接口自动化测试脚本技术文档

笔记本: 测试

创建时间: 2021/8/9 10:36 **更新时间:** 2021/8/9 11:53

作者: 540726844@qq.com

URL: https://blog.csdn.net/rhx_qiuzhi/article/details/80304864

接口自动化测试脚本技术文档

概述

此脚本采用python+requests+excel+unittest+ddt接口自动化数据驱动并生成html报告

环境准备:

- python3.6
- requests
- xlrd
- HTMLTestRunner

项目结构

```
图 文件(P 编辑(E) 视图(Y) 导航(N) 代码(C) 重构(R) 运行(U) 工具(T) Git(G) 窗口(W) 帮助(H) API_Test - main.py - Administrator
□ 项目 ▼
□ API_Te
                    ✓ ■ API_Test D:\haily\wf\API_Test
                                          # -*- coding:utf-8 -*-
                                 用例存放跨路及挑門用例时生成的excel结果存放 import os
    ∨ 🖿 Case

∨ III excel

✓
          # 2021-08-02-16_22_35-request_case 4 import time
         2021-08-02-17_02_3-1-equest_case
2021-08-02-17_06_45-request_case
2021-08-02-17_36_57-request_case
2021-08-02-17_36_57-request_case
2021-08-02-17_55_45-request_case
                                          import HTMLTestRunner

→ Pull Request

          2021-08-02-18_01_04-request_case 8 case_path = os.path.join(os.getcwd())
          2021-08-04-15_17_33-request_case 9
          2021-08-09-10_41_55-request_case 10 report_path = os.path.join(os.getcwd(), 'report')
           # 2021-08-09-10_43_02-request_case_11
                                          #print report_path
           # 2021-08-09-10_59_04-request_case 12
          海疆接口测试.xls
海疆接口测试.xlsx
                                    udef all_case():
测试用例表格系的。解tion表标系estLoader.discover(case_path, pattern="test_*.py", top_level_dir=Nc
           ₩ 海疆接口测试1.xls
                                    15
           ₩ 海疆接口测试2.xls
           海疆接口测试ding.xls
                                               return discover
           🎝 海疆接口测试ding.xlsx
           🐇 海疆接口测试xiugai.xls
           ∰海疆接□测试zhag.xls
                                    19 > -if __name__ == '__main__':
                                    20 # 1. 我服当你时间,这样便于下面的使用。
用例执行判断模块e.strftime("%Y-%m-%d-%H_%M_%S", time.localtime(time.time()))

∨ Imaily_cloud

           💑 __init__.py
           # test_001.py ⁴
                                    22
           toRequests.py 👍
                                     🚜 _init_.py
       //sconf.py
                                    25

∨ □ config

                                    26
                                               # 3、打开一个文件,将result写入此file中
         __init_.py
                                     logs 🗎
                                                                                  title=u'海疆接口自动化测试报告 测试结果如下:'.
        # result_2021-07-15-14_39
                                                                                  description=u'用例执行情况:')
        # result_2021-08-09-10_58_45.html
                                    31 生成HTML测试报告存放路径
      amain.py 👍
                                               runner.run(all_case())
  > IIII 外部库
                              执行主函数 fp.close()
    7 草稿文件和控制台
11 结构
                                           if __name__ == '__main__'
  I<sup>‡</sup> Git ▶ 运行 ≔ TODO ● 问题 図 终端 参 Python Packages ● Python 控制台
```

一些修改

在安装ddt时,调用ddt时发现生成的测试报告没有显示用例名称,故在ddt内做了一些修 改。

```
👸 main.py × 👸 test_001.py × 👸 ddt.py × 👸 inspect.py × 🎉 readexcel.py × 👸 conf.py × 👸 toRequests.py ×
138
            return wrapper
139
140
141
        def mk_test_name(name, value, index=0, name_fmt=TestNameFormat.DEFAULT):
            Generate a new name for a test case.
144
           It will take the original test name and append an ordinal index and a
146
           string representation of the value, and convert the result into a valid
           python identifier by replacing extraneous characters with ``_``.
148
           We avoid doing str(value) if dealing with non-trivial values.
            The problem is possible different names with different runs, e.g.
150
           different order of dictionary keys (see PYTHONHASHSEED) or dealing
           with mock objects.
           Trivial scalar values are passed as is.
           A "trivial" value is a plain scalar, or a tuple or list consisting
           only of trivial values.
           The test name format is controlled by enum ``TestNameFormat`` as well. See
158
           the enum documentation for further details.
160
           # Add zeros before index to keep order
162
            index = "{0:0{1}}".format(index + 1, index_len)
            #if name_fmt is TestNameFormat.INDEX_ONLY or not is_trivial(value):
164
               #return "{0}_{1}".format(name, index)
           if not is_trivial(value) and type(value) is not dict:
166
                                                                                                         修改的内容
               return "{0}_{1}".format(name, index)
167
           if type(value) is dict:
168
169
               try:
                  value = value[1] #value[1]作为value值
               except:
                   return "{0}_{1}".format(name, index)
               value = str(value)
            except UnicodeEncodeError:
175
```

测试用例格式

A	В	c	D	E	F	G	H	I	J	K	L
F列.	名称	请求地址	请求产	请求	请求参数	参数类型	預期响应关键字	用例级别	状态码 🔻	success	测试结果
	正常登录	/login/login	post	0	{"account": "admin", "password": "111111"}	json	登陆成功	K	200	True	
	登录密码为空	/login/login	post	0	("account": "admin", "password": "")	json	登陆失败	K	200	True	
	账户名为空	/login/login	post	()	{"account":"", "password":"1111111"}	json	登录失败		500	False	
ŀ	账户名与密码均为空	/login/login	post	0	{"account":"", "password":""}	json	登录失败	K	500	False	
	输入不存在的账户	/login/login	post	0	["account": "a", "password": "123456"}	json	登录失败	K	500	False	
	输入错误密码	/login/login	post	0	{"account": "aamin", "password": "123456"}	json	登录失败	K	400	False	
	成功保存 公司配置的应用	/company/saveCompApp	post	0	{"companyId":2 , "appID":[9]}	json	保存成功	K	200	True	
	必填项公司id为空	/company/saveCompApp	post	0	{"appID":[9]}	json	保存失败	K	400	False	
	必填项appid为空	/company/saveCompApp	post	0	["companyId":2]	json	保存失败	K	500	False	
0	缺少必填项公司id	/company/saveCompApp	post	0	"appID":[9]	json	保存成功	K	200	True	
1	缺少必填项appid	/company/saveCompApp	post	{}	(companyid :1418417199091008800	json	保存失败	K	500	False	
2	获取存在的 公司id应用列表	/company/getAppList	post	0	("companyId":1415196380457918466	json	成功	K	200	True	
3	获取不存在的 公司id应用列表	/company/getAppList	post	()	{"companyId":1418412733637668865		失败	K	500	False	
4	部门状态 由启用转禁用	/deptment/deptmentStatus	post	0	["deptmentId":141520785267627622 6, "status":2]	params	成功	K	200	True	
5	部门状态 由禁用转启用	/deptment/deptmentStatus	post	0	["deptmentId":141520785267627622 6, "status":1]	json	成功	K	200	True	
6	用户信息 获取当前登录用户信息	/account/getLoginUserInfo	post	{}	None	data	成功	K	200	True	

xlrd读excel数据

1.先从excel里面读取测试数据,返回字典格式

```
∨ Case
                                                       url = data_value.index('请求地址')
                                                       data = data_value.index('请求参数')
       2021-08-02-16_22_35-request_case 34
       2021-08-02-16_22_35-request_case 34

2021-08-02-16_28_43-request_case 35

2021-08-02-17_06_45-request_case 36

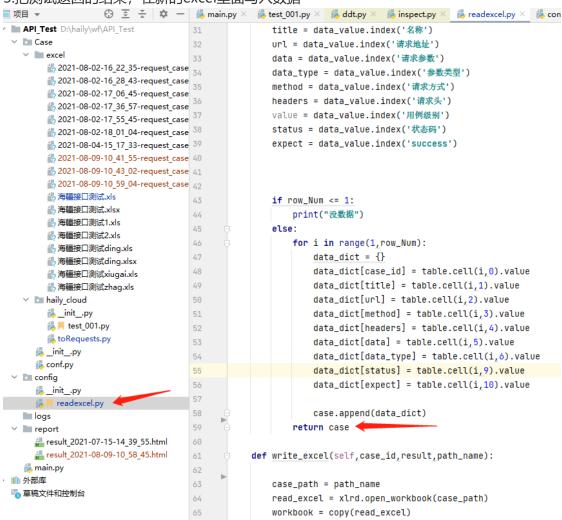
2021-08-02-17_55_45-request_case 37

2021-08-02-18_01_04-request_case 38
                                                       data type = data value.index('參數类型')
                                                       method = data_value.index('请求方式')
                                                       headers = data_value.index('请求头')
                                                       value = data_value.index('用例級别')
                                                       status = data_value.index('状态码')
       $\mathbb{6}\ 2021-08-04-15_17_33-request_case \quad 39
                                                       expect = data_value.index('success')
       40 2021-08-09-10_41_55-request_case
       $2021-08-09-10 43 02-request case 41
       $\tilde{4}$ 2021-08-09-10_59_04-request_case 42
       級 海福接□测试.xls
                                                       if row_Num <= 1:
       海疆接口测试.xlsx
                                                           print("没数据")
       //。海疆接口测试1.xls
                                                       else:
       場 海疆接□测试2.xls
                                                            for i in range(1,row_Num):
        為強力
                                                                data_dict = {}
       🐇 海疆接口测试ding.xlsx
       ≝ 海疆接□测试xiugai.xls
                                                                {\tt data\_dict[case\_id] = table.cell(i_{\ell}0).value}
       // 海疆接口测试zhag.xls
                                                                data_dict[title] = table.cell(i,1).value
  haily_cloud
                                                                data dict[url] = table.cell(i.2).value
                                      51
       init_.py
                                                                data_dict[method] = table.cell(i,3).value
       52
                                                                data_dict[headers] = table.cell(i,4).value
       toRequests.py
                                                                data_dict[data] = table.cell(i,5).value
     🏂 _init_.py
                                                                data_dict[data_type] = table.cell(i,6).value
     🖧 conf.py
                                                                data_dict[status] = table.cell(i,9).value
✓ □ config
                                                                data_dict[expect] = table.cell(i,10).value
    # reade
                                                                case.append(data dict)
   logs 🖿
                                                            return case ∢
∨ I report
    # result_2021-07-15-14_39_55.html
     aresult_2021-08-09-10_58_45.html
                                                   def write_excel(self,case_id,result,path_name):
   // main.py
|||| 外部库
                                                       case_path = path_name
7 草稿文件和控制台
                                                       read_excel = xlrd.open_workbook(case_path)
                                      65
                                                       workbook = copy(read_excel)
                                                       #workbook = read_excel
                                                       #sheet = read_excel.sheet_by_name("Sheet1")
                                                       sheet = workbook.get_sheet(0) #获取第1个表的数据
                                                       sheet.write(case_id,11,result)
```

封装request请求方法

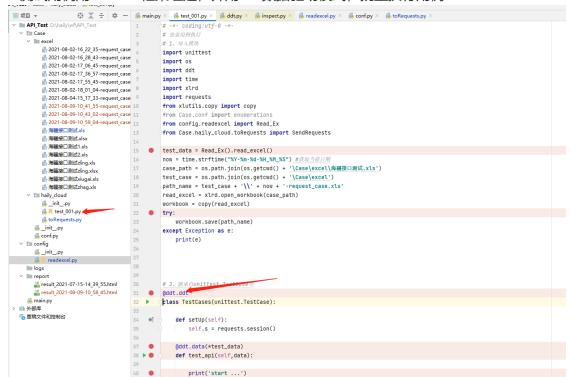
- 1.把从excel读处理的数据作为请求参数,封装requests请求方法,传入请求参数,并返回结果
- 2.为了不污染测试的数据,出报告的时候先将测试的excel复制都应该新的excel

3.把测试返回的结果,在新的excel里面写入数据



测试用例unittest+ddt

1.测试用例用unittest框架组建,并用ddt数据驱动模式,批量执行用例



生成测试报告

海疆接口自动化测试报告,测试结果如下:

Start Time: 2021-08-09 10:43:02 Duration: 0:00:05.926997 Status: Pass 125 Failure 25

用例执行情况:

Show <u>Summary Failed All</u>

t Group/Test case	Count	Pass	Fail	Error	View				
e.haily_cloud.test_001.TestCases	150	125	25	0	Detail				
test api 001 正常登录	130	122	pass		Detail				
test api 002 背景密码为空	fail								
test api 003 账户名为空	pass								
test api 004 聚户名与密码均为空		pass							
test_api_005_输入不存在的账户			pass						
test api 006 输入错误密码			pass						
test api 007 成功保存 公司配置的应用			pass						
test api 008 彩填喷公司d为空			fail						
test api 009 必填项appid为空			pass						
test api 010 缺少必编项公司id			pass						
test api 011 缺少必缴项appid			pass						
test api 012 获取存在的 公司id应用列表			fail						
test api 013 获取不存在的 公司id应用列表			pass						
test api 014 部门状态 由启用转禁用			fail						
test_api_015_额门状态_由禁用转启用			fail						
test_api_016_用户信息_获取当前登录用户信息			fail						
test_api_017_用户信息_新增用户			pass						
test_api_018_用户信息_新增用户_用户名account重复			fail						
test_api_019_用户信息_新增用户_id重复			fail						
test_api_020_用户信息_修改用户信息			fail						
test_api_021_用户信息_删除人员相关信息			pass						
test_api_022_用户信息_设置人员又启动转票用			pass						
test_api_023_用户信息_设置人员又启动转禁用			pass						
test_api_024_用户信息_分页查询用户信息			pass						
test_api_025_用户信息_获取人员详情			fail						
test_api_026_人员信息_保存人员配置好的角色			pass						
test_api_027_人员信息,获取的人员未配置角色列表			pass						
test_api_028_人员信息_获取的人员已配置角色列表			pass						
test api 029 用户信息 给不存在的人员id 分配权限			fail						