# Autotestplat-V5.0自动化测试平台使用手册

## 前言：

1. Autotestplat是一站式自动化测试平台系统，目前集成了接口，性能、AppUI自动化测试。
2. 目前实现了接口自动化完整功能，产品，用户设置，系统参数设置、接口用例、参数化，测试计划，定时设置，测试报告。实现了性能测试，AppUI测试初步功能。
3. 无需修改任何代码或脚本，支持团队，多项目，测试报告详情及保存，直观方便可视化，充分达到了企业级自动化测试平台需求。

## 一、环境安装

第一步：安装pycharm，安装包版本<https://github.com/testdevhome/installPackage>

第二步：安装Python3.6

第三步：安装mysql

第四步：下载Autotestplat源码包 <https://github.com/testdevhome/Autotestplat>

第五步：安装pip install -r requirements.txt

setuptools==28.8.0 -i https://pypi.tuna.tsinghua.edu.cn/simple

kombu==3.0.37 -i https://pypi.tuna.tsinghua.edu.cn/simple

celery==3.1.26.post2 -i https://pypi.tuna.tsinghua.edu.cn/simple

django==2.1.3 -i https://pypi.tuna.tsinghua.edu.cn/simple

django-celery==3.2.2 -i https://pypi.tuna.tsinghua.edu.cn/simple

redis==2.10.6 -i https://pypi.tuna.tsinghua.edu.cn/simple

celery-with-redis==3.0 -i https://pypi.tuna.tsinghua.edu.cn/simple

django-redis==4.0.0 -i https://pypi.tuna.tsinghua.edu.cn/simple

PyMySQL==1.0.2 -i https://pypi.tuna.tsinghua.edu.cn/simple

requests==2.15.1 -i https://pypi.tuna.tsinghua.edu.cn/simple

urllib3==1.22 -i https://pypi.tuna.tsinghua.edu.cn/simple

chardet==3.0.4 -i https://pypi.tuna.tsinghua.edu.cn/simple

certifi==2023.7.22 -i https://pypi.tuna.tsinghua.edu.cn/simple

idna==2.6 -i https://pypi.tuna.tsinghua.edu.cn/simple

adbutils==1.2.11 -i https://pypi.tuna.tsinghua.edu.cn/simple

django-cors-headers==3.0.0 -i https://pypi.tuna.tsinghua.edu.cn/simple

mysqlclient==1.4.6 -i https://pypi.tuna.tsinghua.edu.cn/simple

paramiko==3.3.1 -i https://pypi.tuna.tsinghua.edu.cn/simple

psutil==5.9.5 -i https://pypi.tuna.tsinghua.edu.cn/simple

pywin32==305 -i <https://pypi.tuna.tsinghua.edu.cn/simple>

faker==14.2.1 -i https://pypi.tuna.tsinghua.edu.cn/simple

uiautomator2==2.16.25 -i https://pypi.tuna.tsinghua.edu.cn/simple

weditor==0.6.4 -i <https://pypi.tuna.tsinghua.edu.cn/simple>

selenium==3.141.0 -i https://pypi.tuna.tsinghua.edu.cn/simple

第六步：创建数据库名autotestplat，导入数据autotestplat.sql

第七步：python manage.py runserver 0.0.0.0:80  
第八步：redis-server redis.windows.conf  
第九步：python manage.py celery worker -c 6 -l debug  
第十步：python manage.py celery beat

## 二、简要操作步聚

接口测试

第一步、设置用户，设置产品/项目，用户设置默认产品，也就是当前正在进入使用的产品。  
第二步、在系统设置中设置参数，环境设置即设置host地址，常量即设置固定值作为参数，变量可以是时间戳，随机数字、字符等。  
第三步、在接口用例中进行接口用例录入调试，设置断言，通过正则表达式设置关联参数，执行；在接口中通过{关键字}设置参数。  
第四步、在测试计划中选择接口用例，定时设置，勾选调试，更新，增行，减行，更改顺序。  
第五步、在测试报告中查看测试记录，报告的详情数据。

 性能测试

第一步、选择接口，输入并发数，持续压测时间，点击生成jmeter脚本。  
第二步、点击测试开始。  
第三步、点击查看结果。

AppUI测试

第一步、系统设置App设置。  
第二步、启动weditor，连接手机，抓取App元素控件，录入用例。  
第三步、执行，查看用例结果。

## 三、详细操作步聚

## 产品管理



▲图 1.1

## 用户设置

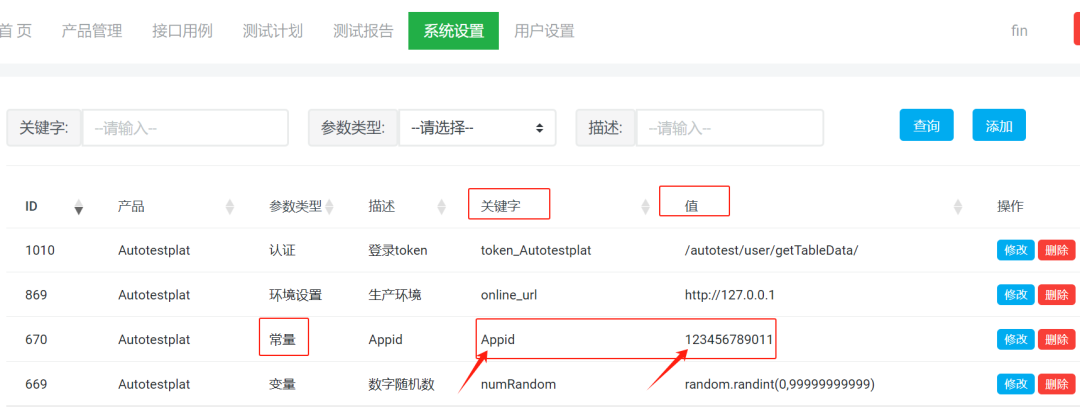
用户设置默认产品，也就是当前正在进入使用的产品，那么以下其他模块（3-6步聚）都默认为当前产品的操作和数据显示。



▲图 2.1

## 系统设置

3.1、常量参数化，例如在引用一些公共相同的数据时，采用常量的方式定义，参数里面对某个固定商品ID进行查看，编辑，添加属性，订购等，就可以给它定义为常量。当要对另一个商品ID进行类似操作时，就改一下常量里面的值即可，因而不需要逐个修改每个ID，只需改一次这个常量的参数值，因为在引用时统一使用的是定义的常量。



▲图 3.1

3.2、变量参数化，例如当新增幂等数据时，自动化新增一个产品名为A的数据，但这个产品A不能重复新增，重复的话会新增失败，这时必须使用变量随机生成不同值的方式避免产品名为A重复，执行一次的时候新增产品A，再次执行时这个变量随机可能生成了C，这时就可以新增成功



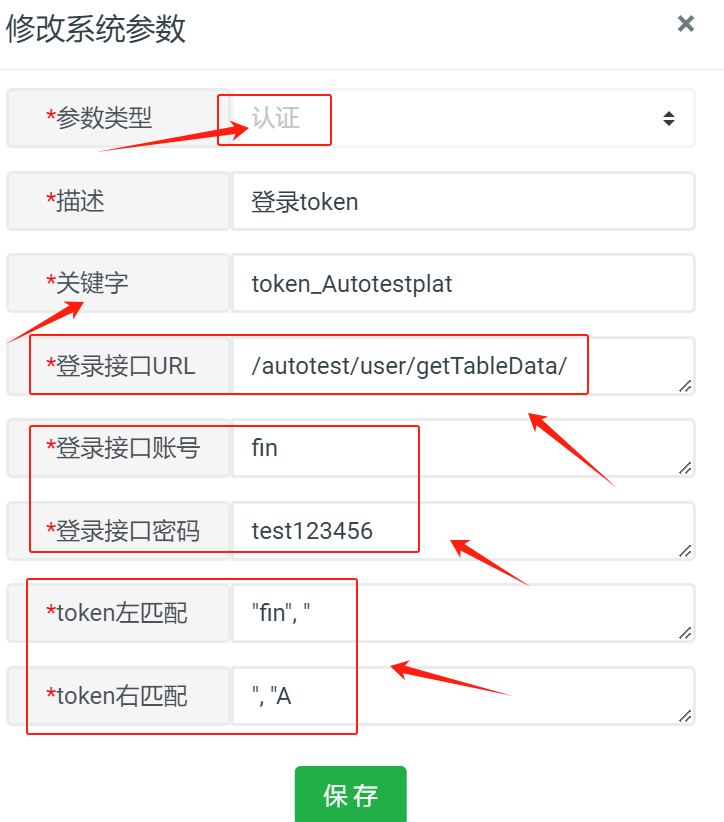
▲图 3.2

3.3、环境设置参数化，环境设置其实就是常量，只是为了方便区分，它专用于环境主机host地址的专用设置



▲图 3.3

3.4、token认证参数化，正则表达式参数化其实就是变量，也是为了便于区分，它专用于在响应数据中动态取值，例如token取值，登录接口执行后需要把动态的token值从响应数据中匹配出来，就可以采用正则表达式的方式提取，另外还可以用jsonpath的方式



▲图 3.4

系统参数设置好以后，在接口测试用例各个参数里面{关键字}进行读取。

## 测试用例

4.1、接口录入，

测试环境，是系统设置里面的环境设置参数，{关键字}；

请求，包括请求方式和url

请求Body，

参数，

认证，

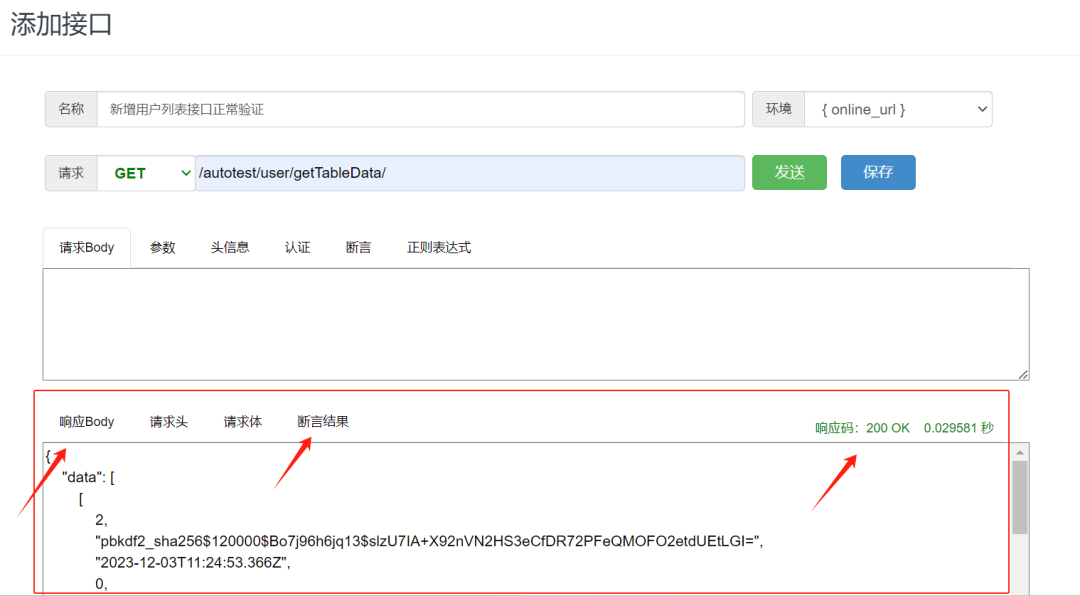
断言，目前断言进包含一个字段的设置；

正则表达式， 正则表达式是对当前接口返回值设置参数后，后续接口请求中可用 {关键字} 动态取值，通常在流程接口中需要使用到。



▲图 4.1

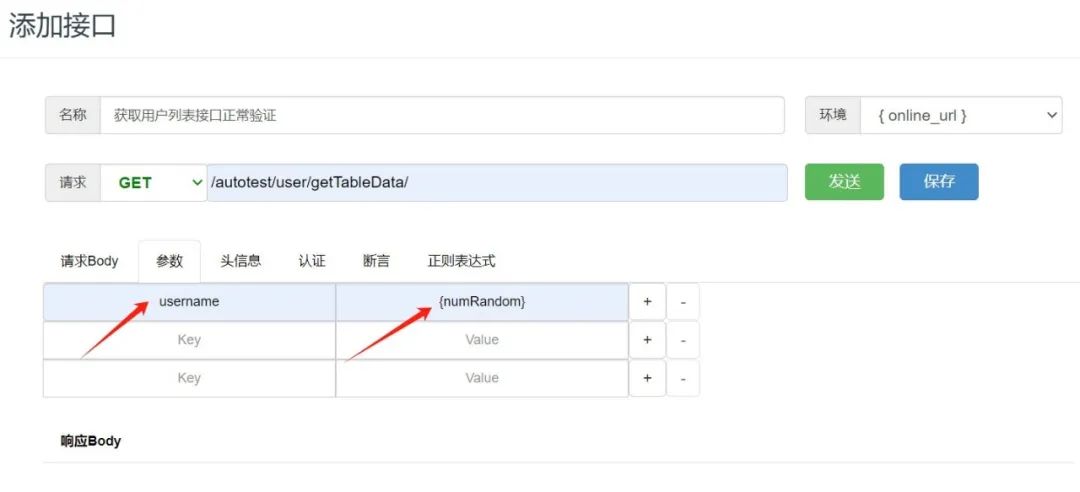
在如上图4.1所示，在各个字段填写正确接口信息，点击发送，可以看到响应Body一栏，出现接口的响应信息，如下图4.2所示：



▲图 4.2

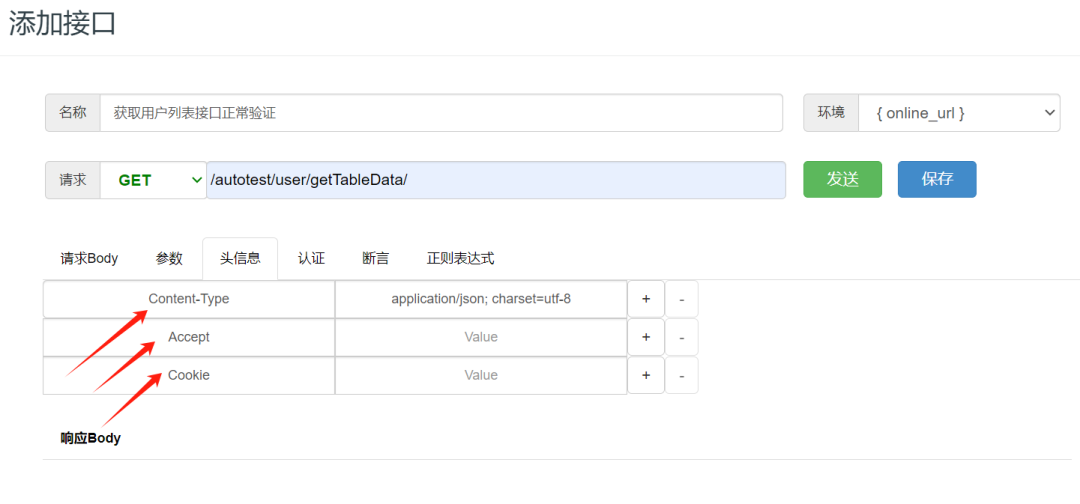
通常接口录入时，如果是post请求且是json数据则在请求Body里面填入。

如果是get请求，或者post请求的form格式数据，则在参数一栏填写，如下图4.3所示：



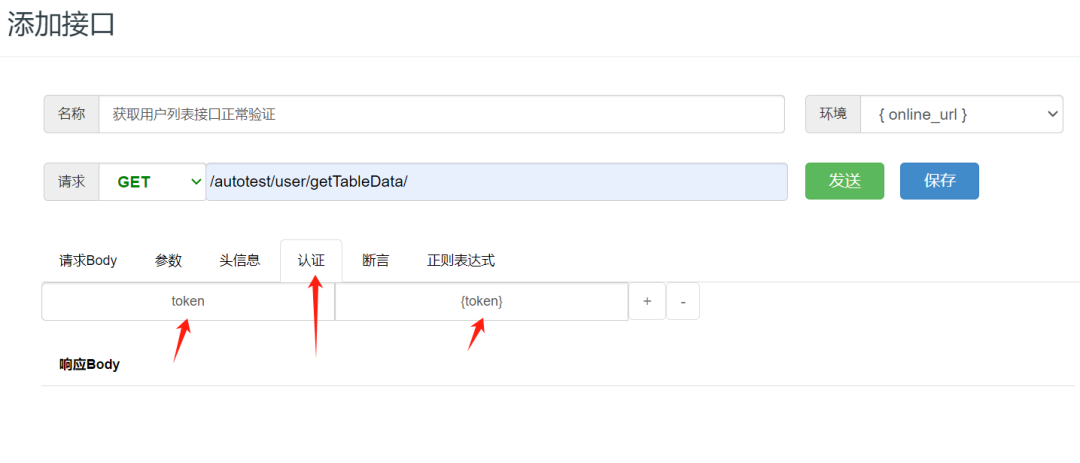
▲图 4.3

头信息 默认设置了3个参数，全删或不填再次编辑时默认是3个字段，如下图4.4所示:



▲图 4.4

认证 一般填token参数，这参数在是上一个登录接口返回值中用正则表达式关键字设置为token，先发送执行上一个接口，则在当前接口中通过认证可以获取token动态值。后面会把token设置在系统参数设置项里面就会更方便一些，不用每次在接口列表中先执行登录接口获取，而是在系统参数里面自动执行登录接口来获取token，功能还在开发中。如下图4.5所示:



▲图 4.5

断言 是输入要判断的响应数据，如下图4.6所示:



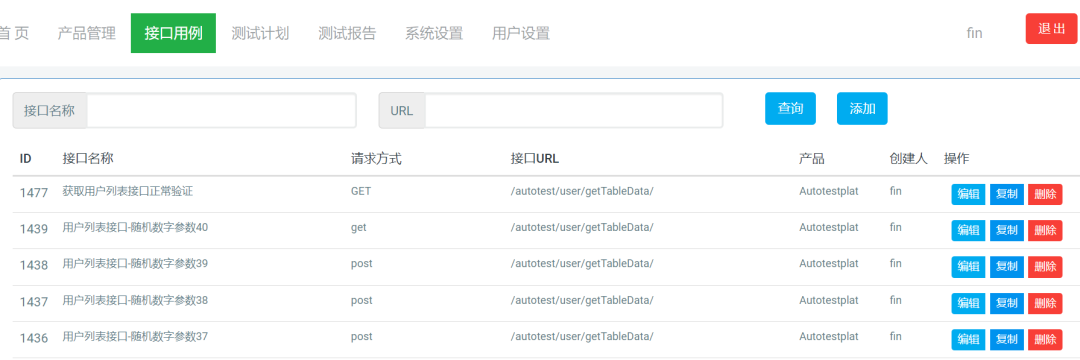
▲图 4.6

正则表达式，填写关键字，如token等，索引是指有多个匹配值时，匹配第0个，第1个，第2个，一般输入0或1，或2，如下图4.7所示:



▲图 4.7

发送，调试通过后，点击保存。此时即完成了接口自动化测试用例的录入。



▲图 4.8

4.2、接口编辑

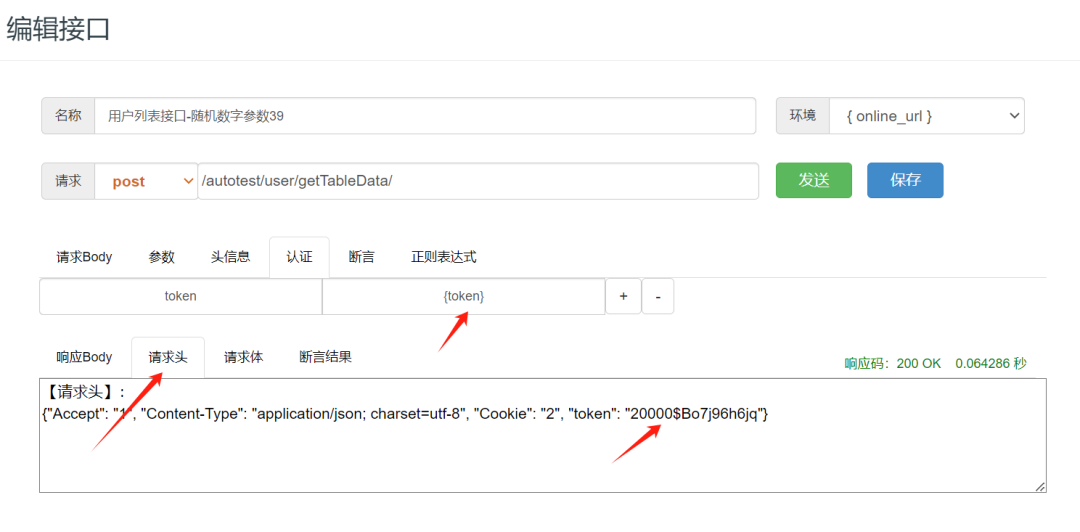
在上图4.8用例列表中，可以编辑和复制，界面信息和新增是一样的。点击编辑，则进入编辑页面，如果接口内容存在一部分相同内容，则可以点击复制来进行新增用例的操作，提高接口用例录入的效率，非常的方便。

如果redis没启动，或者token接口没有执行，则在发送执行当前有token参数的接口时，接口执行速度会很慢，执行完成后会响应Body中报相应error信息，如下图4.9所示:

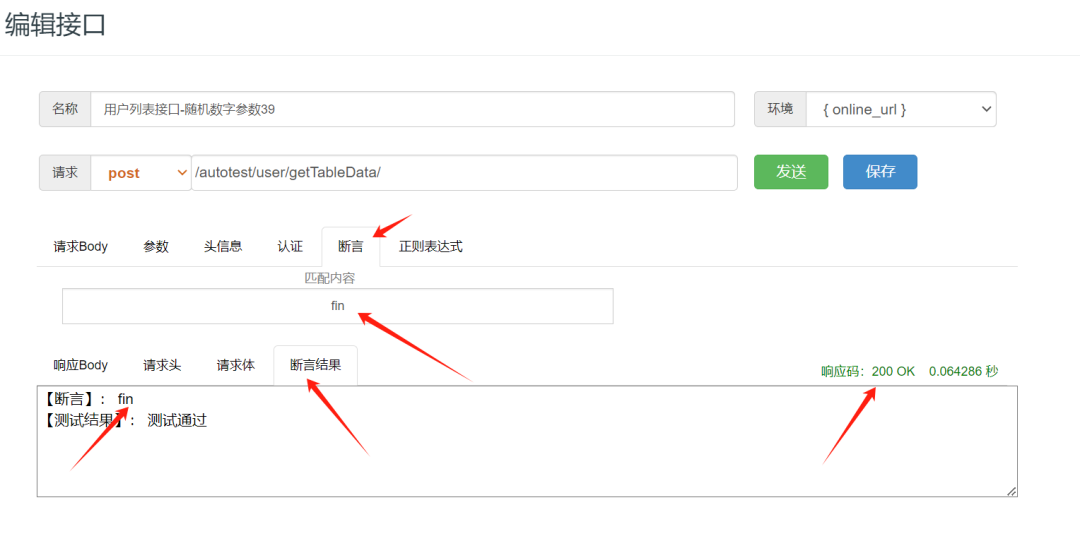


▲图 4.9

这时，你再可以启动redis以及执行有token正则表达式的登录接口，再来执行当前接口，就会不会报异常了，你在请求头中可以看到token的具体值，以及断言结果，响应码和响应时间等接口执行结果信息，如下图4.10，4.11所示:



▲图 4.10



▲图 4.11

接口的参数化值，通过{关键字}获取，参数在系统设置里面。

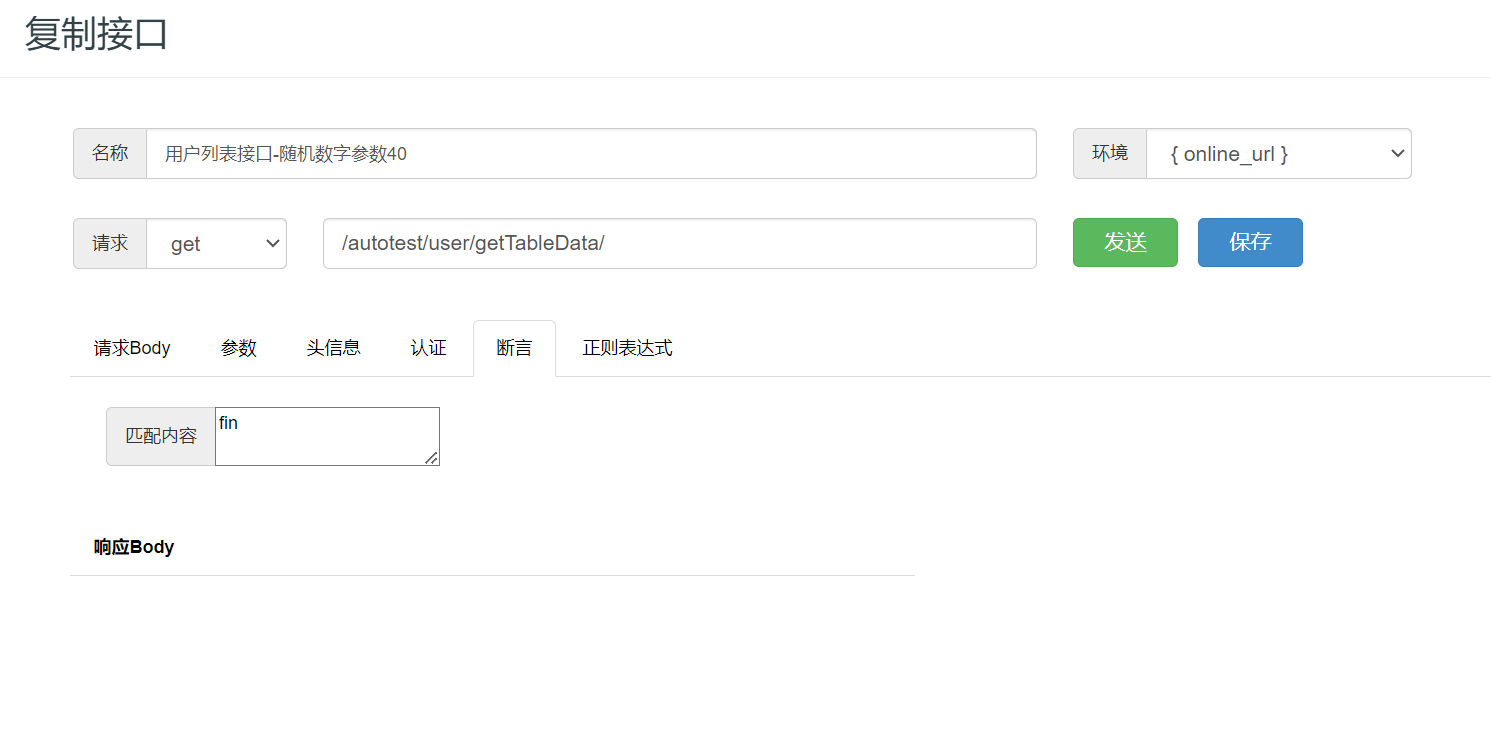
如下接口入参{numRandom}是表示一个动态的随机数字。



▲图 4.12

4.3、接口复制

当有类似或相同内容的接口信息时，通过接口复制以及修改部分内容来新增接口用例以及保存，可以节省接口录入时间，提高效率，非常实用。



▲图 4.13

## 测试计划

5.1、进入测试计划模块，点击“添加”，如下图5.1：



▲图 5.1

5.2、弹出添加测试计划页面，分别输入计划名称，选择测试执行环境，选择定时触发自动执行时间，如下图5.2：



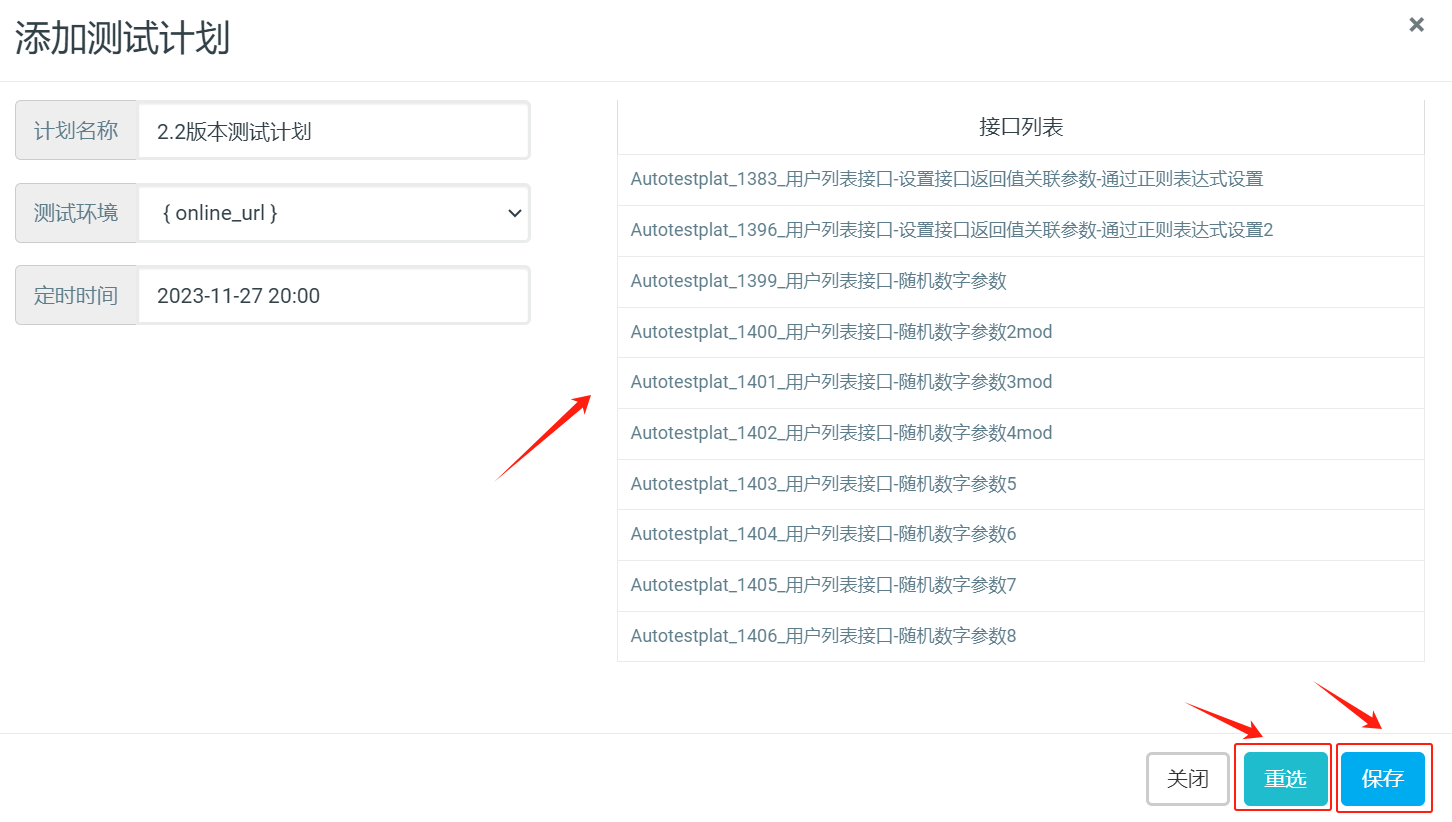
▲图 5.2

5.3 上图5.2中点击“选择接口”，弹出界面中，在第一页点击“全选框”，如下图5.3：

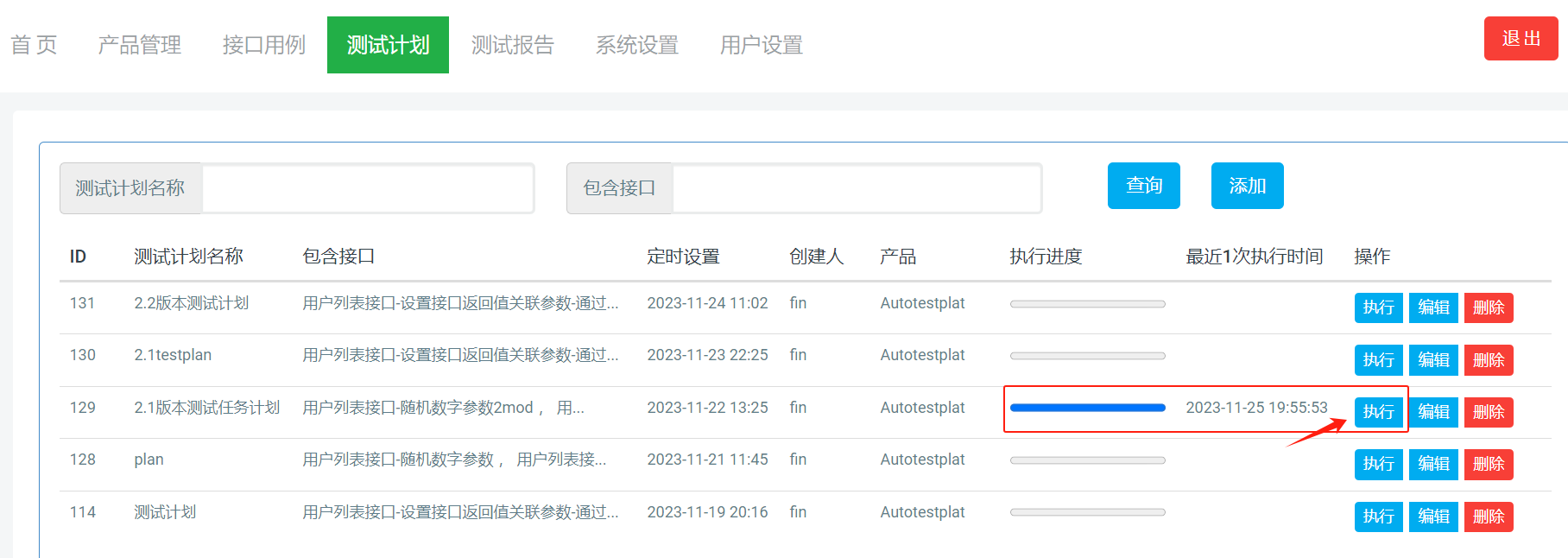
▲图 5.3

5.4、点击“下一页”，复选框勾选编号8、9、10一共10个用例，加入到测试计划中，点击“选择”，如下图5.4：

▲图 5.4

5.5、然后在测试计划页面，可以看到刚刚选择的10个用例，此时，如果想增加或减少用例，点击“重选”。如果确定就点击“保存”。到此测试计划添加创建完成，如下图5.5：

▲图 5.5

5.6、在测试计划模块，可看到刚刚创建的测试计划，可以点击“执行”，如下图5.6，然后等待，执行进度会实时显示执行进度，也可以在django服务启动的后台查看接口运行情况和进度，完成后提示执行成功，可进入测试报告模块查看该测试计划的报告。

▲图 5.6

5.7、然后是编辑，测试计划页面点击“编辑”，如下图5.7：

▲图 5.7

5.8、进入编辑页面，通常情况，需要对测试计划增加用例，减少用例，以及接口有改动。那么测试计划里面要相应的编辑。如下图5.8，在第5行处，点击“+”，会在下面新增一行，点击“-” 会删除当前这一行

▲图 5.8

5.9、在新增的这一行，即第6行处，点击“选择接口”，如图5.9：

▲图 5.9

5.10、在编辑选择接口页面，选择编号为13的接口，点击“选择”，如下图5.10：

▲图 5.10

5.11、在测试计划编辑页面，就可以看到刚刚新增的这条用例，然后点击勾选调试，执行完成后可以看到状态为测试通过，日志输出最后一条用例的执行信息，当显示测试失败时，根据右侧执行日志，在接口用例中重新修改接口信息，再次添加调试，当然，也可以去掉勾选不执行当前这条用例继续调试其他的接口用例。如下图5.11：

▲图 5.11

5.12、然后点击“保存”，此时，新增和删除的一条用例的编辑就完成了。接下来看看修改了接口如何编辑，首页在测试用例模块，修改3条用例的名称或参数，且这3条接口用例在以上测试计划中已加入，修改完用例后，再在测试计划里点“编辑”，此时，刚刚修改的3个测试用例后面，出现了“更新”，如下图5.12：

▲图 5.12

5.13、然后在要同步要更新的接口用例后面点击“更新”，完成后会提示该接口已更新，已保存！然后点击确定。如下图5.13：

▲图 5.13

5.14、在编辑测试计划页面，可以看到刚刚点击的接口名称已经更新成了之前在测试用例模块修改的测试用例名称。

▲图 5.14

5.15、最后就是查询、删除测试计划，这个就没什么好讲的了，这里删除测试计划后，之前执行该测试计划的测试报告不受影响，依然是保留在测试报告模块里面。

## 测试报告

6.1、根据id，测试计划，执行时间，查看测试报告列表



▲图 6.1

6.2、根据报告id，详情查看报告的详细接口自动化测试结果信息



▲图 6.2

6.3、下载保存为报告id的html文件，以及详细接口自动化测试结果信息数据



▲图 6.3

## 性能测试

性能测试基于jmeter引擎，目前版本暂时没有设计存储数据库表，因此测试过程和结果数据都是属于实时一次性的。环境需要安装jdk1.8版本，下载当前jmeter版本是5.6.2，把jmeter文件以及包含的报告汉化包放到Autotestplat目录下，启动django，登录自动化测试平台系统。

7.1，点击**“选择”**，看到选择的接口列表下会显示相应接口名称，要移除选择的话，在选择的接口列表点击相应的接口即可；然后在设置行输入并发线程数、持续时间，如下图7.1所示



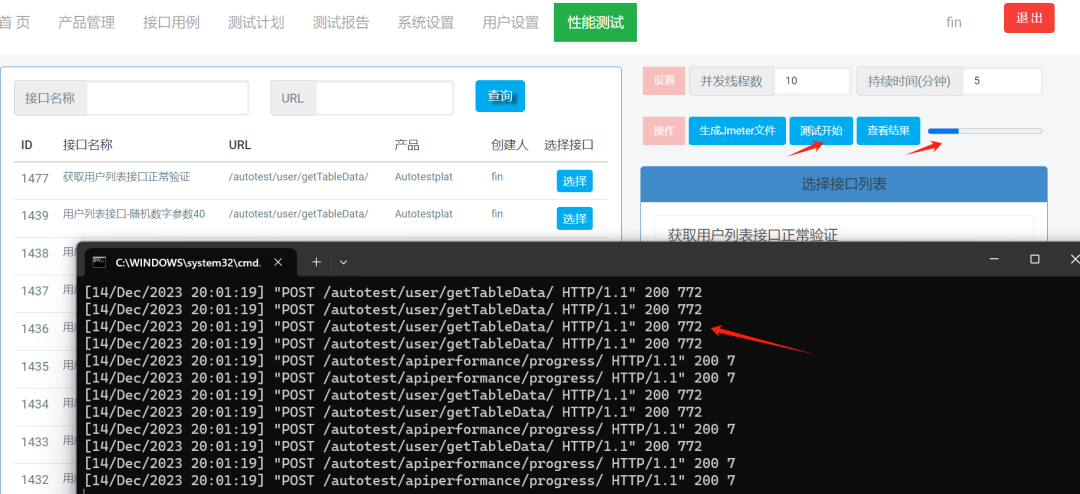
▲图 7.1

7.2   点击**“生成Jmeter文件”**，接口运行没问题的话，几毫秒时间，界面就右侧会显示 状态：已生成；生成的apitest.jmx文件在Autotestplat\apache-jmeter-5.6.2\bin目录；如下图7.2所示：



▲图 7.2

7.3，点击**“测试开始”**，看执行进度条，以及后台并发压测的接口请求，如下图7.3所示：



▲图 7.3

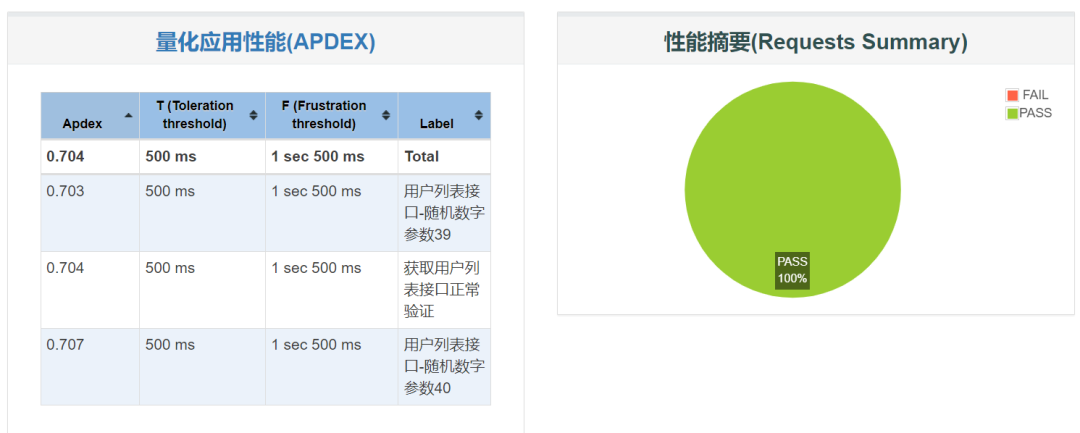
7.4，执行完成后，右侧会显示 状态：测试完成，如下图7.4所示

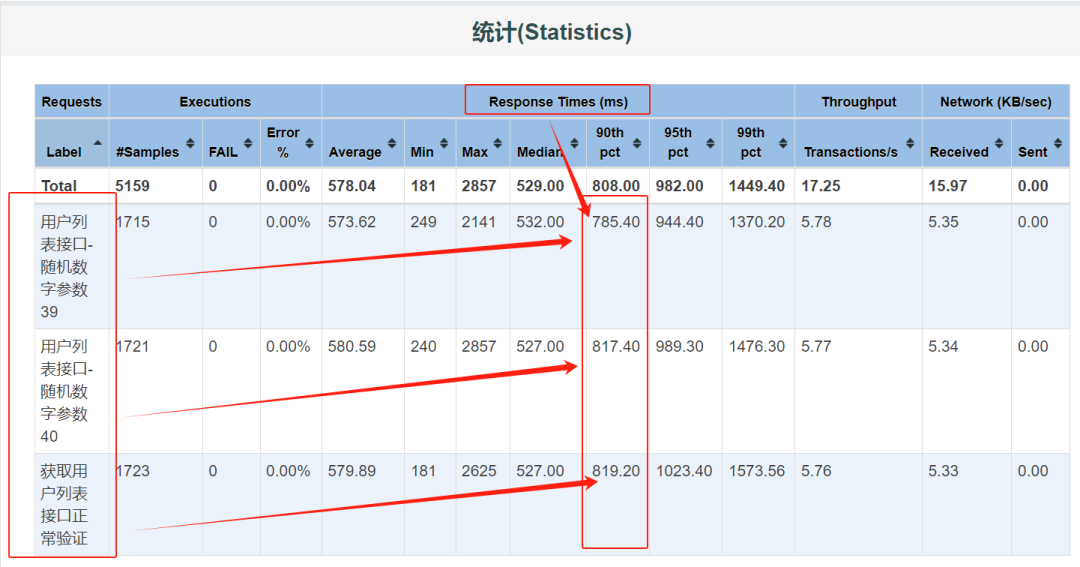


▲图 7.4

7.5，点击**"查看结果"**，概述，图表中可以看响应时间，TPS性能测试结果指标等。如下图7.5







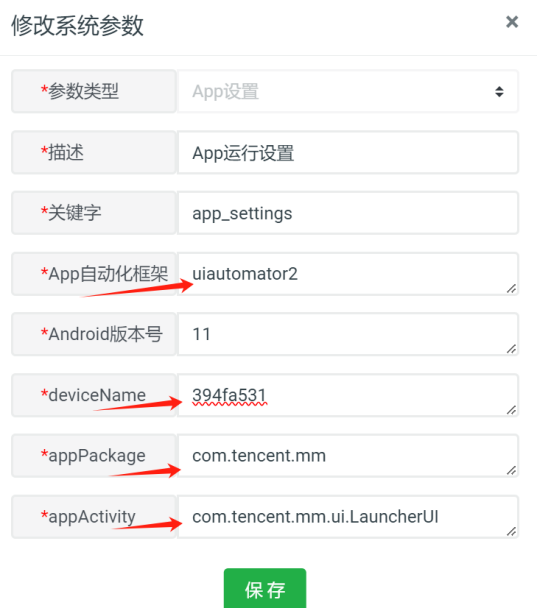
▲图 7.5

## AppUI测试

安装pip install -U uiautomator2

安装pip install -U weditor

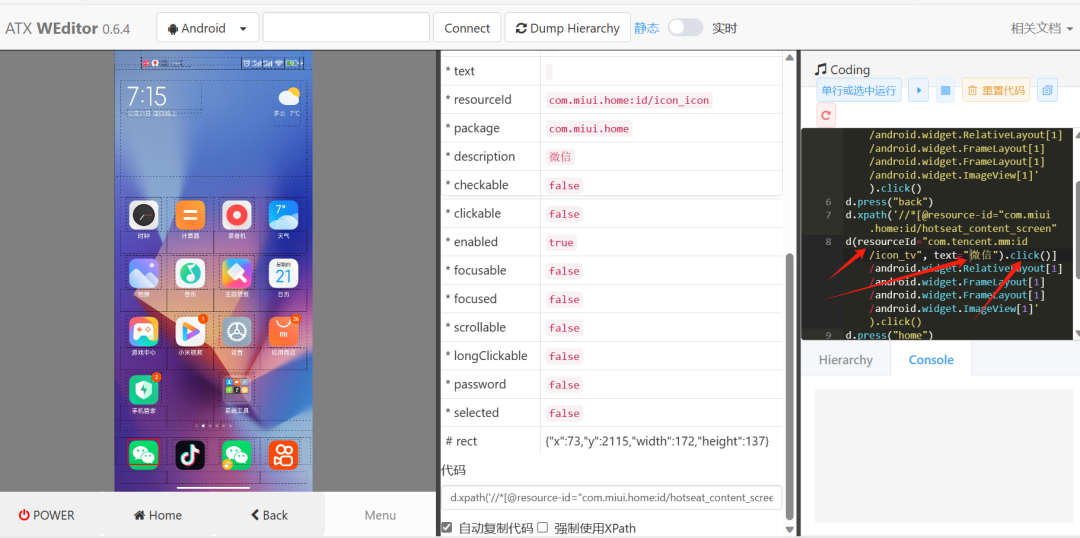
1，在系统设置中，选择App设置项，填写自动化框架，appium或uiautomator2,如果是appium则要必须输入Android版本号，手机序列号，待测app包名和初始页activity。基于原生的uiautomator相对简单一些，所以这里默认设置为它。如下图8.1所示：



▲图 8.1

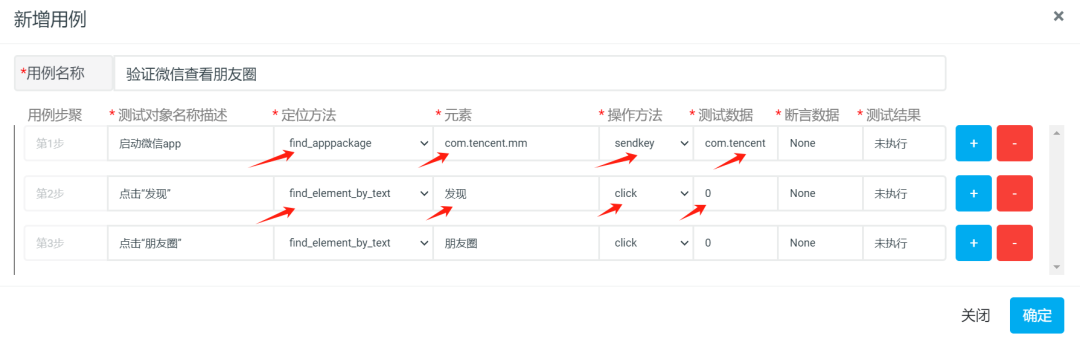
2、手机开启开发者模式，开启debug开关，连接电脑。python -m wedtor，允许手机存储文件模式，以及自动安装atx到手机并启动，然后在电脑上可对手机微信app页面元素控件抓取。如下图8.2所示：

（图片有点小，可点击扩大查看）



▲图 8.2

3、新增AppUI自动化操作微信，查看朋友圈的用例，在AppUI中，点击“新增”，输入用例名称，第1步输入对象名称为启动微信、选择正确的定位方法、元素、操作方法和测试数据，点击“+”，第2步参考第一步的输入，点击发现，第3步点击朋友圈，则用例完成。如下图8.3所示：



▲图 8.3

4，执行用例，连接和设置手机，执行过程如果步聚都为未执行，则该用例测试结果会显示未执行，如果有步聚是失败的，则该用例会显示红色fail，如果每个步聚都是pass，则该用例显示绿色pass。如下图8.4所示



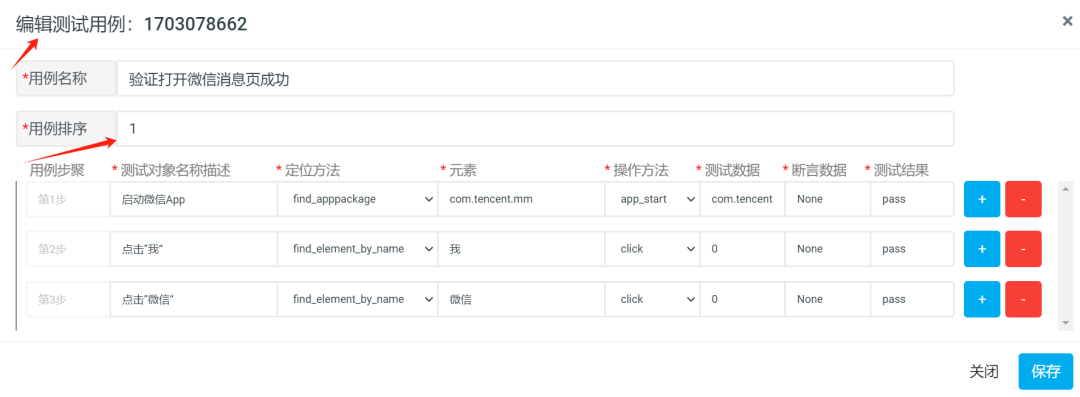
▲图 8.4

5，如果新增时用例有相同或相似步聚，则点击复制，然后修改保存即可，如下图8.4所示



▲图 8.5

6、修改用例时，多了一个字段，即排序，在全量执行用例时，会按照用例排序依次按顺序逐个执行每个用例。如下图8.5，8.6所示：



▲图 8.6

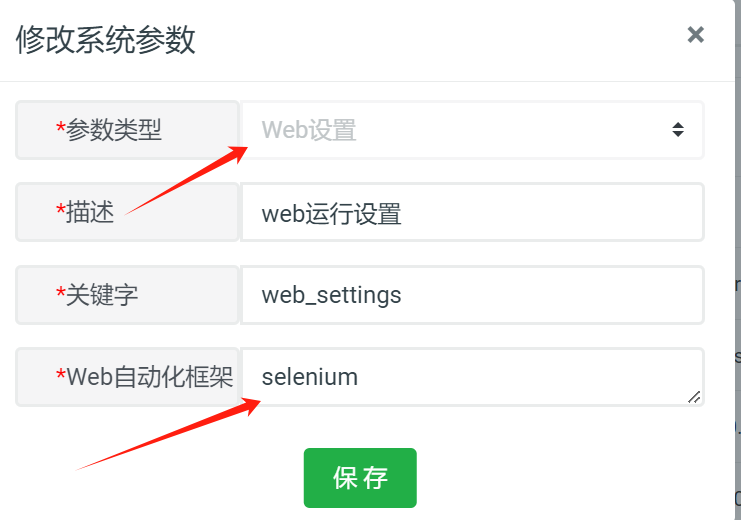


▲图 8.7

## WebUI测试

第1步：安装谷歌浏览器，版本为100.0.4896.75 ，驱动与版本一致即可。

第2步：系统设置中，设置默认web自动化框架为selenium，如下图9.1所示：

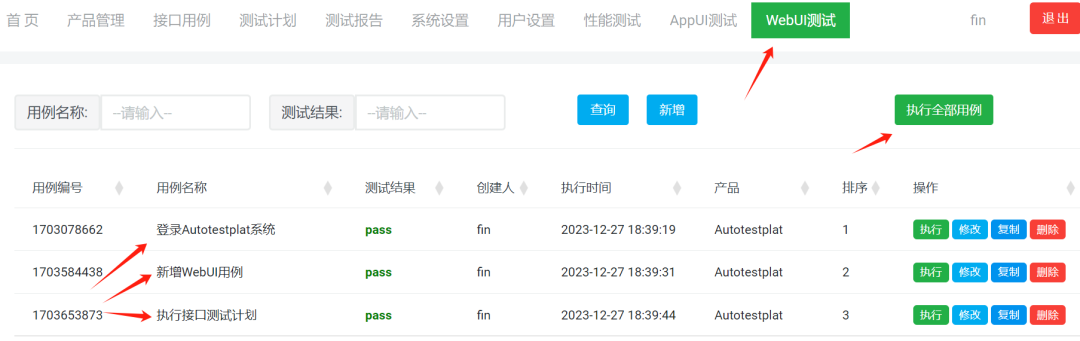


▲图 9.1

第3步：依次录入了3个不同的用例，登录Autotestplt系统，新增WebUI用例，执行接口测试计划。元素定位方式通常用id，name，xpath；操作方法通常用click和sendkey。如下图9.2，9.3所示：



▲图 9.2



▲图 9.3

第4步：执行单个用例，或全量执行用例，查看用例测试结果。

## 四、技术交流

