接口测试

2019年11月11日

0:54

什么是接口测试？

接口测试是测试系统组件间接口的一种测试。

接口测试的重点是检查数据的交换，传递的正确性，以及接口间逻辑依赖关系。

提交接口测试的重要意义：实现开发期并行测试，减少页面层测试的深度，缩短整个项目的测试周期。

我们测试的接口属于哪一类？

服务器接口（基于HTTP协议的接口）

大多数人常说的接口测试，通常是 B/S架构，由客户端（浏览器）调用，或模拟客户端（浏览器）调用服务器提供的请求接口，由服务器完成处理并返回一个应答的过程。

例如：Webservice接口，http接口，jms接口，hessian接口。

HTTP接口传递数据最常用的方式：

Get方式是从服务器上获取数据；在做数据查询时，建议用Get方式；如：商品信息接口、搜索接口、博客访客接口等。

Post方式是向服务器传送数据 ；在做数据添加、修改或删除时，建议用Post方式 ；如：微博图片上传图片接口、登录注册接口等。

接口请求流程

请见图

接口请求参数分析：

\*请求参数必填项、可选项

\*输入合法、非法参数

\*请求参数边界值

接口响应结果：

\*对异常请求的处理

\*请求头的处理

\*存在业务关联

\*涉及数据库交互的

\*结构

\*节点值有效性

\*列表结构排序

HTTP接口测试的类型

1.GET请求无参数

2.GET请求有参数

3.POST请求

接口测试用例编写要点：

\* 测试每个参数类型不合法的情况(类型不合法容易遗漏NULL型)

\* 测试每个参数取值范围不合法的情况

\* 测试参数为空的情况

\* 测试参数前后台定义的一致性

\* 测试每个参数的上下限(这里容易出致命的BUG，如果程序处理不当，可能导致崩溃)

\* 如果两个请求有严格的先后顺序，需要测试调转顺序的情况

接口测试的步骤：

1）请求接口

2）获取返回值

3）断言：判断实际结果是否与预期相同

（测试的时候不可以开代理抓包软件）

接口测试实例讲解（PostMan的使用）

postman使用

（chrome中安装jsonview）

(chrome中安装postman)

一个借口请求需要设置：请求URL，请求方法，请求头，请求参数

说明：GET请求的参数在URL链接上，所以，GET请求的请求头与请求参数如在接口文档中无特别声明时，可以不填。

1.填写请求、参数

2.Tests中添加断言

3.Save as保存

4.创建collection（聚集、集合）

5.左侧点击collection-RUN

6.点击start test 右侧可以看到运行结果

7.命令行运行集合（先保存，后安装newman）

1）GET无参数：

只需验证状态码是否为200

或者返回值中的某个特定字段的校验errno: 0

以众筹产品详细列表为列

<http://api.zhongchou.cn/deal/list?v=1>

在chrome中查看返回的样例

先在postman中教如何进行接口测试（界面化工具的操作）

再用Python实现代码进行接口测试（类和对象的讲解，安装request包）

2）GET有参数：

验证返回的状态码为200

验证默认值count==20

验证默认值start==0

验证书的数量==20

验证books中包含关键字“小王子”

以豆瓣开发者api为例

搜索图书：

接口地址：GET <http://api.douban.com/v2/book/search>

参数 意义 备注

q 查询关键字 q和tag必填其一

tag 查询的tag q和tag必填其一

start 取结果的offset 默认为0

count 取结果的条数 默认为20，最大为100

脚本：

//验证返回数据中包含关键字

tests["验证response中包含 小王子"] = responseBody.has("小王子");

//验证返回的状态码是否为200

tests["Status code is 200"] = responseCode.code === 200;

//验证start=0&count=20

//验证start=0

var jsonData = JSON.parse(responseBody);

var b\_count = jsonData.count; //定义变量b\_count

tests["start"] = jsonData.start === 20;

//验证count=20

tests["count"] = jsonData.count === 20;

//验证返回的图书列表长度为默认值

tests["图书列表长度为默认值20"] = jsonData.count === 20;

tests["图书列表长度为默认值为b\_count"] = jsonData.count === b\_count;

GET的参数是个字符串，POST的参数要封装到HTTP请求头里，是字典

3）POST请求

验证

以登录为例

<http://api.zhongchou.cn/user/login?v=1'>

参数：

'identity':'18311338905','password':'a000000'

requests包使用说明中文文档：

<http://docs.python-requests.org/zh_CN/latest/user/quickstart.html>

<http://blog.csdn.net/shanzhizi/article/details/50903748>

面试题2：平常你是怎么测试接口的？

· 通过性验证：首先肯定要保证这个接口功能是好使的，也就是正常的通过性测试，按照接口文档上的参数，正常传入，是否可以返回正确的结果。

·

参数组合：现在有一个操作商品的接口，有个字段type，传1的时候代表修改商品，商品id、商品名称、价格有一个是必传的，type传2的时候是删除商品，

商品id是必传的，这样的，就要测参数组合了，type传1的时候，只传商品名称能不能修改成功，id、名称、价格都传的时候能不能修改成功。

·

·

接口安全：

1、绕过验证，比如说购买了一个商品，它的价格是300元，那我在提交订单时候，我把这个商品的价格改成3元，后端有没有做验证，更狠点，我把钱改成-3，是不是我的余额还要增加？

2、绕过身份授权，比如说修改商品信息接口，那必须得是卖家才能修改，那我传一个普通用户，能不能修改成功，我传一个其他的卖家能不能修改成功

3、参数是否加密，比如说我登陆的接口，用户名和密码是不是加密，如果不加密的话，别人拦截到你的请求，就能获取到你的信息了，加密规则是否容易破解。

4、密码安全规则，密码的复杂程度校验

·

·

异常验证：

所谓异常验证，也就是我不按照你接口文档上的要求输入参数，来验证接口对异常情况的校验。比如说必填的参数不填，输入整数类型的，传入字符串类型，长度是10的，传11，总之就是你说怎么来，我就不怎么来，其实也就这三种，必传非必传、参数类型、入参长度。

·

·

性能测试

接口并发情况，如上面提到的：一个账号，同时（大于2个请求）对最后一个商品下单，或不同账号，对最后一个商品下单

接口响应时间，响应时间太长了，肯定需要优化，一般都是毫秒级别

·

面试题5：没有接口文档，如果做接口测试？（这是个送命题）

没有接口文档，那还能咋办，瞎测呗！一个公司的开发流程里面，如果接口文档都没有，是无法展开接口测试的，你都不知道这个接口干什么的，也不知道具体每个字段代表什么意思，那还测啥呢？

--当然，你肯定不能回答面试官不测(心理mmp，脸上笑嘻嘻)，接下来就是扯犊子时间

1.没有接口文档，那就需要先跟开发沟通，然后整理接口文档（本来是开发写的，没办法，为了唬住面试官，先说自己整理了）

2.没有接口文档，可以抓包看接口请求参数，然后不懂的跟开发沟通

本题主要靠情商，通俗来说就是忽悠能力，先唬住面试官了再说，进去了也是瞎测测，随时做好背锅的准备

面试题6：在手工接口测试或者自动化接口测试的过程中，上下游接口有数据依赖如何处理？

用一个全局变量来处理依赖的数据，比如登录后返回token,其它接口都需要这个token,那就用全局变量来传token参数

面试题7：依赖于第三方数据的接口如何进行测试？

这个标准答案是：mock

接着面试官会问你，如果mock的，然后你就顺着坑继续挖，搭建mock服务，

面试题9：如何模拟弱网测试

fiddler和charles都可以模拟弱网测试，平常说的模拟丢包，也是模拟弱网测试

面试题10：如何分析一个bug是前端还是后端的？

平常提bug的时候，前端开发和后端开发总是扯皮，不承认是对方的bug

这种情况很容易判断，先抓包看请求报文，对着接口文档，看请求报文有没问题，有问题就是前端发的数据不对

请求报文没问题，那就看返回报文，返回的数据不对，那就是后端开发的问题咯