在下个版本/批次时使用，来验证新版本/批次，没有改动到旧功能

ABCDEF版本

3.1 上线、写自动化脚本A ---- 3.2要测试，就跑原来的自动化脚本A，确保原来功能没有影响，上线后，又要写自动化脚本B ---- 3.3要测试，就跑原来的自动化脚本B，确保原来功能没有影响，上线后，又要写自动化脚本C

7.3 接口自动化，运行脚本的需要多少时间？

1个接口花费时间，十几毫秒~500毫秒以内，平均 200ms

1s能跑5个接口

1个项目有多少接口用例？小项目300~500、中型项目500~1500，大项目1500+

跑100个用例，需要20s

小项目：100秒、中项目：15*20=300s，约5分钟、大项目：3000接口用例，也只花了10分钟

（存在一个接口，对应几个功能）

7.4 jmeter接口自动化

①参数化：包含**随机参数化**，**用户参数**、**文件参数化**。

参数化一般针对具有统一性的数据等进行参数化（比如host、注册新增或者修改的账号密码、手机号），目的是为了后续统一进行修改

②动态关联：是由于存在一些接口之间相互关联，即后面的接口发送请求时需要依赖前面接口返回的响应。（比如登陆后的cookie/token、下单后的订单号、待处理列表中的id、（审批流程的业务单号）可以使用**边界值**、**正则表达式**从前面接口提取出来给到后面的接口使用。如果是JSON格式的响应可以直接使用**JSON提取器**

③断言：使用断言，让脚本自行去判断实际结果与预期结果是否一致，比如响应断言、JSON断言。

编写好jmeter脚本后，需要使用Jenkins+ant每天定时执行jmeter脚本，并发送邮件，来确保接口没有异常报错。

Jenkins地址：<http://106.13.221.67:8080/jenkins/login>

账号密码：admin/admin888

服务器IP：106.13.221.67

账号：test01、test02、test03（其中之一）

密码：Cxy2022.?fw

7.5 pytest+request+pytest框架

使用的是python+request+ pytest框架来完成接口自动化测试，框架是早就搭建好了的，我这边主要是负责后面批次迭代，涉及到的接口测试用例脚本的编写新增/修改，其实我自己也能在个人电脑参照着也能搭建起这个框架

框架主要包含配置，数据，公共方法，测试用例，报告，日志等模块：

config：配置主要存放IP地址、域名路径、数据库的连接配置

common：主要包括读取excel文件、数据库的连接、读取配置文件、打印日志等操作公共方法

data：数据层，主要是存放接口测试所使用到的数据

run_case：执行测试用例主程序

report：报告，生成存放报告目录

test_case：测试用例脚本所在位置路径，

pytest框架的原理就是在run_case目录下，使用pytest去搜索并且执行以test开头的测试用例脚本，执行完成生成后，会生成相应的测试报告到report目录下。

八、UI自动化测试

8.1 元素定位（八种）

如何选择合适的元素定位方法？

- ①在进行元素定位时，如果看到有唯一的id/name/class值，那么就直接定位
- ②如果id/name/class值不唯一时，那么我会选择css选择器/xpath路径定位
- ③如果是超链接(a标签)，那么直接使用link_text/partial_link_text定位
- ④还有一种是不怎么使用的tag_name定位，这个是需要通过标签的索引进行定位，而索引一般不好计算。不常用。

8.2 时间等待

共同点：超过设置的等待时间，如果没有等待需要的元素，都会报错。

不同点：

强制等待：需要导包，必须强制等待设置时间；

隐性等待：不需要导包，会等待整个页面都加载渲染完成，就立即执行；

显示等待：需要导包，代码量相对较多，只为了等待需要的元素出现，就立即执行。

8.3 页面滑动/滚动

只需要死记一种方法即可：快速跳转滚动

先使用八大元素定位方法，进行确定目标元素(比如：提交审核按钮)，再直接快速跳转滚动：

```
driver.find_element(By.ID,"publishBnt").location_once_scrolled_into_view
```

8.4 文件上传

之前我们开发进行文件上传所使用的元素标签是input标签

我们进行文件上传的时候，使用元素定位后，通过send_keys方法，把文件的路径&文件名.拓展名进行上传。

8.5 下拉框操作

下拉框都是select 标签，那么需要导入并使用select 包来进行操作

先通过八大元素定位到下拉框后

再使用select 包，通过索引/文本/value值 来进行定位操作

8.6 隐藏元素的操作

面试：在之前有没遇到过隐藏的元素？在元素定位的时候，有无遇到什么难题/困难？

（揣摩面试官的心理：并不是需要你说一些他没有遇到的困难或难题，只是想考察下你的思维逻辑表达能力、临场应变能力）

隐藏元素操作：需要跟前端开发确认到存在的隐藏元素，通过八大元素定位到下拉框后

再使用JS脚本（execute_script()）进行修改/删除

8.7 内嵌页面

通过页面元素属性，去查找是否存在frame/iframe，来判断是否存在内嵌页面

如果存在，那么通过switch_to.frame('name/id值') 来进入内嵌页面，再进行定位操作

如果需要退出内嵌页面，可以通过switch_to.parent_frame() 退出当前内嵌页面

（switch_to.default_content() 退出到最外层页面）

8.8 窗口操作（警示窗/二次确认，弹窗页面切换）

通过switch_to.alert切换到警示窗/二次确认，再通过dismiss() 来取消，通过accept()来确认

弹窗：先通过driver.current_window_handle 获取所有的句柄，在通过driver.switch_to.window (句柄)的索引来进行切换

8.9 断言

使用的是assert 断言，判断实际结果跟预期结果是否一致，一致则执行成功，不一致则会断言失败；

方法一：通过元素定位后.text()方法，获取到实际的文本内容来断言

方法二：通过driver.title 获取页面的标题来进行断言

方法三：通过元素定位后.get_attribute('元素')方法，获取到元素属性来进行断言

8.11 python+selenium+pytest框架

UI自动化测试我这边只做了一年左右（不要说太多）：

使用的是python+selenium+pytest框架来完成接口自动化测试，框架是公司早就搭建好了的，我这边主要是负责后面批次迭代，涉及到的功能用例脚本的编写新增/修改，其实我自己也能在个人电脑参照着也能搭建起这个框架

（UI 自动化测试，执行的就是功能测试/功能测试用例，就是点点点）

原来我们自动化测试主要是用的python+selenium+pytest框架做的，

主要对自动化进行规划，把数据、代码、资源、公共方法进行分离，方便后续维护；

UI自动化测试其实主要通过几个方面：元素定位，参数化，断言，模块的封装，数据闭环等来实现的，

其中元素定位：主要是css定位，也会用一些xpath, id, name, classname, link_text等等定位

参数化主要是用列表、字典、以及导入exell表格库（xlrd）对数据进行参数化；

断言：主要是用assert断言来，判断实际结果是否与预期结果相符；

对于封装，我主要会对界面上的元素，一些公共方法（数据库操作，读取excel数据公共方法等），还有测试数据（变量，字典，列表excel表等）和一些元素操作，测试用例等等进行封装；

最后闭环的话，就是都要把新增的数据删除，打开的界面都要关闭。

（介绍自己复杂技能

1、自己已经掌握，可以不按老师讲的废话来说，自己总结出来要点

2、不太明白，看之前上课的笔记、视频，看懂后再总结

3、自己看笔记/视频还不懂，只能找老师，找同学问下，看懂再总结

4、怎么都搞不定，只能死记硬背

5、要不还是放弃吧~

)

8.12 自动化测试现状

小公司：不做自动化测试（包括接口自动化、web--UI自动化、APP--UI自动化）

大公司：10% 公司会做自动化测试

10% 里面，有8%是只做接口自动化测试，只有2% 会涉及到UI自动化

（现状：面试会问，但实际工作是没有。

面试官问你自动化的技能知识，学生回答完后，可以反问面试官：不好意思，我想了解一下，咱公司这边，自动化测试是做到了什么程度？有多少人专门负责？涉及了多少个项目？

8.13 UI自动化测试脚本写了多少用例，运行时间是多久？

因为UI自动化测试，刚开始做，只做了1年左右，**UI自动化覆盖率30%~40%**，2年+的覆盖率在70%

只讲核心功能流程的正常场景用例，正向用例转换成UI自动化测试脚本，

UI自动化用例执行，1条用例脚本执行时间，3~9s，平均5s 一条

功能——功能用例（项目≠批次迭代）——运行时间

1个功能——平均 5条用例

项目功能 项目用例数 用例x每个用例执行时间x30%

小型项目：200+ 1000 1000x5x30%=1500s 约25分钟，可以说执行不到半小时

中型项目：320+ 1600 1600x5x30%=2400s 约40分钟，可以说不到一个小时

大型项目：600+ 3000 3000x5x30%=4500s 约75分钟，可以说一个半小时左右

（涉及到数据的时候，不要说精确的数据，说大概就好了，

比如说，功能数在两三百，具体也记不太清楚了；

用例数在六七百，没怎么统计过，项目中有3个测试，我自己写得比较多，有两三百，其他人会相对少点。）

九、APP测试

9.1 APP怎么测？（专项测试）

APP还是以功能测试为主，跟web项目的不同点在于专项测试会考虑更多：

比如安装卸载测试、弱网测试、权限测试、交互性测试、兼容性测试、稳定性测试、APP的性能测试。

我主要负责XXXX(三/四个)专项测试(下面再针对自己负责的三个四专项测试张开来讲)

安装卸载测试：主要考虑首次与覆盖安装升级能否成功，卸载时候能否把数据同步清除干净；

弱网测试：需要模拟APP在网络差的情况下，功能能都正常使用，不会出现报错、崩溃的重大错误；

权限测试：考虑的是APP跟手机的软硬件之间的权限，有权限的情况下可以正常使用，无权限是需要提示用户相应的开启方式，比如（可以稍微结合自己项目来讲）定位、存储权限、麦克风、相机；

兼容性测试：公司这边有专门测试的手机，先保证初步的兼容性测试可以通过；后续会找第三方做安卓top300，iOS top50的兼容性测试，并提供兼容性测试报告。

稳定性测试：是使用adb工具中的monkey命令进行测试，一般会通宵测试，在晚上大家都下班之后，使用基本的命令adb shell monkey -p 包名 -v -v -v 次数。为了使monkey更贴近使用习惯，我们会加上思考时间，让点击的间隔500毫秒，还会根据项目需求设置指定的事件比例，最后面加上忽略报错。一般设置的执行次数在：7万次左右。如果出现BUG，让开发解决，解决后我们再通过上种子数去 回归验证BUG是否解决。

思考时间与执行次数的计算：如果是500ms点击一次，那么1秒点击2次，21:00 ~06:00 需要点击多少次？

性能测试：是使用腾讯的性能狗工具进行测试，一般测试在APP使用的过程中，对手机帧率、CPU、GPU、运行内容、流量、电量是使用消耗情况，对比判断APP的性能是否达标。

（并不是所有的批次迭代都需要做APP专项测试，稳定性测试、性能测试，只有大版本迭代时才做；

如果没有主动提及稳定性测试、APP性能测试，面试官有提问到，那么可以说是其他同事/老大/组长做的。）

9.2 APP怎么做接口测试？

通过接口文档--去做接口测试

(切记：抓包跟接口测试没有半毛钱关系)

会出现一个项目：同时Web端、APP端、（PC端）、小程序。举例：唯品会、京东

哪怕也有不同端，同一个功能，接口文档都是同一份。只是在接口(文档)中某个参数代表了不同端。

比如：可能使用ajax参数中的不同值，代表不同端，0是APP端操作，1是web端的操作，2是小程序的操作。

9.3 APP测试跟web项目测试有什么不同？

web的**兼容性**是考虑不同浏览器之间的兼容，APP项目的兼容，考虑的是不同手机品牌，不同手机操作系统，不同屏幕大小分辨率之间的兼容

APP除了需要做接口的**性能测试**，APP本身也有性能测试，通过性能狗进行测试，通过性能狗判断手机帧率、CPU、GPU、运行内容、流量、电量是使用消耗情况，对比判断APP的性能是否达标。

其他的专项测试，是APP特有的，而web项目是不需要的，比如有安装卸载测试、权限测试、交互性测试、稳定性测试。

9.4 APP跟小程序测试有什么不同？（小程序如何测试？）

小程序跟APP很类似，都是以功能测试为主。

但APP是以手机操作系统为载体，而小程序需要依赖载体APP，比如微信、支付宝、百度等.....

小程序除了功能界面的测试，也是需要进行交互性测试、兼容性测试、权限测试。

比如：之前有测试过一个微信小程序：需要将小程序上传到微信小程序开发者平台。通过微信的扫描进行安装到微信中进行使用，测试还是主要以按照需求原型，以功能界面测试为主，不同的是：

交互性测试：小程序主要考虑的是与载体APP微信之间的交互，比如卡包、微信支付；

权限测试：小程序是继承载体APP微信的权限，微信拥有的权限，小程序才可以使用，微信没有的权限，小程序是不能使用的，需要提示怎么运行载体微信APP获得权限。

兼容性测试：小程序的兼容性测试，需要测试的是与不同载体APP版本的兼容，比如最近微信版本方面的兼容。

十、（接口）性能测试

10.1 性能测试概述

(如果没有特别强调APP方面的性能测试，默认都是接口的性能测试)

1、我这边没有负责过大型项目的整体性能测试，只是做过批次迭代的接口性能测试；（打预防针）

2、做性能测试，需要确定哪些接口是有必要做性能测试的；

3、需要先确定接口的性能指标：

①从业务产品得到明确的性能指标；

②如果没有的话，只能从历史活动数据中获取作为参考；

③如果是新项目系统，只能使用二八原则估算一下作为参考。

(80%用户会集中在20%时间内进行使用，可以计算到峰值TPS)

4、我们是使用jmeter工具进行性能测试的，最主要的是在线程组设置**并发用户数**（之前批次迭代并发用户在300~800），添加聚合报告来监控查看性能指标

5、此外，我们还需要对**服务器进行监控**，是使用easynmon工具，监控服务器的CPU（低于80%）、(运行)内存、网络宽带、磁盘读 写。

6、在性能测试的过程中，需要关注**聚合报告上的接口性能指标**：平均响应时间（小几十毫秒~两三百毫秒），top响应时间，TPS/QPS（大几百~两三千），错误率(大多数不能超1%就行，但涉及到金额是接口是不允许有错误)

7、通过观察聚合报告的接口指标以及服务器的指标，来判断接口性能是否达标，是否测试通过。

8、之前性能测试，大概流程就是这样操作。

(以上核心都能表述出来，性能测试都算过关，可以达到入门的水平)

10.2 性能不达标，可能是什么原因？

(你怎么去确认为什么不达标？你怎么解决？)

(期望招一个性能测试大佬)

1、之前也试过性能指标不达标，那么都是由开发自行去定位解决的

2、我这边只是负责进行性能测试，把测试结果告诉/同步开发

3、通过之前的性能测试，我这边也大概熟悉知道开发是从哪些方面去进行定位排查性能不达标原因

那么可以从以下几个方面去定位排查：



①Windows电脑以及jmeter：电脑的配置是否OK，jmeter压测脚本是否合理，有无使用随机参数化等增加负担，要使用非GIU界面进行压测

②网络是否OK，使用千兆光纤，可以排查下网速是否收到影响

③中间件的连接数是否合理，要及时修改

④服务代码质量是否OK，开发写的代码太垃圾导致性能不达标

⑤数据库有无使用索引，SQL语句是否最精简，SQL能从一个表查到数据，就不要使用多表查询（到此，性能测试以及达到良好状态，再往下也说不上内容）

十一、项目挑选以及介绍
