Python+UIautomator使用流程

安装ADB 最新版本（环境变量配置）

ADB和Fastboot for Windows

<https://dl.google.com/android/repository/platform-tools-latest-windows.zip>

ADB和Fastboot for Mac

<https://dl.google.com/android/repository/platform-tools-latest-darwin.zip>

ADB和Fastboot for Linux

<https://dl.google.com/android/repository/platform-tools-latest-linux.zip>

python 第三方库安装

执行下述命令：pip install --pre uiautomator2#安装名为uiautomator2的第三方库

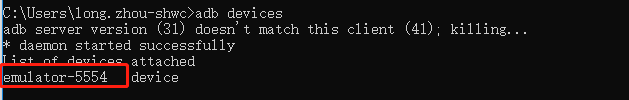
pip install pillow #安装pillow：图像处理标准库

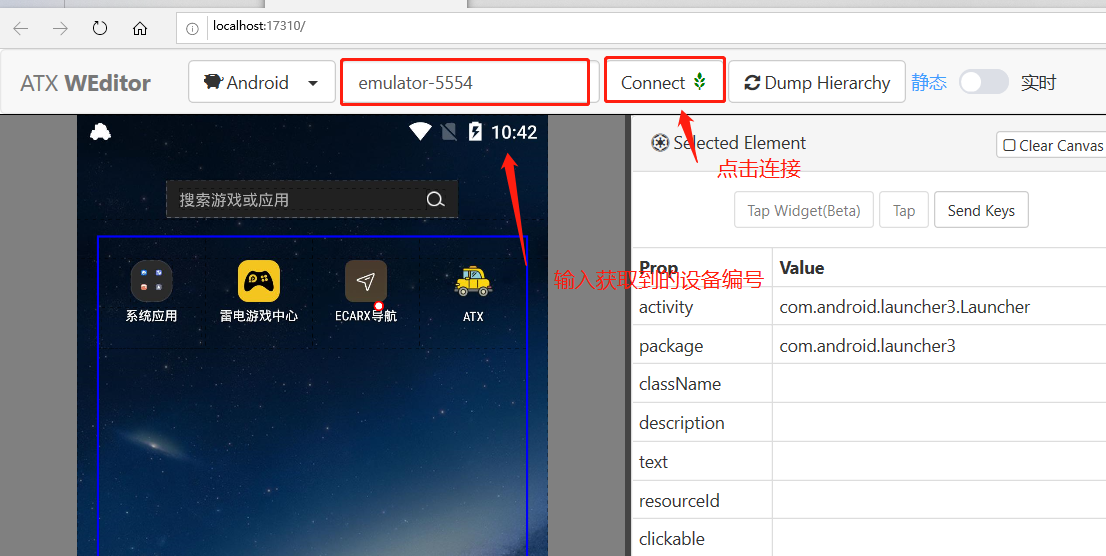
界面工具安装

pip install --pre --upgrade weditor#可在cmd中输入亦可在python中输入

cmd输入 python -m weditor 打开界面工具

python3 -m uiautomator2 init #初始化uiautomator2

另起cmd输入 adb devices 获取设备编号



在手机的屏幕上显示Toast

Toast显示

d.toast.show(“”)

d.toast.show(“”,1.0)#显示1.0秒，默认为1.0秒

**获取toast**

d.toast.get\_message(5.0, 10.0, "default message")

**获取后断言使用**

assert "Short message" in d.toast.get\_message(5.0, default="")

## xpath定位和常用方法

d.xpath("xpath轴']").click()#通过xpath定位

d(text="Settings").wait(timeout=10.0)#等待某个文本出现

d(resourceId="com.smartisanos.clock:id/text\_stopwatch").click()#通过resourceID定位

d(text="Settings").set\_text("你好")#文本输入

d.click(x，y)#通过坐标点击

d.double\_click(X，Y，0.1)#双击屏幕间隔时间0.1s

[d.long\_click(x，y)](https://www.jianshu.com/p/f0f03febd47d)#长按某个坐标点

[d.long\_click(X，Y，0.5)#0.5为长按屏幕的时间](https://www.jianshu.com/p/f0f03febd47d)

d.swipe(sx, sy, ex, ey)#传入开始坐标和结束坐标拖动屏幕

d.swipe(sx, sy, ex, ey, 0.5)#0.5代表拖动速度

d.press("home")#回到手机home页面

d.press("back")#手机的返回键点击

d.open\_notification()#打开通知栏

d.open\_quick\_settings()#打开通知栏设置d(className="android.widget.TextView").click()#通过cassname定位d(description="..").click()#通过其他description定位

1、安装，只指出url：d.app\_install（'http://some-domain.com/some.apk'）  
2、启动：d.app\_start（“com.example.hello\_world”）＃以包名称开头  
3、停止应用：d.app\_stop（“com.example.hello\_world”）  
         d.app\_clear（'com.example.hello\_world'）  
        停止所有应用：d.app\_stop\_all（）  
5、推送一个文件到手机：d.push（“foo.txt的”，“/ SD卡/”）  
6、推和重命名：d.push（“foo.txt的”，“/sdcard/bar.txt”）  
7、推送并更改文件模式：d.push（“foo.sh”，“/ data / local / tmp /”，mode = 0o755）  
8、从设备中拉取文件：d.pull（“/ sdcard / tmp.txt”，“tmp.txt”）＃如果文件在设备上找不到，FileNotFoundError将会出现   
    d.pull（“/ sdcard / some-file-not-exist.txt”，“tmp.txt”）  
7，应用连接会话：

1、启动应用：sess = d.session（“com.netease.cloudmusic”  
2、会话连接运行中的程序：sess = d.session（“com.netease.cloudmusic”  
3、检测应否崩溃：SESS（文字= “音乐”）点击（）  
    检查会话是否正常：sess.running（）  
8，手机硬件操作

1、d.screen\_on（）＃打开屏幕d.screen\_off（）＃关闭屏幕  
2、获取当前屏幕状态：d.info.get（'screenOn'）＃android 4.4  
3、按软/硬件  
d、press（“home”）＃按home键，用键名   
d、press（“back”）＃按返回键，与主要的名称   
d、press（0x07，0x02）＃按下键码0×07（“0”）与META ALT（0x02）  
d、unlock（）解锁屏幕                                                                                                                   
这些目前支持：  
home                    back                  left                  right  
up                        down  
center                   menu                 search             enter  
delete ( or del)          recent (recent apps)        volume\_up             
volume\_down         volume\_mute           camera    power  
9、事件：

支持百分比d.long\_click（0.5，0.5）

1、点击屏幕：d.click（X，Y）

2、长按屏幕：d.long\_click（X，Y）或d.long\_click（X，Y，0.5）＃长按0.5秒（默认）

3、拖动：d.drag（SX，SY，EX，EY）或d.drag（SX，SY，EX，EY，0.5）＃刷新0.5秒（默认）

4、刷卡：d.swipe（sx，sy，ex，ey）或d.swipe（sx，sy，ex，ey，0.5）＃刷新0.5秒（默认）

5、滑动：＃由点划动（X0，Y0）到点（X1，Y1），然后到点（X2，Y2）

     ＃时间将加速0.2秒bwtween两点

     d.swipe（（X0，Y0），（X1，Y1），（X2，Y2），0.2）

10，屏幕相关：

1、获取方向：d.orientation

2、设置方向：d.set\_orientation（ 'L'）

      值：natural或者n

       left或者l

       right或者r

        upsidedown或者u（不能设置）

3、不能选择/解冻：d.freeze\_rotation（）＃冻结旋转d.freeze\_rotation（False）

4、截屏：image = d.screenshot（）＃支持png和jpg

        image.save（ “home.jpg”）

5、截屏：

         import cv2

         image = d.screenshot（format ='opencv'）

         cv2.imwrite（'home.jpg'，image）

6、屏幕UI转xml：xml = d.dump\_hierarchy（）

7、打开通知或快速设置：d.open\_notification（）d.open\_quick\_settings（）

11、选择器：识别屏幕对象，请参阅UiSelector Java doc。

支持的参数：

text，textContains，textMatches，textStartsWith  
className， classNameMatches  
description，descriptionContains，descriptionMatches，descriptionStartsWith  
checkable，checked，clickable，longClickable  
scrollable，enabled，focusable，focused，selected  
packageName， packageNameMatches  
resourceId， resourceIdMatches  
index， instance   
1、选择：d（text ='Clock'，className ='android.widget.TextView'）

2、获取孩子或孙子：d（className =“android.widget.ListView”）.child（text =“Bluetooth”）

3、兄弟姐妹：d（文本= “谷歌”）兄弟（类名= “android.widget.ImageView”）。

4、获取子节点根据文本：d（className =“android.widget.ListView”，resourceId =“android：id / list”）\

  .child\_by\_text（“Bluetooth”，className =“android.widget.LinearLayout”）

5、获取子节点根据文本允许滚动搜素：d（className =“android.widget.ListView”，resourceId =“android：id / list”）\

  .child\_by\_text（ “蓝牙”，allow\_scroll\_search =真，类名= “android.widget.LinearLayout”）

6、支持多级：d（className =“android.widget.ListView”，resourceId =“android：id / list”）\

   .child\_by\_text（“Wi-Fi”，className =“android.widget.LinearLayout”）\

   .child（className =“android .widget.Switch”）\ .click（）

7、支持相对定位：

 •d（A）.left（B），在A的左侧选择B.

 •d（A）。右（B），在A的右侧选择B.

 •D（A）.up（B），在A之上选择B.

 •d（A）.down（B），在A下选择B.

 d（文= “无线网络连接”）。右（的className = “android.widget.Switch”）。点击（）

8、第一个文本带有“Add new”的实例：d（text =“Add new”，instance = 0）

9、在屏幕上查找：d（text =“Add new”）。count

 d（text =“添加新的”）[0]＃第一个

 d（text =“添加新的”）[1]＃第二个

10、显示控件信息：view.info

11、检查控件是否存在：d（text =“Settings”）.exists或者d.exists（text =“Settings”）

12、检查控件信息：d（text =“设置”）.info

13、文本控件操作：

 d（text =“Settings”）.get\_text（）＃获取小部件文本

 d（text =“Settings”）.set\_text（“My text ...”）＃设置文本

 d（text =“Settings”）.clear\_text）＃清除文字

14、点击：

 d（text =“设置”）.click（）

 ＃等待元素出现最多10秒钟，然后点击

 d（text =“设置”）.click（超时时间= 10）

 ＃点击的别名

 d（text=“设置”）.tap（）

15、长按：d（text =“设置”）.long\_click（）

16、拖动元素到另一个位子：

 d（text =“Settings”）.drag\_to（x，y，duration = 0.5）

 d（text =“Settings”）.drag\_to（text =“Clock”，duration = 0.25）#clock的中心点

17、从一点到另一点的两点手势

 d（text =“Settings”）.gesture（（sx1，sy1），（sx2，sy2），（ex1，ey1），（ex2，ey2））

18、特定UI对象上的两点手势

 •从边缘到中心

 •从中心到边缘

 ＃从边缘到中心。这里是“以”不“在”

 d（text =“Settings”）。pinch\_in（percent = 100，steps = 10）

 ＃从中心到边缘

 d（text= “设置”）pinch\_out。（）

19、等待用户界面出现或消失

 d（text =“Settings”）。wait（timeout = 3.0）＃return bool

 d（text= “设置”）。wait\_gone（超时= 1.0）

20、＃设置默认元素等待超时（秒）

 d.wait\_timeout = 30.0

21、控件不知道控件的情况下

 d.set\_fastinput\_ime（真）＃切换成FastInputIME输入法

 d.send\_keys（“你好123abcEFG”）＃adb广播输入

 d.clear\_text（）＃清除输入框所有内容（需要android-uiautomator.apk版本> = 1.0.7）

 d.set\_fastinput\_ime（假）＃切换成正常的输入法

22，显示：吐司

 d.make\_toast（“Hello world”）

 d.make\_toast（“Hello world”，1.5）＃显示1.5秒

child\_by\_description 就是找到孙子有特定描述的儿童，其他参数与之相似child\_by\_text 。

child\_by\_instance 就是要找到在其子层次结构中指定实例的任何位置具有子UI 元素的子元素。在它滚动没有可见的视图上执行。

详细信息请参阅下面的链接： • UiScrollable ，getChildByDescription ，getChildByText ，getChildByInstance

12、常见的问题：

    如图1所示，发现SDK的某些功能或手机的不能用：直接打开uiautomator 的应用程序（INIT 成功后，就会安装上的），点击关闭UIAutomator  或者d.service （“uiautomator” ）.stop （）

    2502错误：

控制台：adb shell am instrument -w -r -e debug false -e class com.github.uiautomator.stub.Stub \  
  com.github.uiautomator.test / android.support.test.runner.AndroidJUnitRunner

13. 复杂屏幕操作

2.等待对象显示并点击，等待最多10s：

d(text="Settings").click(timeout=10)

3.当10s内对象出现就点击，默认0s：

clicked = d(text='Skip').click\_exists(timeout=10.0)

4.点击并轮询对象直到消失，其中maxretry为最多点击次数，默认10；interval为轮询时间间隔，默认1：

d(text="Skip").click\_gone(maxretry=10, interval=1.0)

5.长按指定的对象，duration为按住时间，timeout为超时时间：

d(text="Settings").long\_click(duration=10,timeout=10)

6.拖动指定对象到x,y位置，0.5s完成：

d(text="Settings").drag\_to(x, y, duration=0.5)

7.拖动指定对象到另一个对象位置，0.5s内完成：

d(text="Settings").drag\_to(text="Clock", duration=0.5)

8.双指从(a,b)，(c,d)滑动(a1,b1)，(c1,d1)，步长100：

d().gesture((a,b)，(c,d), a1,b1)，(c1,d1)，steps=100)

9.从屏幕外侧向中心滑动，percent为左右起始位置占两边的比例：

d().pinch\_out(percent=33, steps=100)

10.从屏幕中心向外侧滑动，percent为左右起始位置占两边的比例：

d().pinch\_out(percent=33, steps=100)

11.等待对象：

d(text='相机').wait(10)

12.等待对象消失，最多等待10s：

d(text='相机').wait\_gone(timeout=10)

13.滚动：

a.向上滚动：d(scrollable=True).scroll(steps=10)

b.向下滑动：d(scrollable=True).scroll.vert.backward()

c.水平向右滚动：d(scrollable=True).scroll.horiz.forward(steps=50)

d.水平向左滚动：d(scrollable=True).scroll.horiz.backward(steps=50)

e.水平滑动到最左边：d(scrollable=True).scroll.horiz.toBeginning(steps=100, max\_swipes=1000)

f.水平滑动到最右边：d(scrollable=True).scroll.horiz.toEnd(steps=100, max\_swipes=1000)

g.竖直滑动到结尾：d(scrollable=True).scroll.toEnd()

h.竖直滑动到开头：d(scrollable=True).scroll.toBeginning(steps=50)

i.滑动到指定位置（测试）：d(scrollable=True).scroll.to(text ='测试')

#### 截图

d.screenshot（ “ home.jpg ”） # get PIL.Image格式化图像，你需要首先安pillow   
image = d.screenshot（） # default format =“pillow”

image.save（ “ home.jpg ”）＃或home.png。目前，只有PNG和JPG支持

//

脚本框架选择

Pytest+allure（版本需相同，若不同则报告无法生成）

Pip install allure-pytest（建议在pycharm设置中手动导入，可选择对应版本）

Allure 需下载第三方包并添加环境变量

脚本编写可在weditor界面工具中完成

转换为python脚本需在开头添加：

import uiautomator2 as ut2

d = ut2.connect\_usb('设备编号')#设备编号可通过cmd命令adb devices获取