

## 18 | 眼前一亮：带你玩转GUI自动化的测试报告

2018-08-08 茹炳晟

软件测试52讲

[进入课程 >](#)



讲述：茹炳晟

时长 10:55 大小 5.01M



在 GUI 自动化测试系列的文章中，我围绕着 GUI 自动化测试进行了各种讨论：从最原始的 GUI 测试谈起，逐渐引入了脚本与数据的解耦，并谈论了页面对象模型，以及在此基础上的业务流程模型，接着分享了一些 GUI 自动化测试过程中的新技术，最后和你讨论了 GUI 自动化测试的稳定性问题。

今天，我会再和你聊聊 GUI 自动化测试过程中另外一个很实用的部分：GUI 自动化测试报告。

GUI 测试报告是 GUI 自动化测试的重要组成部分，当有任何的测试用例执行失败时，我们首先就会去分析测试报告，希望从中看到测试用例到底是在哪一步出错了，错误发生时被测系统是在哪个页面上，并且前序步骤又是哪些页面等等。

## 早期的基于视频的 GUI 测试报告

为了分析测试用例的执行过程与结果，早期就出现了基于视频的 GUI 测试报告。也就是说，GUI 自动化测试框架会对测试执行整个过程进行屏幕录像并生成视频。

这种基于视频的测试报告可以提供清晰的 GUI 测试执行上下文，看起来也很不错。但是，这种方式主要的问题是：

1. 报告的体积通常都比较大，小的几 MB，大的上百 MB，这对测试报告的管理和实时传输非常不利。
2. 分析测试报告时，往往需要结合测试用例以及服务端的日志信息，视频报告这一点上也有所欠缺。

所以，理想中的 GUI 测试报告应该是由一系列按时间顺序排列的屏幕截图组成，并且这些截图上可以高亮显示所操作的元素，同时按照执行顺序配有相关操作步骤的详细描述。

但是，早期的商业 GUI 自动化测试软件也只是具备最基本的顺序截图，并不具备高亮所操作元素的功能，后来商用工具厂商根据用户的实际使用反馈，逐渐完善和改进。

目前，商业的 GUI 自动化测试软件，比如使用最为广泛的 UFT（就是以前的 QTP），已经自带了截图以及高亮显示操作元素功能。也就是说，使用 UFT 执行一个 GUI 自动化测试用例，你无需做任何额外的工作，就能得到一份比较理想的 GUI 测试报告。

## 开源 GUI 测试框架的测试报告实现思路

但是，如果你使用的是开源软件，比如 Selenium WebDriver，那就需要自己去实现截图以及高亮显示操作元素的功能。实现的思路通常是：

利用 Selenium WebDriver 的 screenshot 函数在一些特定的时机（比如，页面发生跳转时，在页面上操作某个控件时，或者是测试失败时，等等）完成界面截图功能。

具体到代码实现，通常有两种方式：

1. 扩展 Selenium 原本的操作函数；
2. 在相关的 Hook 操作中调用 screenshot 函数。

下面，我会分别针对这两个实现方式，给出具体的示例，帮你理解并实现这个功能。

## 第一，扩展 Selenium 原本的操作函数实现截图以及高亮显示操作元素的功能

既然 Selenium 原生的 click 操作函数并不具备截图以及高亮显示操作元素的功能，那我们就来实现一个自己 click 函数。

当自己实现的 click 函数被调用时：

首先，用 Javascript 代码高亮显示被操作的元素，高亮的实现方式就是利用 JavaScript 在对象的边框上渲染一个 5-8 个像素的边缘；

然后，调用 screenshot 函数完成点击前的截图；

最后，调用 Selenium 原生的 click 函数完成真正的点击操作。

那么，以后凡是需要调用 click 函数时，都直接调用这个自己封装的 click 函数，直接得到高亮了被操作对象的界面截图。

如图 1 所示，就是用这种方式产生的界面截图，图中依次显示了登录过程中每一个操作的控件，第一张高亮了“Username”的输入框，因为自动化代码会在“Username”框中输入用户名；第二张高亮了“Password”的输入框，因为自动化代码会在“Password”框中输入密码；第三张高亮了“Sign in”按钮，因为自动化代码会去点击这个按钮。

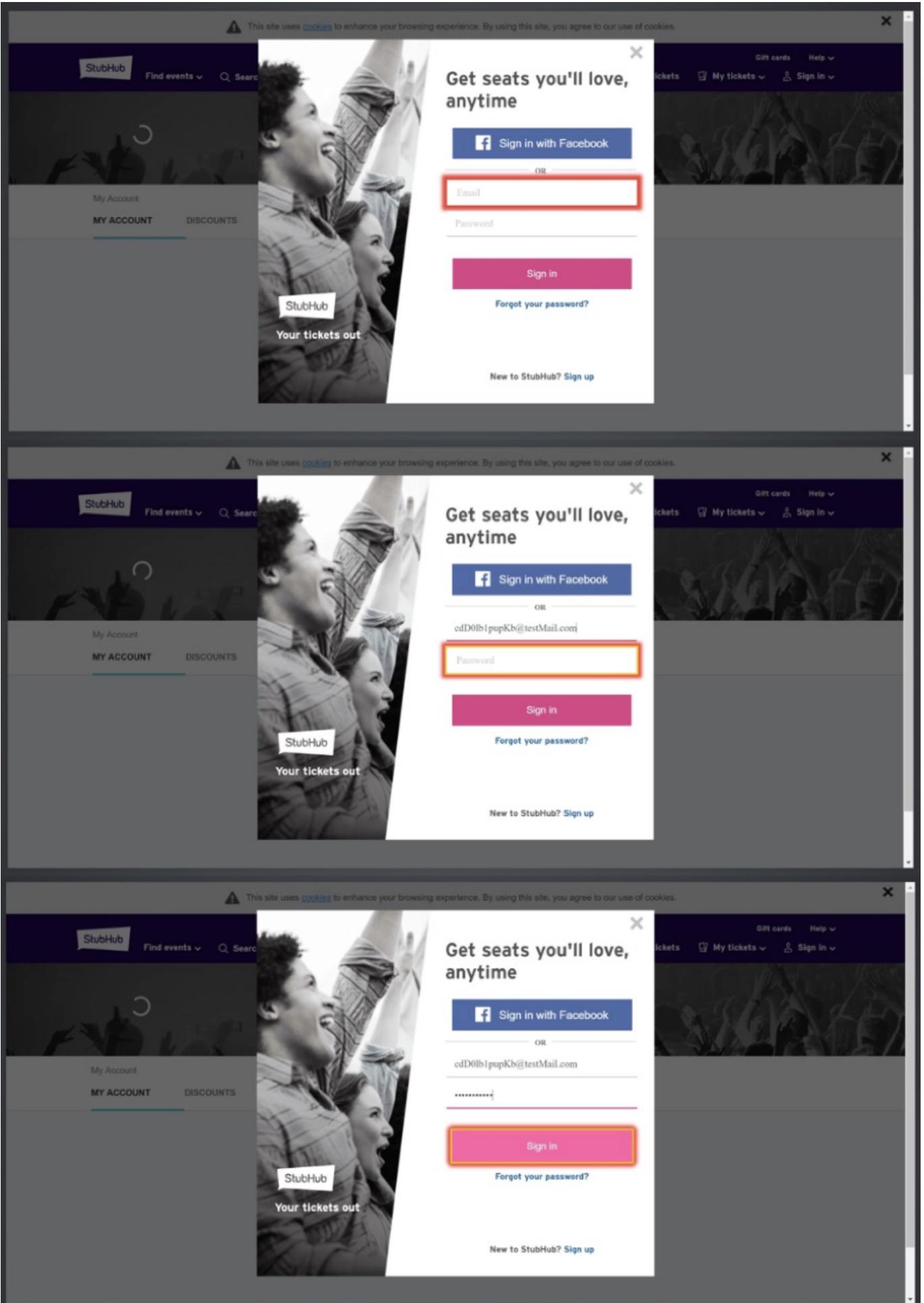


图 1 GUI 界面的时间顺序截图示例

## 第二，在相关的 Hook 操作中调用 screenshot 函数实现截图以及高亮显示操作元素的功能

其实使用 Hook 的方法比较简单和直观，但是你首先要理解什么是 Hook。

Hook 中文的意思是“钩子”，直接通过定义介绍什么是“钩子”会有些难以理解，那么我就通过一个实例来跟你解释一下。

当执行某个函数 F 时，系统会在执行函数 F 前先隐式执行一个空实现的函数，那么当你需要做一些扩展或者拦截时，就可以在这个空实现的函数中加入自定义的操作了。那么这个空实现的函数就是所谓的 Hook 函数。

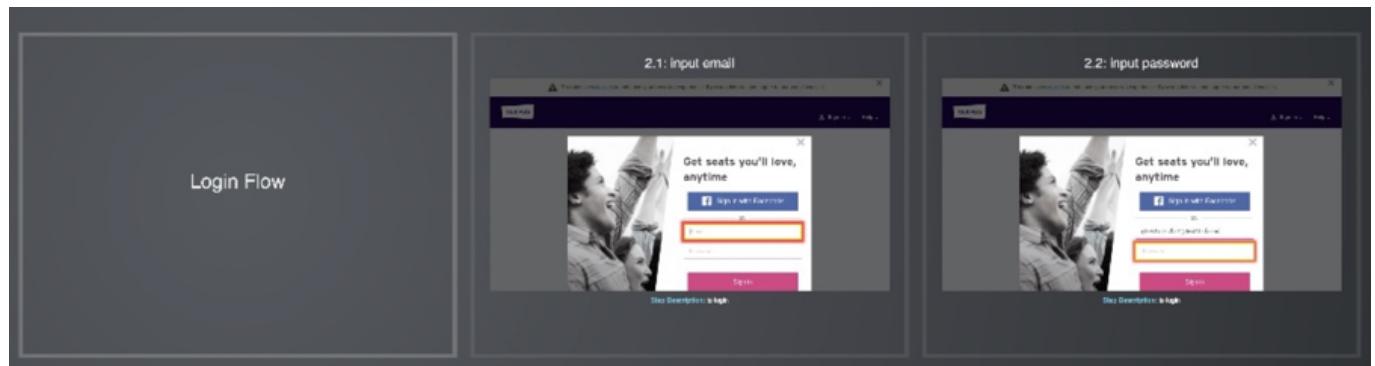
这样的例子有很多，比如 Java 的 main 函数，系统在执行 main 函数之前会先在后台隐式执行 premain 函数；JUnit 和 TestNG，都有所谓的 BeforeTest 和 AfterTest 方法，这些都是可以在特定步骤的前后插入自定义操作的接口。

说到这里，你可能已经知道要怎么做了：我可以在这些 Hook 函数中添加截图、元素高亮，以及额外的任意操作，比如更多的详细日志输出等等。

另外，我在前面的文章中分享了基于业务流程的脚本封装，你可以再思考一下，如何在 GUI 报告中体现出业务流程的概念，这样的测试报告会具有更好的可读性。

比如，图 2 所示的 GUI 测试报告就显示了具体的 Flow 名称。这个功能，就是通过 Hook 函数实现的。

具体的实现逻辑也比较简单的，就是在 Flow 开始的第一个 Hook 函数中调用增加报告页的函数，并在这个新增的报告页中输出 Flow 的名字。



## 图 2 在 GUI 测试报告中体现业务流程

上面所讲的 GUI 测试报告都是针对一个国家的，当面对多个国家站点的 GUI 测试时，事情就会变得更加复杂，你就必须去考虑全球化 GUI 测试报告应该如何设计。

## 全球化 GUI 测试报告的创新设计

所谓全球化测试是指，同一个业务在全球各个国家都有自己网站。比如，一些大型全球化电商企业在很多国家都有自己的站点，那么对这些站点的测试除了要关注基本的功能，以及各个国家特有的功能外，还要去验证界面布局以及翻译在上下文环境中是否合适。

早期的做法是，雇佣当地的测试工程师，由他们手工执行主要的业务场景测试，并验证相关的页面布局，以及翻译内容与上下文中的匹配度。在当地专门雇佣的这些测试工程师，被称为 LQA。

显然，聘请 LQA 的效率非常低，主要原因是：全部测试工作都由 LQA 在项目后期手工执行，执行前还需要对他们进行业务培训；同时，我们需要准备非常详尽的测试用例文档，LQA 也要花很大的精力去截图并完成最终的测试报告。

为了解决这种低效的模式，最好的解决方法就是：利用 GUI 自动化测试工具生成完整的测试执行过程的截图。这样，LQA 就不再需要去手工执行测试用例了，而是直接分析测试报告中业务操作过程中 GUI 界面截图就可以了，然后发现页面布局问题或者是不恰当的翻译问题。

这个方案看起来已经比较完美了，LQA 的工作重点也更清晰了，但这并不是最优的方案。因为这些 LQA 在实际工作中，还会有以下三个比较痛苦的地方：

1. 需要经常在多个国家的测试报告之间来回切换去比较页面布局；
2. 需要频繁切换到美国网站（也就是主站）的报告，去比较翻译内容与上下文的匹配度；
3. 发现缺陷后，还是需要从 GUI 测试报告中复制截图，并用图像软件标注有问题的点，然后才能打开缺陷管理系统递交缺陷报告。

为了解决这三个问题，我建议你建立以下形式的测试报告。这里有一张图片展示了一份包含多国语言比较报告的示例，听音频的用户可以点击文稿查看如图 3 所示。

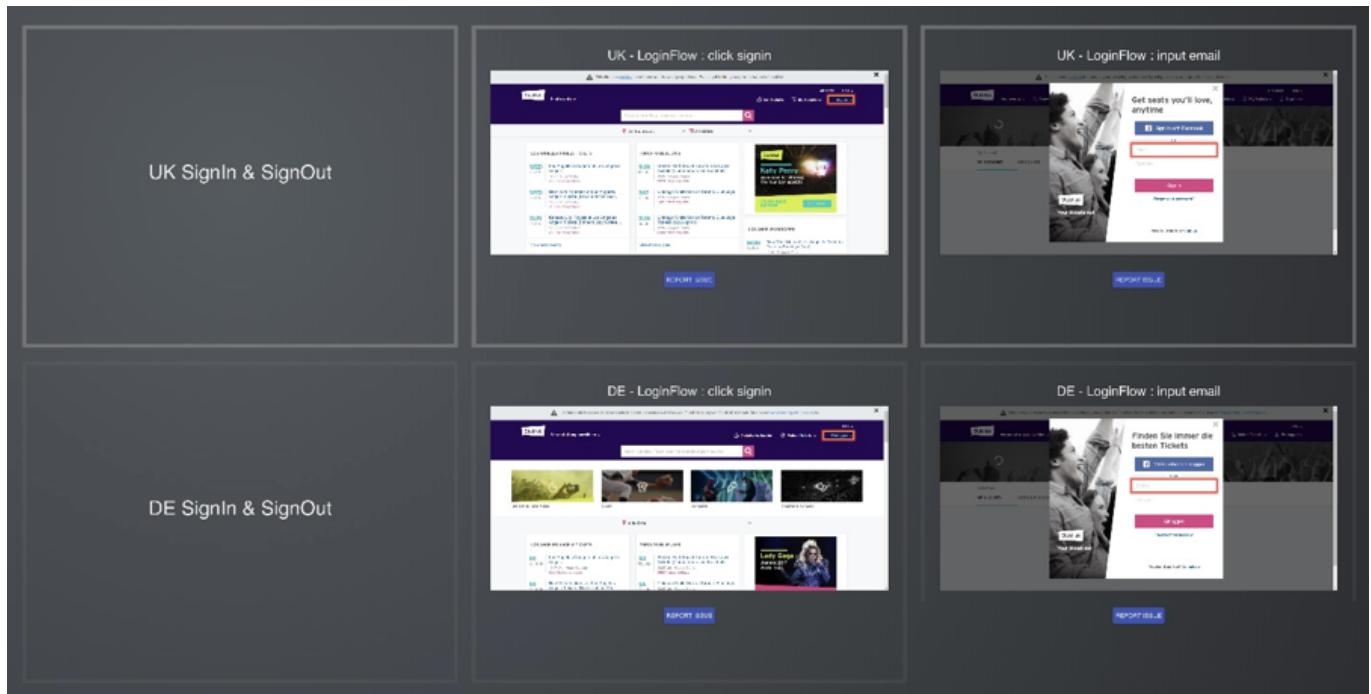


图 3 多国语言比较报告

报告的横向，是一个国家的业务测试顺序截图，比如图中第一行是英国网站的登录业务流程顺序截图，第二行是德国网站的登录业务流程顺序截图。报告的纵向，展示的自然就是同一界面在不同国家的形式了。

整个报告可以用键盘上下左右依次移动。可想而知，这样的 GUI 测试报告设计一定可以大幅提高 LQA 的效率。

同时，由于这个 GUI 测试报告是基于 Web 展现的，所以我们可以直接提供递交缺陷的按钮，一旦发现问题直接递交缺陷，同时还可以把相关截图一起直接递交到缺陷管理系统，这将更大程度地提高整体效率。

那么，怎么才能在技术上实现测试报告和缺陷管理系统的交互呢？其实，现今的缺陷管理系统往往都有对外暴露 API 接口，我们完全可以利用这些 API 接口来实现自己的缺陷递交逻辑。

这种测试报告的形式就是 eBay 在全球化站点测试中采用的方案，目前已经取得了很好地效果，降低了工作量的同时，还大幅度提高了全球化测试的质量。

## 总结

好了，希望上面的测试报告设计方法可以让你有眼前一亮的感觉。接下来，我总结一下今天的主要知识点。

早期基于视频的 GUI 测试报告由于体积较大，而且不能比较方便地和日志适配，所以并不是最好的解决方案。理想的 GUI 测试报告应该是由一系列按时间顺序的屏幕截图组成，并且可以在这些截图上高亮你所操作的元素，同时按照执行时序配有相关操作步骤的详细描述。

商业 GUI 自动化测试框架的 GUI 测试报告已经做得非常成熟，通常不需要做额外的定制或者开发。但是开源 GUI 自动化测试框架的 GUI 测试报告往往需要自己来开发，主要使用了扩展 Selenium 原本的操作函数的方式以及 Hook 函数来实现。

最后，我介绍了 eBay 面对全球化测试过程中 GUI 测试报告的创新设计，希望你也可以借鉴这种方法。

## 思考题

如果自己去开发 GUI 测试报告的功能，你还能想到其他更多实用的功能吗？你又是如何实现这些功能的？

欢迎你给我留言。

The image is a promotional graphic for a software testing course. It features a portrait of a man with glasses and a black t-shirt on the right side. On the left, there's a logo for '极客时间' (Geek Time) with a stylized orange 'G' icon. The main title '软件测试52讲' is displayed in large, bold, dark blue text. Below it, the subtitle '从小工到专家的实战心法' is shown in a smaller, dark blue font. At the bottom left, the author's name '茹炳晟' is listed along with their title 'eBay中国研发中心 测试基础架构技术主管'. A call-to-action at the bottom encourages users to upgrade to a new version and share it with friends for free reading.

新版升级：点击「 请朋友读」，10位好友免费读，邀请订阅更有**现金奖励**。

上一篇 17 | 精益求精：聊聊提高GUI测试稳定性关键技术

下一篇 19 | 真实的战场：如何在大型项目中设计GUI自动化测试策略

## 精选留言 (14)

写留言



sylan215

2018-08-08

27

1. 说起操作步骤截图功能，我记得 Windows7 系统开始，自带一个「问题步骤记录器」，直接在运行框输入 psr.exe 就可以调起，开启监控后，所有操作步骤的说明和截图都会保存成 MHTML 文件，结束时会打包到一个压缩包中，贼好用，没用过的推荐试一下。
  2. 不管是视频还是步骤截图，我理解的应该是不需要所有的都进行记录，而是在出现问题需要定位的时候才需要记录，其他大部分地方都是在自动化操作完之后，有一个校验操...
- [展开 ▼](#)



Cynthia◆...

2018-08-09

3

我的理解是只有出现问题的时候才需要截图，不过全截图也OK，只要在报告中可以点击指定用例看到对应截图就可以了。  
另外就是因为截图比较占空间，应该也需要对应的清理功能，比如说没有问题的用例，隔一段时间就把对应的截图清理掉，或者是把所有的截图清理掉，不占用空间。

[展开 ▼](#)



dimi

2018-09-13

2

说这么多能不能show下真实代码呢 或者介绍一下主流框架怎么实现的，老是来需的，截个图了事。还有测试报告不只是截图吧。



Robert小七

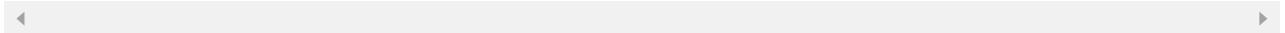
2018-08-08

2

我目前做的一般在定位元素失败时才会进行页面截图，测试报告也都是执行一个业务流程后将执行结果和异常信息写入到报告中，最后生成HTML形式的测试报告！

展开 ▼

作者回复: 失败时截图还是正常时都截图其实都是可以的，我建议在测试框架级别提供一个可配置的参数，按需设置即可。比如你想每个步骤都截图，那么需要把autoScreenshot设置为true，否则设置为false。这个参数可以通过测试框架的命令行暴露给用户。



**红娟**

2018-08-08

↑ 2

每次都有新的收获😊

展开 ▼



**小老鼠**

2018-10-25

↑ 1

能做到这点对于一个中小型公司来说非常了不得了，但我总觉得除了获得图片还可以获取一些别的信息，比如HTTP response 包，以便于开发修改缺陷。



**Struggling**

2018-08-11

↑ 1

打卡。之前没有GUI相关的测试经验，有个疑问，关于GUI的测试报告都是以截图的方式吗？会不会有相应法律图形界面显示总结性的报告，比如总共跑了多少个业务场景用例，通过多少失败多少等，甚至覆盖率等

展开 ▼



**hoppaz**

2018-08-09

↑ 1

文章开篇提到视频GUI测试报告看不到log，文章后面提到了截图高亮可以看到流程，但是似乎也没有提到如何在GUI测试报告中显示相应的log，请老师指导一下采取什么方案比较直观高效呢？

展开 ▼

作者回复: 由于基于截图的测试报告的具体数据结构其实是个大的json文件，json中的每个节点元素记录了时间戳，测试用例输出的log，截图文件的保存路径等信息，所以就可以很方便的在报告中同步显示对应的log信息。同时，有一点文中没有提到的是，同一个测试报告其实是可以按照不

同的view来展示的，一种是基于时序来显示截图的view，还有一种是按log顺序的文本view。两者可以来回切换，也可以结合在一起，比如基于时序截图的view中也可以同步显示log信息。



口水窝

2019-04-03



希望茹老师给以简单的实例操作，操作之后才能更深了解。

展开▼



ggggggijj

2018-12-06



ALLURE推荐

展开▼



subona

2018-11-05



学到了新的思路和方向

展开▼



海

2018-08-12



老师，图2是怎么实现的，非常好的创意

展开▼



缪情

2018-08-09



看了这篇文章后对测试报告的原理以及实现思路有了非常清晰的理解，每篇文章都收获满满



闹忒龙

2018-08-08



对performance test和load test比较感兴趣，不知道大公司都用的什么技术流程，老师能

不能讲讲：）

作者回复：非互联网公司很多是用loadrunner的，但是互联网公司由于需要的并发量太高，loadrunner已经不能满足，所以很多是用Jmeter，而且会用自己的方式来扩展Jmeter。当然现在云端还有很多收费的性能压测服务，比如阿里的pts等。这部分内容会在后面性能测试的章节详细来谈。

